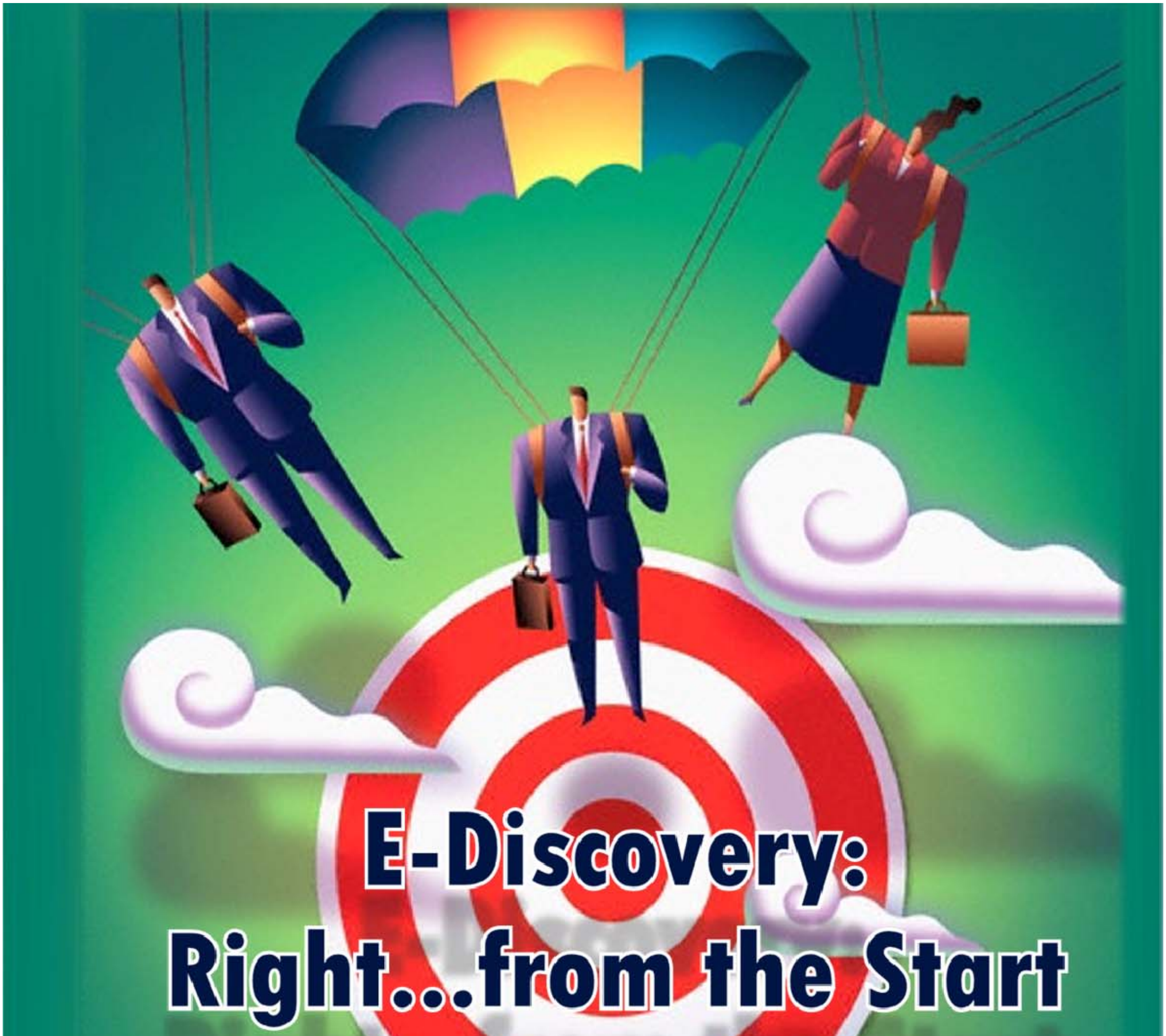


E-Discovery: Right...from the Start

Craig Ball

LSC TIG Conference 2010: E-Discovery in Small Cases:

Right from the Start: 8 Ways to Hit the Ground Running
E-Discovery for Everybody: The EDna Challenge
Surefire Steps to Splendid Search
Geek Speak: Lawyer's Guide to the Language of Data Storage and Networking
Meeting the Challenge of E-Mail in Civil Discovery
Technology Primer: Backups in Civil Discovery
Selected BIYC Columns on Lower Cost Approaches to E-Discovery

E-Discovery: Right…from the Start
LSC TIG Conference 2010: E-Discovery in Small Cases:
Craig Ball

About this Collection

America's halcyon days of hammer and harness are behind us. We are all knowledge workers now. Even those who drive trucks or empty bedpans are tasked by pixels and tracked by bytes. The evidence of what we do and say, of when and where and how we go, of what we own and earn and spend, is digital. *More than 99% of it will never exist as anything but electronically stored information*, and most takes forms that require special tools or expertise to see and interpret.

A lawyer without the skills needed to properly search electronic evidence is all-but-incompetent to manage litigation today. The evidence is digital. It's there. It's waiting for you--eager to tell its compelling story, ready to show your client was right and the other side should pay big or go hence without day. The lawyer who can get to the digital evidence—find it, understand it and use it--enjoys an enormous advantage.

It's an advantage within the reach and budget of lawyers and litigants in almost any case. Like the courts themselves, access to evidence can't be just for the privileged.

The selected articles and columns that follow were chosen because they illustrate lower cost, brainy-not-brawny approaches to electronic discovery. They are a small sample of the articles I've written about electronic discovery and computer forensics, many available at **www.craigball.com**. I hope you find them, along with my blog posts, webcasts and other resources, to be a valuable, accessible introduction to the technology and best practices of electronic discovery. Thanks!

Craig Ball, January 2009

Right from the Start
Smart First Steps in Electronic Discovery

Craig Ball
© 2009

Certainly it's smart to prepare for e-discovery—to be "proactive" about electronically stored information (ESI) and implement early case assessment systems and strategies. But sometimes, the lawsuit's the first sign of trouble, and you have to choose which fires to fight…and fast.

Don't be paralyzed by fear of failure or confusion about where to begin. There are no perfect e-discovery efforts. Before the ESI experts come aboard, there are things you can and must do. Here's a quick compendium of eight ways to hit the ground running:

1. **Apply the five Ws of good journalism—who, what, when, where and why—**to get a handle on your core preservation duties. Immediately make a list of the people, events, time intervals, business units, records and communications central to the case.
    a. List the apparent key players (don't forget assistants who, *e.g.,* handle the boss' email and significant third parties over whom your client has a right of direction or control).
    b. Hone in on what happened—both from your perspective and theirs—and posit what ESI sheds light either way or tends to explain or challenge the key players' actions and attitudes.
    c. Decide what dates and time periods are relevant for preservation. Is there a continuing preservation obligation going forward?
    d. Determine which business units, facilities, systems and devices most likely hold relevant ESI.

    Your lists will change over time, but a focused, thoughtful and well-documented effort, diligently implemented, is more defensible, less costly and invariably more effective than a scattershot approach. Don't delay. It needn't be flawless right now; reasonable will do.

2. **Focus on the fragile first.** What potentially relevant ESI has the shortest shelf life and requires quickest action to preserve while it's still reasonably accessible? Voice mail, web mail and text messaging, computers requiring forensic examination, web content and surveillance video are examples of ESI that tend to be rapidly discarded or overwritten. Grabbing e-mail of key custodians *before* it migrates to backup media can save a bundle and accelerate search and processing.

3. **Protect employees from themselves.** People who wouldn't dream of shredding a paper record will purge ESI with nary a thought. In the blink of an eye, history will be reinvented as employees delete overly candid e-mail and commingled personal and business communications. The results are often catastrophic and always costly. Assess whether those entrusted with preservation can be trusted to perform, and don't rely on custodial preservation *alone* when its failure is reasonably foreseeable.

4. **Holds should be instructional, not merely aspirational.** Too many lawyers draft legal hold instructions designed to protect lawyers. Broadly disseminating a form hold directive saying "keep everything" isn't helpful and will come back to haunt you at deposition. "I *got* that memo," they say, "but I didn't *do* anything."

   Custodians need to know where to start. Tell them what to do and how to do it. Give examples that inform and deadlines that demand action. Get management buy in for the time needed to comply. Better a handful of key players take the hold directive seriously than dozens or hundreds of minor players wink at it.

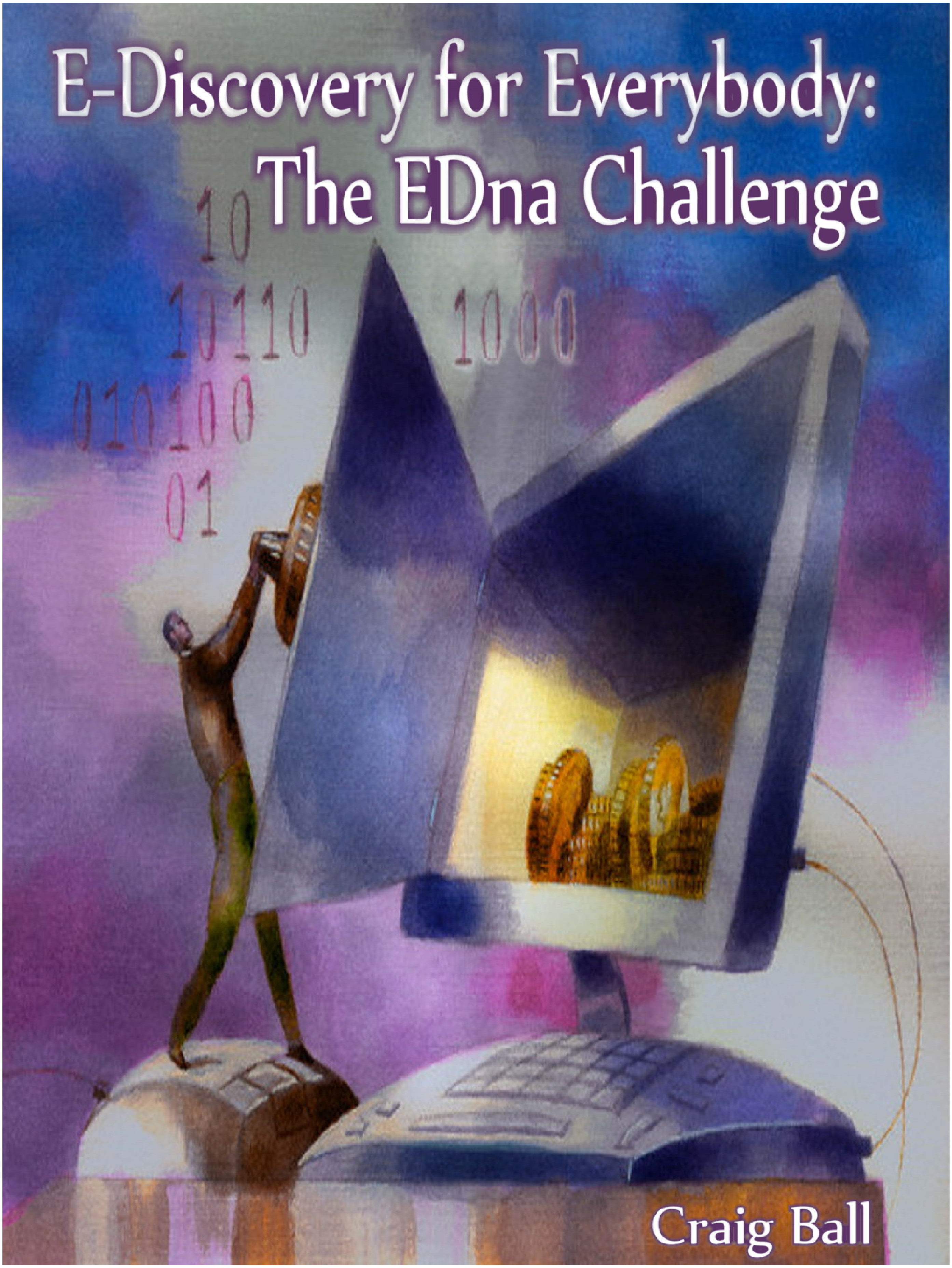
5. **Boots on the ground.** Good doctors don't diagnose over the phone. Likewise, good lawyers meet key players and get a firsthand sense of how they operate. Seek out the people who manage the systems that hold the evidence, and learn the "who, what, when, where and why" of your client's ESI face-to-face. It's not just enormously helpful—it's what courts demand.

6. **Build the data map, including local collections and databases.** Federal practice requires identification of potentially relevant ESI, but it's a best practice everywhere. That goes for the less-accessible stuff, too. Courts won't accept, "We don't know what we have or where it is," so be ready to identify potentially relevant ESI that you will and won't ex**p**lore or produce. Data stored off the servers or on databases pose special challenges made harder by turning a blind eye to its existence. Don't fall prey to, "If we don't tell them we have it, they won't ask for it."

7. **Consider how you'll collect, store, search, review and produce ESI.** All ESI is just a bunch of ones and zeros. Making sense of it, controlling costs and minimizing frustrating "do-overs," rides on how you choose to process and produce information. So add an "H"—*How*—to those five Ws, and ponder your options for how the data gets from here to there.

8. **Engage the other side.** Even warring nations cease fire to carry off fallen comrades. You don't have to like or trust the opposition, but you have to be straight with them if you want to stay out of trouble in e-discovery. Tell the other side what you're doing and

what you’re unwilling to do. Collaborate anywhere you can. Lawyers over-discover cases more from ignorance and mistrust than guile or greed; but, even when you face someone gaming the system, your documented candor and good faith effort to cooperate will serve you well in court.



E-Discovery for Everybody: The EDna Challenge

Craig Ball

E-Discovery for Everybody: The EDna Challenge

Craig Ball

E-discovery is just for big budget cases involving big companies, handled by big firms.

Right, and suffrage is just for white, male landowners.

Some Neanderthal notions take longer than others to get shown the door, and it's time to dispel the mistaken belief that e-discovery is just for the country club set.

Today, evidence means *electronic* evidence; so, like the courts themselves, access to evidence can't be just for the privileged. Everyone gets to play.

If you think big firms succeed at e-discovery because they know more than you do, think again. Marketing hype aside, big firm litigators don't know much more about e-discovery than solo practitioners. Corporate clients hire pricey vendors with loads of computing power to index, search, de-duplicate, convert and manage terabytes of data. Big law firms deploy sophisticated in-house or hosted review platforms that let armies of associates and contract lawyers plow through vast plains of data--viewing, tagging, searching, sorting and redacting with a few keystrokes. *The big boys simply have better toys.*

A hurdle for everyone else is the unavailability and high cost of specialized software to process and review electronic evidence.

A Mercedes and a Mazda both get you where you need to go, but the e-discovery industry has no Mazdas on the lot. This article explores affordable, off-the-shelf ways to get where you need to go in e-discovery.

One Size Doesn't Fit All

First, let's set sensible expectations: Vast, varied productions of ESI cannot be efficiently or affordably managed and reviewed with software from Best Buy. If you're grappling with millions of files and messages, you'll need to turn to some pretty pricy power tools.

The key consideration is workflow. Tools designed for ESI review can save considerable time over cobbled-together methods employing off-the-shelf applications; and, when every action is extrapolated across millions of messages and documents, seconds saved add up to big productivity gains.

But few cases involve millions of files. Most entail review of material collected from a handful of custodians in familiar productivity formats like Outlook e-mail, Word documents, Excel spreadsheets and PowerPoint presentations. Yes, volume is a challenge in these cases, too; but, a mix of low-cost tools and careful attention to process makes it possible to do defensible e-discovery on the cheap.

Paper Jam

More from comfort than sense, ESI in smaller cases tends to be printed out. Paper filled the void for a time, but lately the cracks are starting to show. Lawyers are coming to appreciate that printing evidence isn't just more expensive and slower, it puts clients at an informational disadvantage.

When you print an electronic documents, you lose three things: Money, time and metadata. Money and time are obvious, but the impact of lost metadata is often missed. When you move ESI to paper or paper-like formats like TIFF images, you cede most of your ability to search and authenticate information, along with the ability to quickly and reliably exclude irrelevant data. Losing metadata isn't about missing the chance to mine embedded information for smoking guns. That's secondary. Losing metadata is like losing all the colors, folders, staples, dates and page numbers that help paper records make sense.

The EDna Challenge

I polled a group of leading e-discovery lawyers and forensic technologists to see what tools and techniques they thought suited to the following hypothetical:

> Your old school chum, Edna, runs a small firm and wants your advice. A client is about to send her two DVDs containing ESI collected in a construction dispute. It will be Outlook PST files for six people and a mixed bag of Word documents, Excel spreadsheets, PowerPoint presentations, Adobe PDFs and scanned paper records sans OCR. There could be a little video, some photographs and a smattering of voicemail in WAV formats. "Nothing too hinky," she promises. Edna's confident it will comprise less than 50,000 documents and e-mails, but it could grow to 100,000 items before the case concludes in a year or two.
>
> Edna's determined to conduct an in-house, paperless privilege and responsiveness review, sharing the task with a tech-savvy associate and legal assistant. All have late-model, big screen Windows desktop PCs with MS Office Professional 2007 and Adobe Acrobat 9.0 installed. The network file server has ample available storage space. Edna doesn't own Summation or Concordance, but she's willing to spend up to $1,000.00 for new software and hardware, but not a penny more. She's open to an online Software as a Service (SaaS) option, but the review has to be completed using just the hardware and software she currently owns, supplemented only by the $1,000.00 in new purchases. Her team will supply as much brute force as necessary. She's too proud to accept a loan of systems or software, and you can't change her mind or budget.
>
> ***How should Edna proceed?***

Goals of the Challenge

Ideally, the review method employed should:

1. Preserve relevant metadata;
2. Incorporate de-duplication, as feasible;
3. Support robust search of Outlook mail and productivity formats;
4. Allow for efficient workflow;

5. Enable rudimentary redaction;
6. Run well on most late-model personal computers; and
7. Require no more than $1,000.00 in new software or hardware, though it's fine to use fully-functional "free trial" software so long as you can access the data for the 2-3 year life of the case.

I had some ideas (shared later in this article), but expected my colleagues might point me to better mousetraps. Instead, I was struck by the familiarity and consistency of their excellent suggestions as compared to options that have been around for years. Sadly, there's not that much new for those on shoestring budgets; that is, developers remain steadfastly disinterested in 85% of the potential market for desktop discovery tools.

One possible bright spot was the emergence of hosted options. No one was sure the job could be begun--let alone completed--using SaaS on so tight a budget; but, there was enough mention of Saas to make it seem like a possibility, now or someday soon.

Advice to Edna

While the range of proposals was thin, the thought behind them was first-rate. All responding recognized the peril of using the various Microsoft applications to review the ESI. Outlook's search capabilities are limited, especially with respect to attachments. If Edna expected to reliably search inside of every message, attachment and container file, she would need more than Outlook alone.

Notable by their absence were any suggestions to use Google's free desktop indexing and search tool. Though a painful interface for e-discovery, Google Desktop installed on a dedicated, "clean" machine would be capable of reading and searching Outlook e-mail, Word documents, Excel spreadsheets, PowerPoint presentations, PDF files, Zip archives and even text within music, video and image files. It wouldn't be pretty--and Edna would have to scrupulously guard against cross-contamination of the evidence with other data--but Google Desktop *might* get much of the job done without spending a penny.

Quin Gregor of Strategic Data Retention LLC in Georgia was first to respond with an endorsement of my two favorite affordable workhorses, the ubiquitous dtSearch indexing and search tool ($199.00 at www.dtsearch.com) and Aid4Mail ($69.95 at www.fookes.com), a robust utility for opening, filtering and converting common e-mail container files and message formats. Quin described a bankruptcy case where a microscopic budget necessitated finding a low-end option. He reports that dtSearch and Aid4Mail saved the day.

Ron Chichester, an attorney and forensic examiner in Texas pointed to the many open source Linux tools available without cost. These command line interface tools are capable of indexing, Bayesian analysis and much of the heavy lifting of the tools used by e-discovery vendors; but. Ron acknowledged that Edna and her staff would need a lot of Linux expertise to integrate the open source offerings. Bottom line: The price is right, but the complexity unacceptable.

Florida e-discovery author and blogger, Ralph Losey, a partner at AkermanSenterfitt, suggested using an online review tool like Catalyst and tried to dance around the budget barrier by pointing out that the cost could be passed on to the client. Ralph argued that hosting would save enough lawyer time to pay for itself. No doubt he's right; but, passing on the costs isn't permitted in the Edna Challenge and, even in a real world situation, unless the savings were considerable, Edna's likely to keep the work--and the revenue--in house.

Another Floridian, veteran forensic examiner, Dave Kleiman, suggested that Edna blow her budget on alcohol and amphetamines because she has a lot of toil ahead of her. Party on, Dave!

Our northern neighbor, Dominic Jaar of Ledjit Consulting Inc. in Quebec, took a similar doleful tack. Dominic thought that SaaS might be a possibility but added that Edna should use her grand to take an e-discovery course because she needs to learn enough to "stay far from the case." Else, he offered, she could go forward and apply the funds to coffee and increased malpractice coverage. *Ouch!*

John Simek of Sensei Enterprises in Virginia prudently suggested that Edna use part of her budget to buy an hour of a consultant's time to help her get started. John predicted that a SaaS approach would be priced out-of-reach, but was another who thought salvation lay with dtSearch. John recognized that Adobe Acrobat could handle both the redaction and light-duty OCR required. As for the images, video and sounds, Edna's in the same boat, rich or poor. She's just going to have to view or listen to them, one-by-one.

Jerry Hatchett with Evidence Technology in Houston suggested LitScope, a SaaS offering from LitSoft. Jerry projected a cost of around $40/GB/month, which would burn through Edna's budget in about 3 months...if she didn't buy any Starbucks. Following up, I discovered that LitScope can't ingest the native file formats Edna needed to review unless accompanied by load files containing the text and metadata of the documents and messages. The cost to pre-process the data to load it would eat up Edna's budget before she looked a single page. That, and a standard $200 minimum on monthly billings coupled with a 6 month minimum commitment, made this SaaS option a non-starter. Attractive pricing, to be sure, but not low enough for Edna's shallow pockets.

The meager budget forced George Rudoy, Director of Global Practice Technology & Information Services at Shearman & Sterling, LLP in New York, to suggest using Outlook 2007 as the e-mail review tool, adding the caveat that metadata may change. Unlike earlier versions, Outlook 2007 claims to extend its text search capabilities to attachments. Unfortunately, it doesn't work very well in practice, meaning Edna and her staff will need to examine each attachment instead of ruling any out by search. George also urged Edna to buy licenses for Quick View Plus--a universal file viewer utility--and hire an Access guru to design a simple database to track the files and hyperlink to each one for review.

From Down Under, Michelle Mahoney of Mallesons Stephen Jaques in Melbourne shared several promising approaches. She suggested Karen's Power Tools (a $30 suite of applications at www.karenware.com) as a means to inventory and hash the files and Microsoft Access as a means to de-duplicate by hash values. Michelle also favored hyperlinking from Access for review, working through the collection progressively, ordering them by file type and then filename. She envisions adding fields to the database for Relevant and Privileged designations and a checkbox for exceptional files that can't be opened and require further work.

For the e-mail files, Michelle also turns to Outlook as a review tool, proposing that folders be created for dragging-and-dropping items into Relevant Non Privileged; Relevant Privileged and Non Relevant groups. She echoed warnings about metadata modification and gives her thumbs up to Aid4Mail.

Finally, Michelle offers more kudos for dtSearch as the low cost tool-of-choice for keyword searching. dtSearch will allow Edna to run keywords across files, including emails and attachments, and it is a simple file copy option to copy them, with or without original path, into a folder. Messages emerge in the generic MSG mail format, and Edna can either produce them in that format (with embedded attachments) or use Aid4Mail to copy them into an Outlook PST file format. For further discussion of using dtSearch as a low-cost e-discovery tool, see, Craig Ball, *Do-It-Yourself Digital Discovery***,** (Law Technology News, May 2006); *infra* at 37.

Tom O'Conner, Director of the Legal Electronic Document Institute in New Orleans, observed that he often gets requests like Edna's from his clients in Louisiana and Mississippi and weighed in with a mention of Adobe Acrobat, noting that it might be feasible to print everything to Acrobat and use Acrobat's annotation and redaction features. As mentioned, Acrobat also offers rudimentary OCR capabilities to help deal with the scanned paper documents in the collection and even has the ability to convert modest volumes of e-mail to PDFs directly from Outlook. For further discussion of using Adobe Acrobat to process Outlook e-mail, see, Craig Ball, *Adobe Brings an Acrobat to Perform EDD* (Law Technology News, June 2008); *infra* at 54.

Tom concludes that, although working with the tools you already own and know can be cumbersome, it's sometimes a better approach that trying to master new tools under pressure.

Ohio-based e-discovery consultant, Brett Burney, had some very concrete ideas for Edna. He thought she could try to find some SaaS solution to host the data, suggesting Lexbe, NextPoint or Trial Solutions as candidates. Brett was most familiar with Lexbe and knew of small law firms that had successfully and inexpensively used their services.

Brett guessed Edna's budget might allow her to upload everything to Lexbe, review it quickly and then take everything down before the hosting costs ate up her budget. He reported that Lexbe will accept about any file format, by uploading it yourself or sending it to Lexbe to load. Brett put the cost at $99 per month for 2 users and 1GB of storage. Noting that Edna needs to host more than 1GB of data, he predicted her outlay should be close to $200/month. Brett added, "Edna and her crew can upload everything with the tools they have, get it reviewed pronto (i.e. less than a month), and then take everything down--paying only for what they use."

For the Outlook e-mail, Brett thought Edna should turn to Adobe Acrobat and convert the PST container files to PDF Portfolios along the lines of my June 2008 column. Alternatively, Brett suggested Edna use the free Trident Lite tool from Wave Software (www.discoverthewave.com) to get a "snapshot" of the PSTs and then convert relevant messages to PDF or upload them to a hosting provider.

Lisa Habbeshaw of FTI in California pointed to Intella by Vound Software (http://www.vound-software.com) as an all-in-one answer to Edna's needs. Intella offers an efficient indexing engine, user-friendly interface and innovative visual analysis capability sure to make quick work of Edna's review effort. Lisa was unsure if the program could be had for under $1,000, but noted that Vound Software offers a free, fully-functional demo that might fill the bill for Edna's immediate needs. Like Lisa, I'm unsure whether Intella will bust Edna's budget, but it's certainly a splendid new entry to the do-it-yourself market.

Other Great Tools

If the dollar holds its own against the Euro, Edna could accomplish just about everything she needs to do using a terrific tool created in Germany called X-Ways Forensics from X-Ways Software Technology AG. X-Ways Forensics could make quick work of the listing, hashing, opening, viewing, indexing, searching, categorizing and reporting on all that client data; however, it's a complex, powerful forensics tool that would require more time and training to master than Edna can spare. Plus, it would eat up all of her $1,000 budget.

If her budget was bigger, Edna would be very happy attacking the review with the easy-to-use, fast and versatile Nuix Desktop (www.nuix.com). Nuix would allow Edna to begin her review in minutes, and it supports a host of search options. The embedded viewer, hash and classification features foster an efficient workflow and division of review among multiple reviewers. Like Intella, Nuix is an Australian import. Whatever they're doing way down there in Kangaroo land, they're certainly doing something right!

A Few More Ideas for Edna

It's hard to add much to so many fine ideas. Collectively, dtSearch, Adobe Acrobat and Aid4Mail deliver the essential capabilities to unbundle, index, search, OCR and redact the conventional file formats and modest data volumes Edna faces. Her challenge will be cobbling together tools not designed for e-discovery so as to achieve an acceptable workflow and defensible tracking methodology. It won't be easy.

For example, while dtSearch is Best of Class in its price range, it doesn't afford Edna any reasonable way to tag or annotate documents as she reviews them. Accordingly, Edna will be obliged to move each document to a folder as she makes her assessments respecting privilege and responsiveness. That effort will get very old, very fast.

On the plus side, dtSearch offers a fully functional thirty-day demo of its desktop version, so Edna can buy a copy for her long-term use, but rely on 30-day evaluation copies for her staff during the intense review effort--a $400 savings.

While Adobe Acrobat supports conversion of e-mail into PDFs, the process is painfully slow and cumbersome. Moreover, the conversion capabilities break down above 10,000 messages. That sounds like a lot, but it's likely less than Edna will see emerge in the collections of six custodians. Further, Edna may encounter an opponent who smart enough to demand the more versatile electronic formats for e-mail (i.e., PST, MSG or EML). What's Edna going to do if she finds herself locked into a reviewed wedded to image formats?

Whatever tools she employs, Edna will need to be meticulous in her shepherding of the individual messages and documents through the process. To that end, I'd offer this advice:

1. Your first step should be to make a working copy of the data to be processed and secure the source dataset against any usage or alteration. Processing of ESI poses risks of data loss or alteration. If errors occur, you must be able to return to uncorrupted data from prior steps. For each major processing threshold, set aside a copy of the data for safekeeping and carefully document the time the data was set aside and what work had been done to that point (e.g., the status of deduplication, filtering and redaction).

2. From the working copy, hash the files and generate an inventory of all files and their metadata. The processes you employ must account for the disposition of every file in the source collection or extracted from those files (i.e., message attachments and contents of compressed archives). Your accounting must extend from inception of processing to production. By hashing the constituents of the collection as it grows, you gain a means to uniquely identify files as well as a way to identify identical files across custodians and sources.

   A useful tool for hashing files is Karen's Hasher available at http://www.karenware.com. But the best "free" tool for the task is AccessData's FTK Imager, available from www.accessdata.com/downloads. FTK Imager not only hashes files, it also exports Excel-compatible comma delimited listings of filenames, file paths, file sizes and modified, accessed and created dates. Moreover, it supports loading the collected files into a container called a Custom Content Image that protects the data from metadata corruption.

3. Devise a logical division scheme for the components of the collection; e.g., by machine, custodian, business unit or otherwise. Be careful not to aggregate files in a manner that files from one source may overwrite identically named files from other sources.

4. Expand files that hold messages and other files. Here, you should identify e-mail container files (like Outlook .PST files) and archives (e.g., .Zip files) that must be opened or decompressed to make their constituents amenable to search. For e-mail, this can be done using an inexpensive utility like Aid4mail from Fookes Software or Trident Lite from Wave Software. Additionally, e-mail client applications, including Outlook, usually permit export of individual messages and attachments. Though dtSearch includes a command line utility to convert Outlook PST container files to individual messages (.MSG) files for indexing, it doesn't work well or easily compared to Aid4Mail. Finally, most indexing tools are capable of directly accessing text within compressed formats. For example, DTSearch can extract text from Zip files and other archives.

5. A feature common to premium e-discovery tools but hard to match with off-the-shelf software is deduplication. You can use hash values to identify identical files, but the challenge is to keep track of all de-duplicated content and reliably apply tagging for privilege and responsiveness to all deduplicated iterations. Most off-the-shelf utilities simply eliminate duplicates and so aren't suited to e-discovery.

   This is where it's a good investment to secure help from an expert in Microsoft Excel or Access because those applications can be programmed to support deduplication tracking and tagging.

   When employing deduplication, keep in mind that files with matching hash values can have different filenames and dates. The hash identicality of two files speaks to the *contents* of the files, *not* the names assigned to the files by the operating system or to information, like modified, accessed and created dates, stored *outside* the files.

6. Above all, don't process and review ESI in a vacuum. Be certain that you understand the other side's expectations in terms of the scope of the effort, approach to search and--critically--the forms of production they seek. You may not agree on much, but you may be pleasantly surprised to learn that some of the perils of a low budget e-discovery effort (e.g., altered metadata, limited search capabilities, native production formats) don't concern the other side. Further, you may reach accord on limiting the scope of review in terms of time intervals, custodians and types of data under scrutiny. Why look at *all* the e-mail if the other side is content with your searching just communications between Don and Betty during the third week of January 2009?

Finally, Edna may seek an answer to two common questions from those taking the do-it-yourself route in e-discovery:

What if I change metadata?
Certain system metadata values--e.g., last access times and creation dates--are prone to alteration when processed using tools not designed for e-discovery. Such changes are rarely a problem if you adhere to three rules:

1. **Preserve** an unaltered copy of whatever you're about to process;
2. **Understand** what metadata were altered; and,
3. **Disclose** the changes to the requesting party.

By keeping a copy of the data at each step, you can recover true metadata values if particular values proves significant. Then, disclosing what metadata values were changed eliminates any suggestion that you pulled a fast one. Many requesting parties have little regard for system metadata values; but, they don't want to be surprised by relying on inaccurate information.

Can I Use My Own E-Mail Account for Review?

You wouldn't commingle client funds with your own money, so why commingle e-mail that's evidence in a case with your own mail? That said, when ESI is evidence and the budget leaves no alternative, you may be forced to use your own e-mail tools for small-scale review efforts. If so, remember that you can create alternate user accounts within Windows to avoid commingling client data with your own. Better still, undertake the review using a machine with a clean install of the operating system. Very tech-savvy counsel can employ virtual environments (e.g., VMWare products) to the same end.

If using an e-mail client for review, it may be sufficient to categorize messages and attachments by simply dragging them to folders representing review categories; for example:

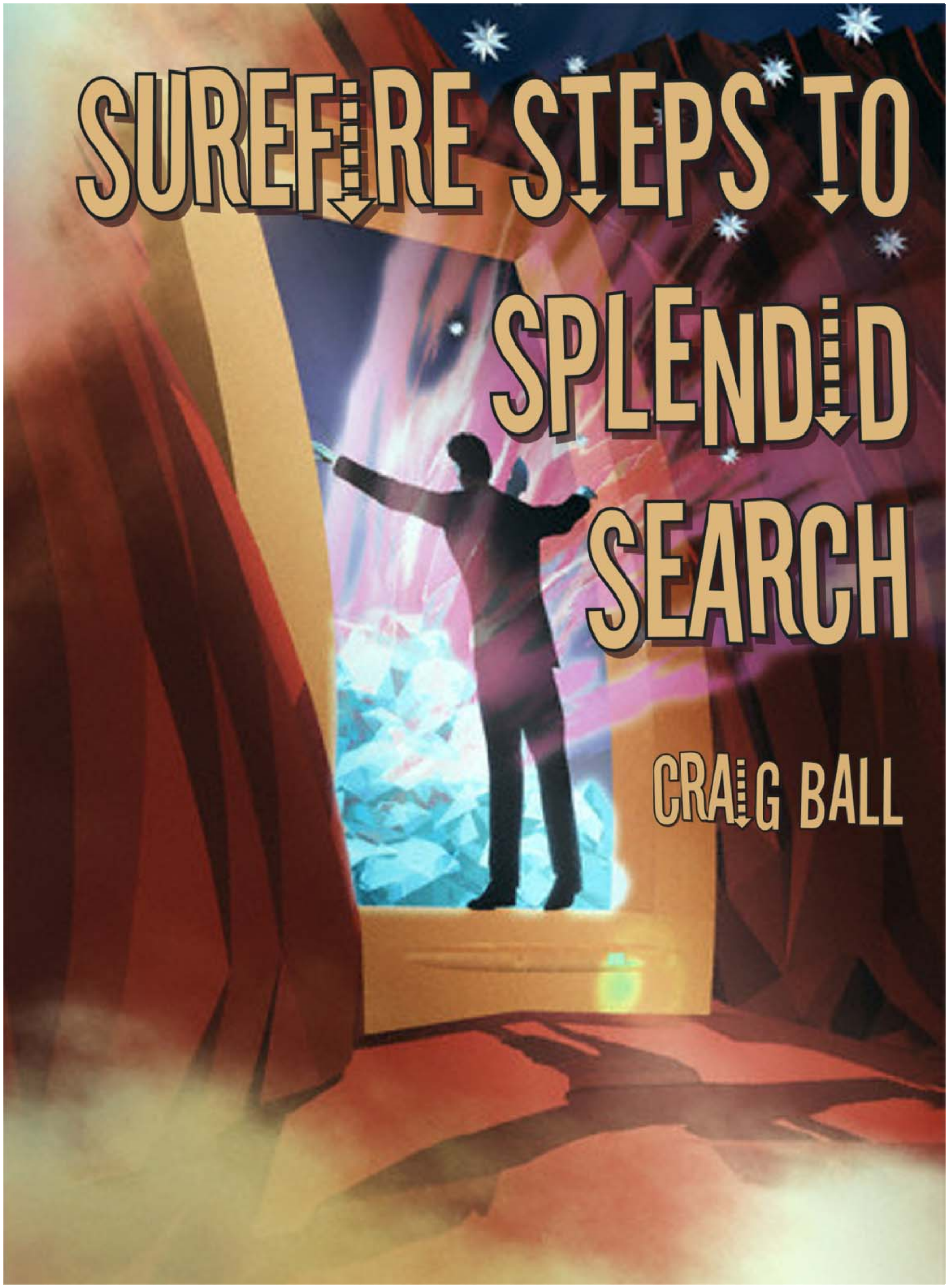
1. Attorney-client privilege: entire item;
2. Work product privilege: entire item;
3. A-C Privilege: needs redaction;
4. W-P privilege: needs redaction;
5. Other privilege;
6. Responsive;
7. Non-responsive.

Once categorized, the contents of the various folders can be exported for further processing or for production, if in a suitable format.

Throwing Down The Gauntlet

The vast majority of cases filed, developed and tried in the United States are not multimillion dollar dust ups between big companies. The evidence in modest cases is digital, too. Solo and small firm counsel like Edna need affordable, user-friendly tools designed for desktop e-discovery--tools that preserve metadata, offer efficient workflow and ably handle the common file formats that account for nearly all of the ESI seen in day-to-day litigation. Using the tools and techniques described by my thoughtful colleagues, Edna will get the job done on time and under budget. The pieces are there, though the integration falls short.

So, how about it e-discovery industry? Can you divert your gaze from the golden calf long enough to see the future and recall the past? Sam Walton became the richest man of his era by selling to more for less. There's a fast growing need...and a huge emerging market. The *real* Edna Challenge is waiting for the visionaries who will meet the need and serve the market.


SUREFIRE STEPS TO
SPLENDID
SEARCH
CRAIG BALL

Surefire Steps to Splendid Search

Craig Ball

Hear that rumble? It's the bench's mounting frustration with the senseless, slipshod way lawyers approach keyword search.

It started with Federal Magistrate Judge John Facciola's observation that keyword search entails a complicated interplay of sciences beyond a lawyer's ken. He said lawyers selecting search terms without expert guidance were truly going "where angels fear to tread."

Federal Magistrate Judge Paul Grimm called for "careful advance planning by persons qualified to design effective search methodology" and testing search methods for quality assurance. He added that, "the party selecting the methodology must be prepared to explain the rationale for the method chosen to the court, demonstrate that it is appropriate for the task, and show that it was properly implemented."

Most recently, Federal Magistrate Judge Andrew Peck issued a "wake up call to the Bar," excoriating counsel for proposing *thousands* of artless search terms.

> Electronic discovery requires cooperation between opposing counsel and transparency in all aspects of preservation and production of ESI. Moreover, where counsel are using keyword searches for retrieval of ESI, they at a minimum must carefully craft the appropriate keywords, with input from the ESI's custodians as to the words and abbreviations they use, and the proposed methodology must be quality control tested to assure accuracy in retrieval and elimination of 'false positives.' It is time that the Bar—even those lawyers who did not come of age in the computer era—understand this.

No Help

Despite the insights of Facciola, Grimm and Peck, lawyers still don't know what to do when it comes to effective, defensible keyword search. Attorneys aren't *trained* to craft keyword searches of ESI or implement quality control testing for same. And their experience using Westlaw, Lexis or Google serves only to inspire false confidence in search prowess.

Even saying "hire an expert" is scant guidance. Who's an expert in ESI search for your case? A linguistics professor or litigation support vendor? Perhaps the misbegotten offspring of William Safire and Sergey Brin?

The most admired figure in e-discovery search today—*the Sultan of Search*—is Jason R. Baron at the National Archives and Records Administration, and Jason would be the first to admit he has no training in search. The persons most qualified to design effective search in e-discovery earned their stripes by spending thousands of hours running searches in real cases--making mistakes, starting over and tweaking the results to balance efficiency and accuracy.

The Step-by-Step of Smart Search

So, until the courts connect the dots or better guidance emerges, here's my step-by-step guide to craftsmanlike keyword search. I promise these ten steps will help you fashion more effective, efficient and defensible queries.

1. **Start with the Request for Production**
2. **Seek Input from Key Players**
3. **Look at what You've Got and the Tools you'll Use**
4. **Communicate and Collaborate**
5. **Incorporate Misspellings, Variants and Synonyms**
6. **Filter and Deduplicate First**
7. **Test, Test, Test!**
8. **Review the hits**
9. **Tweak the Queries and Retest**
10. **Check the Discards**

1. Start with the Request for Production

Your pursuit of ESI should begin at the first anticipation of litigation in support of the obligation to identify and preserve potentially relevant data. Starting on receipt of a request for production (RFP) is starting late. Still, it's against the backdrop of the RFP that your production efforts will be judged, so the RFP warrants careful analysis to transform its often expansive and bewildering demands to a coherent search protocol.

The structure and wording of most RFPs are relics from a bygone time when information was stored on paper. You'll first need to hack through the haze, getting beyond the "any and all" and "touching or concerning" legalese. Try to rephrase the demands in everyday English to get closer to the terms most likely to appear in the ESI. Add terms of art from the RFP to your list of keyword candidates. Have several persons do the same, insuring you include multiple interpretations of the requests and obtain keywords from varying points of view.

If a request isn't clear or is hopelessly overbroad, push back promptly. Request a clarification, move for protection or specially except if your Rules permit same. Don't assume you can trot out some boilerplate objections and ignore the request. If you can't make sense of it, or implement it in a reasonable way, tell the other side how you'll interpret the demand and approach the search for responsive material. Wherever possible, you want to be able to say, "We told you what we were doing, and you didn't object."

2. Seek Input from Key Players

Judge Peck was particularly exercised by the parties' failure to elicit search assistance from the custodians of the data being searched. Custodians are THE subject matter experts on their own data. Proceeding without their input is foolish. Ask key players, "If you were looking for responsive information, how would you go about searching for it? What terms or names would likely appear in the messages we seek? What kinds of attachments? What distribution lists would have been used? What intervals and events are most significant or triggered discussion?" Invite custodians to show you examples of responsive items, and carefully observe how they go about conducting their search and what they offer. You may see them take steps they neglect to describe or discover a strain of responsive ESI you didn't know existed.

Emerging empirical evidence underscores the value of key player input. At the latest TREC Legal Track challenge, higher precision and recall seemed to closely correlate with the amount of time devoted to questioning persons who understood the documents and why they were relevant. The need to do so seems obvious, but lawyers routinely dive into search before dipping a toe into the pool of subject matter experts.

3. Look at what You've Got and the Tools You'll Use

Analyze the pertinent documentary and e-mail evidence you have. Unique phrases will turn up threads. Look for words and short phrases that tend to distinguish the communication as being about the topic at issue. What content, context, sender or recipients would prompt you to file the message or attachment in a responsive folder had it occurred in a paper document?

Knowing what you've got also means understanding the forms of ESI you must search. Textual content stored in TIFF images or facsimiles demands a different search technique than that used for e-mail container files or word processed documents.

You can't implement a sound search if you don't know the capabilities and limitations of your search tool. Don't rely on what a vendor tells you their tool can do, test it against actual data and evidence. Does it find the responsive data you already know to be there? If not, why not? Any search tool must be able to handle the most common productivity formats, e.g., .doc, docx, .ppt, .pptx, .xls. .xlsx, and .pdf, thoroughly process the contents of common container files, e.g., .pst, .ost, .zip, and recurse through nested content and e-mail attachments.

As importantly, search tools need to clearly identify any "exceptional" files unable to be searched, such as non-standard file types or encrypted ESI. If you've done a good job collecting and preserving ESI, you should have a sense of the file types comprising the ESI under scrutiny. Be sure that you or your service providers analyze the complement of file types and flags any that can't be searched. Unless you make it clear that certain files types won't be searched, the natural assumption will be that you thoroughly searched all types of ESI.

4. Communicate and Collaborate

Engaging in genuine, good faith collaboration is the most important step you can take to insure successful, defensible search. Cooperation with the other side is not a sign of weakness, and courts expect to see it in e-discovery. Treat cooperation as an opportunity to show competence and readiness, as well as to assess your opponent's mettle. What do you gain from wasting time and money on searches the other side didn't seek and can easily discredit? Won't you benefit from knowing if they have a clear sense of what they seek and how to find it?

Tell the other side the tools and terms you're considering and seek their input. They may balk or throw out hundreds of absurd suggestions, but there's a good chance they'll highlight something you overlooked, and that's one less do over or ground for sanctions*.* Don't position cooperation as a trap nor blindly commit to run all search terms proposed. "We'll run your terms if you agree to accept our protocol as sufficient" isn't fair and won't foster restraint. Instead, ask for targeted suggestions, and test them on representative data. Then, make expedited production of responsive data from the sample to let everyone see what's working and what's not.

Importantly, frame your approach to accommodate at least two rounds of keyword search and review, affording the other side a reasonable opportunity to review the first production before proposing additional searches. When an opponent knows they'll get a second dip at the well, they don't have to make Draconian demands.

5. Incorporate Misspellings, Variants and Synonyms

Did you know Google got its name because its founders couldn't spell googol? Whether due to typos, transposition, IM-speak, misuse of homophones or ignorance, electronically stored information fairly crawls with misspellings that complicate keyword search. Merely searching for "management" will miss "managment" and "mangement."

To address this, you must either include common variants and errors in your list of keywords or employ a search tool that supports fuzzy searching. The former tends to be more efficient because fuzzy searching (also called *approximate string matching*) mechanically varies letters, often producing an unacceptably high level of false hits.

How do you convert keywords to their most common misspellings and variants? A linguist could help or you can turn to the web. Until a tool emerges that lists common variants and predicts the likelihood of false hits, try a site like **http://www.dumbtionary.com** that checks keywords against over 10,000 common misspellings and consult Wikipedia's list of more than 4,000 common misspellings (Wikipedia shortcut: **WP:LCM**).

To identify synonyms, pretend you are playing the board game Taboo. Searches for “car” or” automobile” will miss documents about someone’s “wheels” or “ride.” Consult the thesaurus for likely alternatives for critical keywords, but don’t go hog wild with Dr. Roget’s list. Question key players about internal use of alternate terms, abbreviations or slang

6. Filter and Deduplicate First

Always filter out irrelevant file types and locations before initiating search. Music and images are unlikely to hold responsive text, yet they’ll generate vast numbers of false hits because their content is stored as alphanumeric characters. The same issue arises when search tools fail to decode e-mail attachments before search. Here again, you have to know *how* your search tool handles encoded, embedded, multibyte and compressed content.

Filtering irrelevant file types can be accomplished various ways, including culling by binary signatures, file extensions, paths, dates or sizes and by de-NISTing for known hash values. The National Institute of Standards and Technology maintains a registry of hash values for commercial software and operating system files that can be used to reliably exclude known, benign files from e-discovery collections prior to search. **http://www.nsrl.nist.gov**.

The exponential growth in the volume of ESI doesn’t represent a leap in productivity so much as an explosion in duplication and distribution. Much of the data we encounter are the *same* documents, messages and attachments replicated across multiple backup intervals, devices and custodians. Accordingly, the efficiency of search is greatly aided—and the cost greatly reduced—by *deduplicating* repetitious content *before* indexing data for search or running keywords. Employ a method of deduplication that tracks the origins of suppressed iterations so that repopulation can be accomplished on a per custodian basis.

Applied sparingly and with care, you may even be able to use keywords to <u>exclude</u> irrelevant ESI. For example, the presence of keywords “Cialis” or “baby shower” in an e-mail may reliably signal the message isn’t responsive; but *testing and sampling must be used to validate such exclusionary searches*.

7. Test, Test, Test!

The single most important step you can take to assess keywords is to test search terms against representative data from the universe of machines and data under scrutiny. No matter how well you think you know the data or have refined your searches, testing will open your eyes to the unforeseen and likely save a lot of wasted time and money.

The nature and sample size of representative data will vary with each case. The goal in selection isn’t to reflect the average employee’s collection but to fairly mirror the collections of employees likely to hold responsive evidence. Don’t select a custodian in marketing if the key players are in engineering.

Often, the optimum custodial choices will be obvious, especially when their roles made them a nexus for relevant communications. Custodians prone to retention of ESI are better candidates than those priding themselves on empty inboxes. The goal is to flush out problems *before* deploying searches across broader collections, so opting for uncomplicated samples lessens the value.

It’s amazing how many false hits turn up in application help files and system logs; so early on, I like to test for noisy keywords by running searches against data having nothing whatsoever to do with the case or the parties (e.g., the contents of a new computer). Being able to show a large number of hits in wholly irrelevant collections is compelling justification for limiting or eliminating unsuitable keywords.

Similarly, test search terms against data samples collected from employees or business units having nothing to do with the subject events to determine whether search terms are too generic.

8. Review the Hits

My practice when testing keywords is to generate spreadsheet-style views letting me preview search hits in context, that is, flanked by 20 to 30 words on each side of the hit. It’s efficient and illuminating to scan a column of hits, pinpoint searches gone awry and select particular documents for further scrutiny. Not all search tools support this ability, so check with your service provider to see what options they offer.

Armed with the results of your test runs, determine whether the keywords employed are hitting on a reasonably high incidence of potentially responsive documents. If not, what usages are throwing the search off? What file types are appearing on exceptions lists as unsearchable due to, e.g., obscure encoding, password protection or encryption?

As responsive documents are identified, review them for additional keywords, acronyms and misspellings. Are terms that should be finding known responsive documents failing to achieve hits? Are there any consistent features in the documents with noise hits that would allow them to be excluded by modifying the query?

Effective search is an *iterative* process, and success depends on new insight from each pass. So expect to spend considerable time assessing the results of your sample search. It’s time wisely invested.

9. Tweak the Queries and Retest

As you review the sample searches, look for ways you can tweak the queries to achieve better precision without adversely affecting recall. Do keyword pairs tend to cluster in responsive documents such that using a Boolean *and* connector will reduce noise hits? Can you approximate the precise context you seek by controlling for proximity between terms?

If very short (e.g., three letter) acronyms or words are generating too many noise hits, you may improve performance by controlling for case (e.g., all caps) or searching for discrete occurrences (i.e., the term is flanked only by spaces or punctuation).

10. Check the Discards

Keyword search must be judged both by what it *finds* and what it *misses*. That's the "quality assurance" courts demand. A defensible search protocol includes limited examination of the items <u>not</u> generating hits to assess whether relevant documents are being passed over.

Examination of the discards will be more exacting for your representative sample searches as you seek to refine and gain confidence in your queries. Thereafter, random sampling should suffice.

No court has proposed a benchmark or rule-of-thumb for random sampling, but there's more science to sampling than simply checking every hundredth document. If your budget doesn't allow for expert statistical advice, and you can't reach a consensus with the other side, be prepared to articulate why your sampling method was chosen and why it strikes a fair balance between quality assurance and economy. The sampling method you employ needn't be foolproof, but it must be rational.

Remember that the purpose of sampling the discards is to promptly *identify and resolve* ineffective searches. If quality assurance examinations reveal that responsive documents are turning up in the discards, those failures must receive prompt attention.

Search Tips

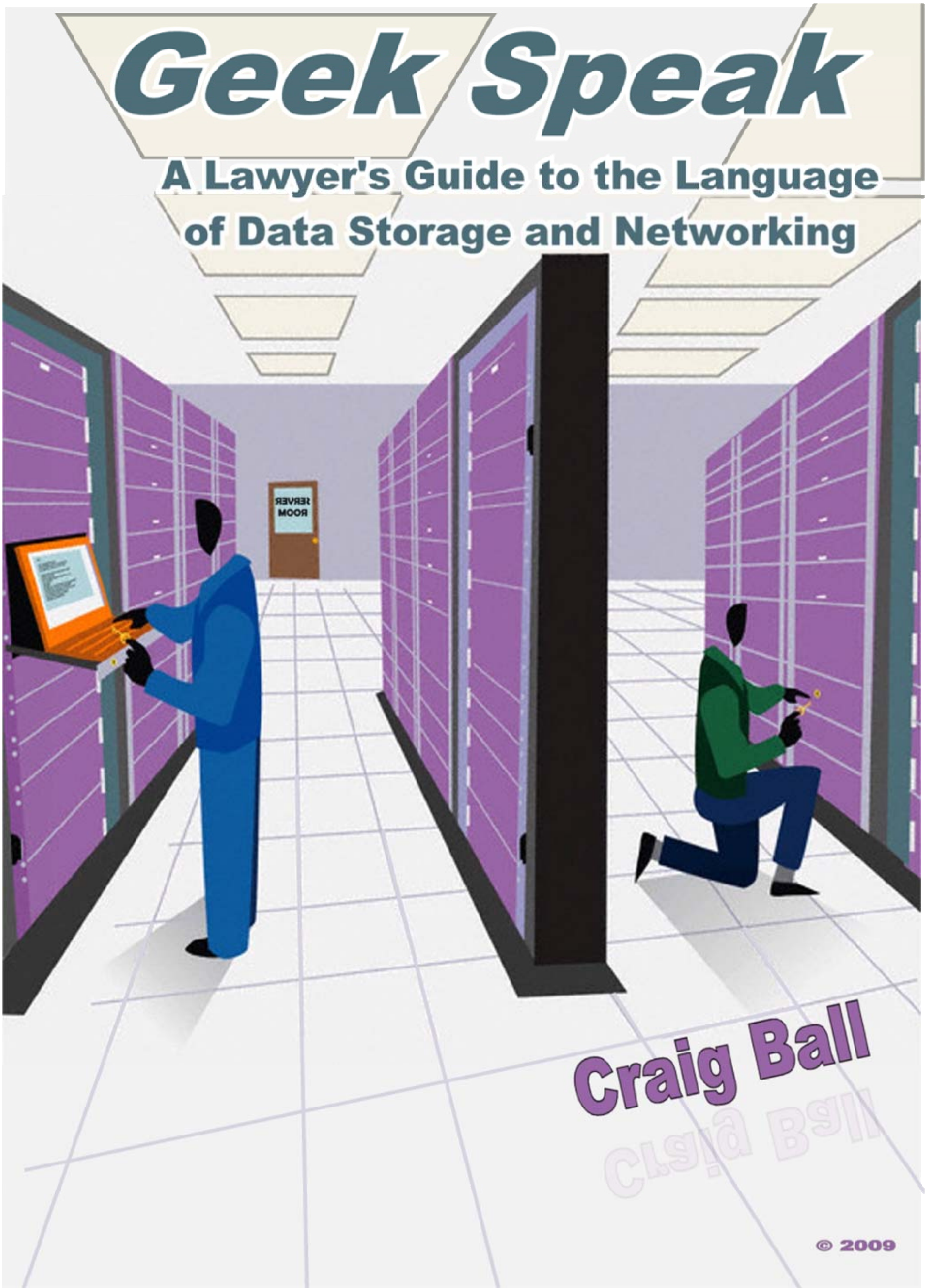
Defensible search strategies are well-documented. Record your efforts in composing, testing and tweaking search terms and the reasons for your choices along the way. Spreadsheets are handy for tracking the evolution of your queries as you add, cut, test and modify them.

Effective searches are tailored to the data under scrutiny. For example, it's silly to run a custodian's name or e-mail address against his or her own e-mail, but sensible for other collections. It's often smart to *tier* your ESI and employ keywords suited to each tier or, when feasible, to limit searches to just those file types or segments of documents (i.e., message body and subject) likely to be responsive. This requires understanding what you're searching and how it's structured.

When searching e-mail for recipients, it's almost always better to search by e-mail address than by name. In a company with dozens of Bob Browns, each must have a unique e-mail address. Be sure to check whether users employ e-mail aliasing (assigning idiosyncratic "nicknames" to addressees) or distribution lists, as these can thwart search by e-mail address or name.

Search is a Science…

…but one lawyers *can* master. I guarantee these steps will wring more quality and trim the fat from text retrieval. *It's worth the trouble*, because the lowest cost e-discovery effort is the one done right from the start.



Geek Speak

A Lawyer's Guide to the Language of Data Storage and Networking

Craig Ball

© 2009

Geek Speak
A Lawyer's Guide to the Language of Data Storage and Networking

Craig Ball

In 1624, when John Donne mused, "No man is an island," he could scarcely have imagined how connected we've become. The bell not only tolls for thee, it beeps and vibrates, too. No iPhone is an iLand.

Networks are the ties that bind our global village and make the world flat. Without networks, our laptops, iPods and Blackberries are just pricey pocket calculators. Networks also transit and store much of the electronic evidence sought in electronic discovery. This article looks at network architecture and data storage devices in the form of an occasionally irreverent glossary offered to help lawyers be at ease discussing the technology of electronic discovery.[1]

Dealing with electronically stored information (ESI) is like living with a teenager—always running in, changing its clothes and heading out again, tracking metadata all over the carpet! But litigants and lawyers aren't relieved of the duty to find and collect potentially relevant ESI just because it's flitting about and messy. They're still obliged to track down the data and make sure it's safe from harm and will stay put (or come home) until needed in discovery. Rooting out responsive data begins with knowing where to look and the right questions to ask, so it helps to have a working knowledge of the terminology of data storage and networking.

Storage and Network and Memory, Oh My!

Though the terms "storage" and "network" are surely familiar, the technologies they describe take many forms, prompting some confusion. Many mistakenly refer to data *storage* devices like hard drives as "memory." Hard drives are *storage*; that is, any non-volatile and semi-permanent electronic, optical, mechanical or magnetic device into which data can be entered and subsequently retrieved on demand. Storage is also a location on a network that enables access to storage devices. *Memory* is a term that should be reserved to devices, particularly *Random Access Memory* or *RAM*, where data resides temporarily during processing but is typically lost or overwritten when an application closes or power is interrupted.[2]

[1] For a more comprehensive (and sober) glossary of e-discovery terms, download The Sedona Conference Glossary for E-Discovery and Digital Information Management (2nd Ed.) from http://www.thesedonaconference.org/dltForm?did=TSCGlossary_12_07.pdf

[2] The line between storage and memory is getting harder to find. Non-volatile flash *memory* is widely used as a means of data *storage* in cameras, thumb drives and solid state drives. Flash memory has almost entirely supplanted photographic film, and solid state drives will soon replace hard drives in laptops and MP3 player. Moreover, it's unclear how *long* information must be "stored" to be called electronically *stored* information. One

A “network” can be any number of computers or devices connected for the purpose of sharing information or capabilities. The largest and most widely used network is, of course, the Internet; but, businesses and homes deploy Wide or Local Area Networks (WANs or LANs) to share databases, mail systems, applications, printers and Internet service. There can be a lot of overlap. WANs may be composed of multiple LANs and connect to the Internet.

B

Backup

Although sharing information and resources is the raison d’être for networking generally, an imperative for business networking is the ability to backup many user’s data from a single location. Without networking and the mapping of users’ storage areas to networked storage devices, users must periodically backup their own data—a responsibility consuming many hours and fostering tragic outcomes.

With networking, each user can be allotted space on a common storage server and the network configured to route that user’s activities to the assigned storage location when the user logs on. The user’s machine may be configured to assign a specified drive letter (e.g., M:) or folder name to the user’s networked storage location. Because the network storage *device* is shared among many users, its allotments are called ***network shares***. But these user-assigned storage areas are typically not “shared” with (i.e., *accessible* to) multiple users. Still other allocations may be open to all or just particular users granted access privileges.

With many users’ critical data consolidated in a single locale, albeit in discrete “shares,” it falls to the ***information technology (IT)*** staff to insure that all that data gets thoroughly and reliably duplicated at regular intervals to protect against its loss as a consequence of system failure or other disaster. Ideally, the duplicate data is physically or electronically transported to a distant secure location unlikely to be affected by the disaster and is then used to get the downed machines back up again; hence the duplicates are called ***backups*** and their use is termed ***disaster recovery***.

Because it’s cheap, durable and portable, magnetic tape is the most common medium used for backup, although remote duplication (***mirroring***) to other network storage devices is fast becoming a viable alternative as hard drive costs plummet. To save time and space, backup regimens seldom copy commercial software programs that can be reinstalled from other media. More time and space is saved—along with network bandwidth--by only occasionally making ***full backups*** of all user created data, opting instead to create more frequent ***differential backups***

case has lawyers worried that the interval may be measured in mere nanoseconds. *Columbia Pictures, Inc. v. Bunnell*, 245 F.R.D. 443 (C.D. Cal. 2007) (defendants ordered to produce contents of RAM).

of files created or changed since the last full backup and ***incremental backups*** of just what's been created or changed since the last incremental backup. When disaster strikes, the full, differential and/or incremental sets are pieced together like Humpty-Dumpty, a process called ***tape restoration***.

Businesses only need disaster recovery data for a brief interval because no business wants to restore its systems with stale data. Accordingly, the only backup tapes essential for recovery are the last complete, uncorrupted set before the river rose. As a cost savings practice, older tapes may be reused by overwriting them with the latest data, a practice called ***tape rotation***.

In practice, companies may keep backup tapes well beyond their utility for disaster recovery--often years longer and occasionally past the companies' ability to access tapes created with obsolete software or hardware. These ***legacy tapes*** are business records—sometimes the last surviving copy—but afforded little in the way of records management. Even businesses that overwrite tapes every two weeks replace their tape sets from time to time as faster, bigger options hit the market. Consequently, old tapes get set aside and forgotten in offsite storage or a box in the corner until their existence is uncovered in discovery.

Backup tapes store data in significantly different ways than the computer systems they protect. Further, large complex enterprises demand large, complex backup systems protecting hundreds of servers. Such backup systems may occupy room-sized silos where robotic arms ceaselessly cycle through thousands of tapes, and databases are required just to track their convoluted contents. This is an arena where broad brush e-discovery efforts go horribly awry and where transparency, close analysis and well-honed choices are vital. Cooperation between opposing sides is essential, and Judges should tread carefully before issuing orders with untoward costs and consequences.[3]

C

Cache

Downloading data over a network is slower than accessing data on a local hard drive, so networked computers sometimes store or "cache" data obtained from the network to avoid the need to download the same data when later needed. Used as a noun, a cache is an area where oft-used information is stored to facilitate its faster access. Devices like hard drives and processors use caching to improve performance, as do certain software programs. For example, Windows computers running the Internet Explorer web browser use a file cache on the local hard drive called Temporary Internet Files which (with some exceptions) holds the HTML code and images of each web page viewed on the machine until the cache is full or emptied by the user. Users revisiting a cached website experience faster page loads because

[3] Judges and counsel may find value in Ball, *What Judges Should Know about Discovery from Backup Tapes* (2008); Available at http://www.craigball.com/What_Judges_Backup_Tapes-200806.pdf

the browser can pull identical data from the cache instead of downloading it from the Web. Though this requires the system to compare the network and cached data to determine if the network data has changed, caching is still faster than needlessly downloading the data a second time.

From the standpoint of electronic discovery, information in the Temporary Internet Files cache may be relevant, especially where Internet usage is at issue or where data (like web mail) may not be available from more accessible locations.

Client

A client, as in ***client-server model***, is a program, computer or other device that connects via a network to another computer or device called the ***server***. Internet browsers are client applications that obtain web pages from web servers. Microsoft Outlook is an e-mail client that connects to e-mail servers like Microsoft's Exchange server. When the client is a personal computer and performs much of the processing of the data, it's ungraciously called a ***fat client***. When the client device or application cedes most processing to the server, it's called a ***thin client*** (or even a ***dumb terminal*** when it has no processing or local storage capabilities at all).

Cloud Computing

Cloud Computing refers to reliance on web-based tools and resources to supplant local applications and storage. It encompasses ***Software as a Service*** (***SaaS***), where users "lease" programs via the Internet (Google Apps is a prime example), as well as the much-touted, yet elusive ***Web 2.0--***a catchall for all manner of web-enabled phenomena: ***social networking, blogs, wikis, Twitter, YouTube, Google mashups*** and arguably any web-centric venture that survived the great dot-com meltdown.

Gen Xers and Millennials embrace "cloud computing" as if they invented it, but Boomers knew cloud computing when it was called client-server or thin client. Then as now, it was screens and keyboards talking to Big Iron elsewhere, the latter doing the heavy lifting. With SaaS and Web 2.0, we've come full circle and are richer for the journey. As cloud computing takes hold, the bits and bytes of our lives will again move out and get their own places, this time in the ether, but we'll have their cell numbers and can call when we need them.

Cloud computing creates new opportunities in e-discovery because the candid, probative revelations once the exclusive province of e-mail now flood ***MySpace*** and ***Facebook***. But cloud computing creates new challenges for e-discovery because it's harder for employers to isolate and search custodial collections without physical dominion of the storage devices and their users' log in credentials. Additionally, repatriation of cloud content depends on the compatibility of cloud formats with local storage formats, including the ability to preserve and produce relevant metadata. Consider ***Gmail***. Though it's feasible to download Gmail messages into a local mail client application like Microsoft Outlook using Gmail's POP3 support feature, the functionality, searchability and some associated metadata will vary between cloud and local counterparts.

Collection

As a noun in e-discovery, collection refers to any discrete set of electronically stored information, particularly the set amassed after targeted retrieval and culling efforts have occurred. However, it's not uncommon to hear parties speak of their entire universe of ESI as the "collection." For this reason, it's important to define the parameters of any ESI collection to insure common expectations.

Container Files

Sometimes called ***compound files***, container files hold other files, often in compressed, encrypted or proprietary formats or nested—container-within-container**--**like Russian matryoshka dolls. Container files commonly encountered in e-discovery include compressed Zip and RAR archives, Outlook PST and OST mail files and Lotus Notes NSF mail files. Container files can severely distort document volume estimations as a function of data volume, e.g., a one gigabyte mail container can easily hold tens of thousands of messages and attachments.

Custodian

A custodian is a caretaker, and in the context of e-discovery, the term refers to a person who holds or is charged with overseeing and maintaining potentially relevant information, whether stored electronically, on paper or by other means. For litigation purposes, one is the custodian of his own e-mail, locally and server-stored documents, voice and electronic messaging, smart phone data and any other information to which he has a right of ownership, access or control, including information in the hands of third parties over whom he may exercise direction or control. Custodian also refers to the persons to whom legal hold notices are directed.

Identifying custodians becomes particularly important when ESI is resides in shared network repositories and no one person bears the duty to preserve, search or produce the data. When *everyone* is responsible, often *no one* steps up. Accordingly, efforts to identify potentially responsive ESI should always inquire into the existence of, or rights of access to, shared repositories.

D

Database

A database is a structured collection of records or information organized according to a framework called a ***data model*** or ***schema*** that typically facilitates search and recall of the records using ***query language***. Massive, costly and enormously complex, databases play vital roles in most large enterprises. For companies like Google, Amazon.com and e-Bay, databases serve as the nexus of virtually all operations. Yet, databases come in all sizes and forms, for tasks as varied as balancing checkbooks, organizing family photos and tracking stock portfolios. Even many common file formats are structured as databases, including Microsoft Outlook mails containers and Adobe Acrobat PDF files.

Databases are the most important resources shared across networks, and they also serve as repositories for much information of importance in e-discovery. Many transactions and documents that would once have been memorialized on paper now exist solely as disparate records stored within databases. Because databases assemble documents on-the-fly and are constantly being updated and purged, they can be particularly challenging sources from which to preserve, isolate and produce responsive data. E-discovery from databases requires detailed assessment of the contents, users, capabilities, applications and schema. Responsive contents may need to be extracted using queries constructed expressly for the purpose of isolating evidence and protecting privileged or confidential content, and the form of production is a key consideration, as many requesting parties lack the hardware and software to assimilate database contents in its native format.

Distributed Data

Distributed data might also be called “willy-nilly data,” in that it describes all the potentially responsive ESI that’s not on the server, but is strewn across laptops, handheld devices, external hard drives, flash drives, CDs, DVDs, home machines, online storage and webmail. Distributed data is costly to collect and sometimes difficult to process because it tends to be the most idiosyncratic ESI and that most prone to obstructive intervention by custodians. A common mistake in e-discovery is assuming that the responsive ESI is on the server without taking reasonable steps to preserve and assess (even by sampling) the contents of distributed data sources.

Domain

A domain is a group of networked computers (typically in the same physical facility) that share common peripherals, directories and storage areas. E-mail systems are customarily organized and backed up by domain.

Domino Server

A Domino server is a network-accessible computer holding users’ centralized e-mail stores and employing the IBM Lotus Notes e-mail application. If an IT person mentions the company’s Domino server (and you aren’t discussing pizza delivery), be prepared for Lotus Notes e-mail and the unique e-discovery challenges and opportunities it entails.

E

ECM

Enterprise Content Management is an umbrella term describing a range of technologies designed to help companies identify, access and use the information stored in their documents, photographs, video, web content, databases and e-mail, especially siloed repositories and unstructured content that tends to be unavailable or difficult to access companywide. ECM applications tend to encompass document management and version control, integration of paper records, records management and retention, web content management and collaboration tools. The most familiar implementation of ECM is probably Microsoft's SharePoint Services (MOSS and WSS).

From an e-discovery perspective, the consequences of a substantial ECM implementation are manifold. ECM may operate at cross-purposes with—or at least complicate--legal hold obligations. Further, collaborative environments are heavily dependent on metadata to support functionality, making preservation and production of a broad range of metadata essential to meet the obligation to produce ESI in reasonably usable forms. Within some ECM environments, documents exist in untraditional and proprietary formats necessitating new and creative approaches to selecting forms of production that preserve look, feel and function of multimedia and informational content. On the positive side, a successful ECM system should facilitate cost-effective identification and search of responsive ESI (though cynics might suggest that savings will be offset by having to deal with all the potentially responsive ESI that ECM makes impossible to ignore).

Enterprise

Enterprise is variously the flagship Federation starship commanded by Captain James T. Kirk, a low cost rental car company favored by skinflint insurance carriers or, in e-discovery, the term of choice when "company" or "business" are insufficiently pretentious.

Ethernet

A set of network cabling and communication protocols for bus topology[4] local area networks. That is, an agreed-upon set of instructions, akin to a language, that permits devices to exchange information. If that's not helpful, think of it as the *other* way computers talk to each other when they're not speaking Internet (TCP/IP).

Exchange Server

An Exchange server is a network accessible computer holding users' centralized e-mail stores and running the Microsoft Exchange e-mail and calendaring application. Typically, users access Exchange servers with Microsoft Outlook mail clients. Microsoft Exchange accounts for

[4] See "Topology," *infra*, for further discussion of network topologies.

65% of market share among all organizations, with significantly larger shares among businesses with fewer than 49 employees and those in the health care and telecommunications sectors. Consequently, Exchange Server e-mail crops up in the overwhelming majority of cases and understanding its architecture is an essential e-discovery skill.[5] ***See also*** the discussion of Microsoft Outlook, ***infra.***

Extensible Markup Language (XML)

Extensible Markup Language or XML provides a basic syntax that can be used to share information between different kinds of computers, applications and organizations without first converting it. XML employs coded identifiers paired with text and other information. These identifiers can define the appearance of content (much like the Reveal Codes screen of WordPerfect documents) or serve to tag content to distinguish whether 09011957 is a birth date (09/01/1957), a phone number (0-901-1957) or a Bates number. Plus, markup languages allow machines to talk to each other in ways humans understand.

Like multilingual speakers agreeing to converse in a common language, as long as two systems employ the same XML tags and structure (typically shared as an XML Schema Definition or .XSD file), they can quickly and intelligibly share information. Parties and vendors exchanging data can fashion a common schema tailored to their data or employ a published schema suited to the task, such as that under development by the Electronic Discovery Reference Model. [6]

Extranet

An extranet is a private network made available via the Internet to a select group of users, typically customers or suppliers. When used to support transactions, extranets are often called ***virtual deal rooms***. Extranets are increasingly used as a collaborative tool in e-discovery and as a host repository for ESI. Access may be secured by use of a VPN connection or by a conventional link employing user ID and password alone.

F

File Server

File servers, the heart of any client-server network, are computers typically equipped with fast, redundant storage devices that store and deliver each user's files and other data. Very small networks may not use dedicated file servers but instead allow workstations to share data amongst themselves in a peer-to-peer configuration.

FTP

File Transfer Protocol or FTP is a set of standards and instructions that permit transfer of files

[5] For a more detailed discussion of Exchange Servers and e-discovery, see Ball, *Meeting the Challenge of E-Mail in Civil Discovery* (2009) at p.25 et seq., *infra* and available at http://www.craigball.com/em2008.pdf

[6] http://edrm.net

between networked computers, most often via the Internet. You'll encounter FTP in e-discovery both as a potential repository to be explored for "orphaned" responsive data not available from other accessible sources and as a mechanism to transfer large volumes of data to and from clients and e-discover service providers.

G

Gateway

A gateway is a combination of hardware and software that allows two networks to communicate. A gateway is essentially a protocol translator that enables, e.g., the wireless network in your home to communicate with the Internet. In this role, the gateway is also called a ***router***.

H

Hub

A hub allows multiple computers to share a network connection, not unlike a power strip allows multiple electrical devices to share AC power from an outlet. Hubs support simple peer-to-peer networking between computers.

I

IM

Instant Messaging or IM is a form of real-time textual communication between two or more persons where such messages are carried by the Internet or a cell phone network. It is the instantaneous receipt and response of IM and its evanescence that distinguishes IM from e-mail. Though relevant, non-privileged IM messages are as subject to preservation and production duties as any other evidence, IM messages typically reside only on the local device sending or receiving the message, not on network servers, and not in active data unless the user has enabled message logging. Accordingly, litigants obliged to preserve IM traffic must either compel message logging and periodic collection of the logs or implement a packet capture mechanism to scan for IM traffic and snare and copy messages as they enter and leave the company's Internet gateway. Neither method is wholly satisfactory.

When a company obliged to preserve IM traffic fails to do so, the data loss may be mitigated by collection from other parties to the dialog or by forensic examination of the machines or devices employed, although recovery of message traffic is by no means assured.

Internet

You're not *really* going to make me define Internet, are you? Where have you been the last 15 years?! Okay, if you insist.

Turning to none other than the august personage of former (convicted but charges dropped) Alaska Senator Ted Stevens in a speech delivered on June 28, 2006 as chairman of the Senate Committee on Commerce, Science and Transportation:

> [T]he Internet is not something that you just dump something on. It's not a big truck. It's a series of tubes. And if you don't understand, those tubes can be filled and if they are filled, when you put your message in, it gets in line and it's going to be delayed by anyone that puts into that tube enormous amounts of material, enormous amounts of material.

So, the Internet is a series of tubes, not a big truck, and it's best to keep a plumber's helper at hand while Web surfing.

Intranet

An intranet is a private web site, typically reserved to the exclusive use of an organization's employees or members. Intranets tend to be hosted internally on a local access network, but may be Internet-enabled so as to permit secure connections by authorized users via the Internet.

IP Address

An Internet Protocol or IP address is a unique series of four numbers joined by periods and sometimes called a Dotted Quad. It is the numerical designation of the host system that connects you to the Internet and is cross-referenced to the domain name such that either the name or the number can be employed to correctly designate your host system. An IP address can also serve as a unique identifier for computers and other Web-enabled devices on a network employing the standard TCP/IP protocol that serves as the basic computer-to-computer language of the Internet. For example, the IP address of the computer used to write this article is 192.168.0.189.

IP addresses can be useful in e-discovery when constructing a company's data map. Using IP addresses, machines claimed to exist can be correlated against those actually connected to a network. An IP address can also tie ESI to a particular device and, thus, a particular user.

ISP

An Internet Service Provider or ISP is a business or other entity that supplies Internet access via dial-up, cable modem, DSL or ISDN lines or dedicated high speed connections. ISPs routinely host their customers' e-mail accounts and thus may be a source of ESI by subpoena or constitute a third party custodian who should be put on notice of legal hold obligations.

J

Journaling

Journaling is a means of archiving electronic messages, principally e-mail, but potentially IM and VM, too. A journaling mail server copies all messages or, per established rules, certain incoming and outgoing messages to a mailbox or storage location serving as the journaling repository. Journaling serves to preempt ultimate reliance on individual users for litigation preservation and regulatory compliance. Properly implemented, it should be entirely transparent to users and secured in a manner that eliminates the ability to alter the journaled collection.

Accordingly, journaling is a valuable safety net for companies obliged to preserve e-mail because of litigation or regulatory obligations, and counsel should inquire to determine if journaling was enabled, as journaled e-mail traffic can mitigate custodial preservation errors and misconduct. Journaling also helps protect the company against rogue employees seeking to conceal wrongdoing by destroying their e-mail stores before leaving.

Exchange Server supports three types of journaling:

- Message-only journaling, which does not account for blind carbon copy recipients, recipients from transport forwarding rules, or recipients from distribution group expansions;
- Bcc journaling, which is identical to Message-only journaling except that it captures Bcc addressee data; and
- Envelope Journaling which captures all data about the message, including information about those who received it.

Envelope journaling is the mechanism best suited to e-discovery preservation and regulatory compliance. Unlike messages preserved after delivery, journaled messages won't include metadata reflecting the addressee's handling of the message, such as foldering or indications that the message was read.

Journaling should be distinguished from e-mail archiving, which may implement only selective, rules-based retention and customarily entails removal of archived items from the server for offline or near-line storage to minimize strain on IT resources and/or implement electronic records management. However, Exchange journaling also has the ability to implement rules-based storage, so each can conceivably be implemented to play the role of the other.

L

LAN

A Local Area Network or LAN is an interconnected group of computers typically situated in a single location and connected by cable or wirelessly. LANs tend to be used in offices and homes to share Internet connections, files and printers, though they may also be configured to exchange e-mail internally.

Lotus Notes

Lotus Notes is an IBM client application supporting e-mail, calendaring, web browsing and a host of collaborative features. Notes works in conjunction with an IBM Lotus Domino server, although it can also be configured to retrieve e-mail from Microsoft Exchange servers. Though Lotus Notes reportedly has just a 10% overall market share, it enjoys a much higher percentage base among manufacturers with at least 5,000 employees, and IBM claims it has sold 140 million Notes licenses worldwide. Still, the relative infrequency with which E-discovery service providers encounter Lotus Notes means that not all providers are equipped or experienced to process Notes content.

Unlike Microsoft Exchange, which is a purpose-built application designed for messaging and calendaring, Lotus Notes is more like a toolkit for building whatever capabilities you need to deal with documents—mail documents, calendaring documents and any other type of document used in business. Notes wasn't designed for e-mail—e-mail just happened to be one of the things it was tasked to do.

Notes is database-driven and distinguished by its replication and security. Lotus Notes is all about copies. Notes content, stored in ***Notes Storage facility*** or ***NSF*** files, is constantly being replicated (synchronized) here and there across the network. This guards against data loss and enables data access when the network is unavailable, but it also means there can be many versions of Notes data stashed in various places within an enterprise. Thus, discoverable Notes mail may not be gone, but lurks within a laptop that hasn't connected to the network since the last business trip.

M

Mail Client

A mail client is any software application used to prepare, send, receive and read e-mail. E-mail clients can be rudimentary or, more common today, feature-laden productivity tools like Microsoft Outlook or Lotus Notes, which offer a sophisticated and highly-customizable interface. The configuration of a user's mail client may determine whether messages are stored locally, on the mail server or in both places. Additionally, the mail client records and manages key metadata detailing a user's handling of e-mail, including the user's folder structure and various flags indicating whether, *inter alia*, the user opened a particular message, tied it to a calendar entry or flagged it for action.

Microsoft Outlook

Microsoft Outlook is an e-mail client and calendaring tool coupled with several other productivity features to comprise a personal information manager (PIM) toolset. Outlook serves as both a standalone mail client compatible with all mail protocols in common use, but in business, it's usually deployed in conjunction with ***Microsoft Exchange Server*** or, lately, ***Microsoft Office SharePoint Server*** (MOSS).

Despite the confusing similarity of their names, Outlook is a much different and substantially more sophisticated application than Outlook Express (now called Windows Mail). One of many important differences is that where Outlook Express stores messages in plain text, Outlook encrypts and compresses messages. The most significant challenge Outlook poses in discovery is the fact that all of its message data and folder structure, along with all other information managed by the program (except the user's Contact data), is stored within a single, often massive, database file with the file extension .pst. The Outlook PST file format is proprietary and its structure is poorly documented, limiting your options when trying to view or process its contents to Outlook itself or one of a handful of PST file reader programs available for purchase and download via the Internet.

While awareness of the Outlook PST file has grown, even many lawyers steeped in e-discovery fail to consider a user's Outlook .ost file. The OST or offline synchronization file is commonly encountered on laptops configured for Exchange Server environments. Designed to afford access to cached messages when the user has no active network connection., e.g., while on airplanes, local OST files often hold messages purged from the server—at least until re-synchronization. It's not unusual for an OST file to hold e-mail unavailable from any other comparably-accessible source.

By default, when a user opens an attachment to a message from within Outlook (as opposed to saving the attachment to disk and then opening it), Outlook stores a copy of the attachment in a "temporary" folder. But don't be misled by the word "temporary." In fact, the folder isn't going anywhere, and its contents—sometimes voluminous--tend to long outlast the messages that transported the attachments. Thus, litigants should be cautious about representing that Outlook e-mail is "gone" if the attachments are not.

The Outlook "viewed attachment folder" will have a varying name for every user and on every machine, but it will always begin with the letters "OLK" followed by several randomly generated numbers and uppercase letters (e.g., OLK943B, OLK7AE, OLK167, etc.).

Mirroring

Mirroring refers to the creation of an exact copy of a dataset. Mirroring may be used locally for data integrity and protection or across a network as a form of backup, duplicating the entire contents of a server to some distant, identical system. Disk mirroring, also called RAID 1, entails simultaneously writing identical data to two different hard drives, affording redundancy should either drive fail.

N

Nearline Storage

Nearline storage refers to voluminous data that, while not in such demand as to require instantaneous access via the network, must nonetheless be available from time-to-time without human intervention. Nearline data tends to be stored on high capacity media (like magnetic tape) that can be robotically loaded on demand, occasioning only a brief delay between a request and delivery of data.

NAS

Networked Attached Storage or NAS is a dedicated file server designed expressly for data storage. Because a NAS isn't called upon to do general computing tasks, it can employ a file system built exclusively for its limited role. When inquiring about devices, be careful not to reference only computers and servers, as a too-literal interpretation might allow someone to overlook a NAS.

Node

Anything connected to a network can be termed a "node;" however, anyone who uses the word node in this way must be termed a "nerd."

O

Offline Data

Offline data denotes ESI housed on media that is not connected to the network and requires human intervention, e.g., mounting or restoration, to access the contents. Backup tapes sent offsite for storage, legacy systems in the warehouse and even a CD-R in your desk drawer are examples.

The e-discovery challenge of offline data is that it must be proven not reasonably accessible to be excluded from search and production. Even then, producing parties must identify offline data with sufficient specificity to allow the requesting party to determine if the producing party is right about the data's inaccessibility. But there's the catch: how does a producing party do that without examining the contents?

To economically manage offline data, insure that its contents are indexed and the media clearly labeled *when the data goes offline* so as to obviate the costly and time-consuming need to bring it online, albeit briefly, to identify its contents. This isn't going to help with legacy data, but it's a no-brainer going forward.

P

Partition

A partition is a division of the storage area of a hard drive such that a single physical drive can be seen by the computer as multiple drives. If you think of an unpartitioned hard drive as a big metal cabinet, a partition is the division of that cabinet into file drawers. Though it's most common to encounter drives created with a single partition encompassing the entire storage area of the drive, in Windows, a hard drive can currently have up to four primary partitions or three ***primary partitions*** and one so-called ***extended partition*** that can be subdivided into as many as 24 extended partitions. Only one of the four partitions can be designated as an active partition, signaling the partition that holds the operating system the machine should boot on start up.

Partitioned hard drives can hold multiple operating systems such that a snippet of code called a ***boot loader*** can point the system to a partition other than the active partition to initiate a different operating system. Thus, a machine with a single drive can be configured to boot in Windows Vista, Linux or Windows XP via a start up menu. From the standpoint of e-discovery, a thorough search for ESI should include accounting for the full storage capacity of a hard disk, in case responsive data lurks on another partition. If you think this sounds farfetched, take a look at *Phoenix Four, Inc. v. Strategic Res. Corp.*[7]

Path

The complete local or network address to a particular folder, file or device, expressed hierarchically from a root location of a server or disk volume. If I were a file, the path to me might be expressed as ***Earth:\\North America\USA\Texas\Austin\78735\3723 Lost Creek Blvd\Lab\Craig Ball***. Traversing a path to a file is sometimes called "drilling down."

Peer-to-Peer Network

In a ***peer-to-peer*** or ***P2P*** network, each connected computer serves as both client and server for the purpose of sharing resources, but most often for sharing files (notably copyrighted music and video, as well as adult content and pirated software).

Peripheral

Just about any device you connect to a computer by cabling or networking (other than another computer or server) is called a peripheral. It most commonly refers to printers and scanners.

Protocol

An agreed-upon set of instructions, akin to a language, that permit devices to exchange information. Networks notably employ Ethernet or TCP/IP protocols to intelligibly transmit and receive data. As language can be thought of as a "protocol" for written or oral communications, a network protocol is a framework to sensibly interpret the ones and zeroes of digital communications.

[7] No. 05 Civ. 4837, 2006 WL 1409413 (S.D.N.Y. May 23, 2006).

R

RAID

A ***Redundant Array of Independent (or Inexpensive) Disks*** or ***RAID*** is a way of combining multiple hard drives to achieve greater performance, greater reliability or a mix of the two. The various types of RAID configurations are numbered. The three most commonly used configurations are RAID 0, RAID 1 and RAID 5.

A RAID 0 divides (or *stripes*, in storage parlance) data between two hard drives to combine the capacity into a single large volume and to increase the speed at which data is read and written. But because the data zigzags across two drives, a failure of either drive means the loss of all data.

A RAID 1 opts for complete redundancy, mirroring all contents between two drives such that a failure of either drive results in no loss of data--the trade off being that you can use only half of the combined capacity of the two drives and get no performance boost.

A RAID 5 uses three or more disks, garnering some of the speed boost seen in RAID 0 and the ability to fully recover all data should any one drive fail.

Because any one drive in a RAID 5 array can fail without data loss, RAID storage allows for the removal and replacement of drives from the array without the need to down the server. Thus, RAID storage—particularly RAID 5 configurations with more than 3 disks—are ubiquitous in mission critical servers. RAID 5 arrays are typically seen by the server as a single logical disk with a capacity of about two-thirds of the combined capacity of all disks in the array.

Despite its reliability, a RAID is not a substitute for a backup. A fire, flood or disgruntled employee won't destroy just one or two drives in the array, and all data will be unrecoverable absent a backup.

Root

Root refers to top level of a file system's directory structure, typically C:\ in a Windows system. In hacking, it also refers to a level of unrestricted access to a system, where "getting root" means taking unauthorized control of the system, often using hacker tools called ***root kits***.

Router

A router (sometimes called a ***switch***) is a device that directs the flow of the data packets by which information is transferred across a network. Unlike a hub, which merely relays all packets to all connections, a router actually assigns unique addresses to connections and steers packets to and from those addresses.

S

SaaS

Software as a Service or ***SaaS*** is software distribution mechanism where, instead of purchasing applications and installing them, programs are accessed on the Internet or downloaded on-the-fly as needed. The advantage of SaaS is that there is no need to purchase upgrades or install patches because the software's always up-to-date. The down side is that you do not own the software and must continue to pay for its use, as well as security concerns. In e-discovery, complications derive from the loss of physical dominion of the devices storing the data, as discussed previously under Cloud Computing. A notable example of SaaS is the Google Apps package of applications, which virtualizes a user's e-mail, contacts and calendar, along with document, spreadsheet and presentation authoring tools. The provider of SaaS is called an ***Application Service Provider*** or ***ASP***.

SAN

A ***Storage Area Network*** or ***SAN*** is a mass storage configuration that allows *network*-attached devices to be shared among servers at very high speeds yet appear as if they are *physically* attached to each server. SANs are tied to two important trends in networking: ***storage replication*** (where data is remotely mirrored for disaster recovery) and ***virtualization*** (where physical devices are subdivided into multiple virtual devices that appear to be distinct, physical machines like servers but actually exist as emulations using software). SANs allow large aggregations of physical storages devices to be logically re-allocated to various servers and tasks. Instead of adding a 120GB hard drive to a server, a 120GB "slice" of a multi-terabyte array can be assigned to appear and function as a physically-connected 120GB drive.

Server

A server is a device or application that delivers information to networked devices. When applied to hardware, server usually denotes a computer optimized and tasked to perform certain functions for other machines on the network. Servers tend to be isolated in locked and refrigerated server rooms, protected by backup systems and equipped with fail-safe or redundant components mounted in accessible racks, all to minimize downtime and increase security. Though a single server can perform a variety of tasks, businesses tend to dedicate servers to particular functions, such as storing user data, running applications like databases, delivering web content, managing printing, routing Internet traffic, handling e-mail stores, etc.

Share

Also called a ***Network Share***, see the discussion of shares in **Backup,** above.

Single Instance Storage

Networks and e-mail systems are replete with multiple iterations of identical documents. When an entire department receives an e-mail with the same attachment, or when thousands of employees keep a copy of the same memo, storage is wasted. Single instance storage performs ***de-duplication*** and replaces the individual copies with a *pointer* to an identical master copy. SIS aids backup by facilitating the use of fewer tapes and reducing the time required to complete the task. When dealing with a SIS volume in e-discovery, be careful to collect the de-duplicated document and not just its SIS pointer.

T

TCP/IP

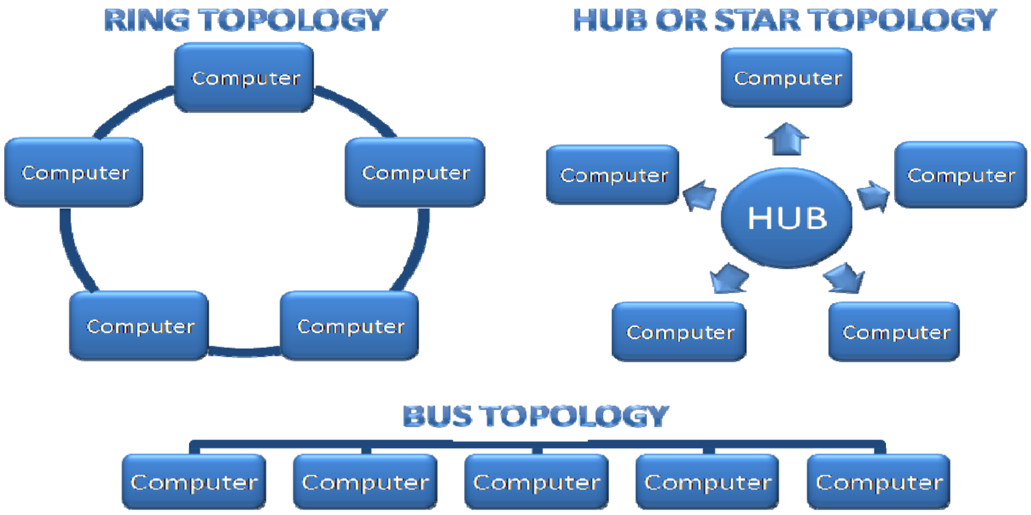
Transmission Control Protocol/Internet Protocol or ***TCP/IP*** is the universal computer-to-computer language of the Internet, but can also be implemented to support an intranet.

Thin Client

See *Client*

Topology

A geometric description of a network's structure based upon the way devices interconnect. Compare communication routes of the Ring, Hub or Star and Bus topologies depicted below.



V

Virtual Machine

Virtual machine or ***VM*** refers to the use of software to emulate or mimic the presence and function of hardware. Using VM software, a complete hardware and software computing environment, including operating systems, applications, data and emulated peripherals, can be stored in a single file. When that file is loaded to a VM player, it looks and works just like a real machine, but runs in a window, like any other piece of software.

Virtual machines have found enthusiastic acceptance in the IT world as a means to deploy, protect and backup virtualized servers, as well as a method to extract more value from hardware because one "real" machine can run many virtual machines without a notable drop in performance.

Because VMs can replicate almost any computing platform or environment, it promises to be a viable form of production for complex ESI. Virtualization enables opposing sides to enjoy comparable levels of functionality in native production even when one side lacks the hardware and software resources of the other. Not only does the evidence look the same for both sides, but it *works* the same way and can be easily shielded from inadvertent alteration and intentional manipulation.

Volume

A volume is a logical division of a hard drive that can hold a single operating system. Where a partition was akin to the physical drawer in a file cabinet, a volume speaks to the division of that drawer into compartments to hold file systems and files.

VPN

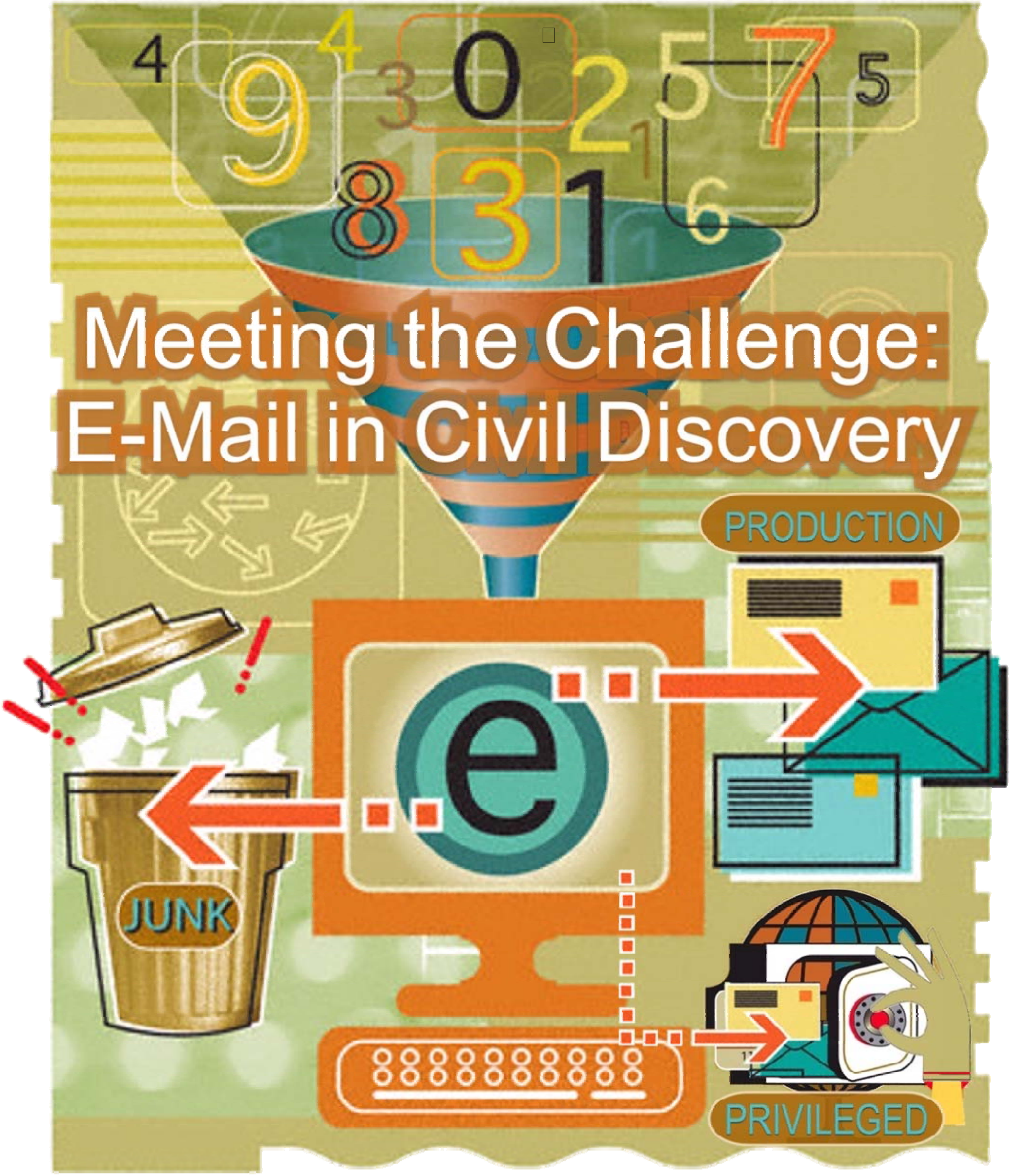
A ***Virtual Private Network*** or ***VPN*** is a private (i.e., secure) network that employs public pathways (i.e., the Internet). By employing authentication protocols and encryption of data as it traverses public pathways, the network traffic over a VPN is protected from interception and thus said to "tunnel" through public areas.

W

Workgroup

A workgroup is a subset of users in a local area network environment who are assigned privileges enabling them to collaborate by sharing files and peripherals. Microsoft Windows uses the term workgroup to identify the participants in a peer-to-peer network.

Meeting the Challenge: E-Mail in Civil Discovery



Craig Ball

Meeting the Challenge: E-Mail in Civil Discovery
Craig Ball

Table of Contents

Introduction

This paper looks at e-mail from the standpoint of what lawyers should know about the nuts-and-bolts of these all-important communications systems. It's technical; sometimes, *very* technical.

When you finish the paper, you'll know *a lot* more about e-mail, and along the way, you may realize that discoverable e-mail can be found in far more places than your client probably checked before the last time you said, "Yes, your Honor, we've given them the e-mail."

So, if you know what's good for you, you should probably stop reading right now.
….

Still here? Okay, you asked for it.

Get the e-mail! It's the war cry in discovery today. More than simply a feeding frenzy, it's an inevitable recognition of e-mail's importance and ubiquity. We go after e-mail because it accounts for the majority of business communications and because e-mail users tend to let their guard down and reveal plainspoken truths they'd never dare put in a memo. Or do they? A 2008 study[29] demonstrated that employees are significantly more likely to lie in e-mail messages than in traditional pen-and-paper communications. Whether replete with ugly truths or ugly lies, e-mail is telling and compelling evidence.

If you're on the producing end of a discovery request, you not only worry about what the messages say, but also whether you and your client can find, preserve and produce all responsive items. Questions like these *should* keep you up nights:

- Will the client simply conceal damning messages, leaving counsel at the mercy of an angry judge or disciplinary board?
- Will employees seek to rewrite history by deleting "their" e-mail from company systems?
- Will the searches employed prove reliable and be directed to the right digital venues?

[29] http://www3.lehigh.edu/News/V2news_story.asp?iNewsID=2892 (visited 11/1/08)

- Will review processes unwittingly betray privileged or confidential communications?

Meeting these challenge begins with understanding e-mail technology well enough to formulate a sound, defensible strategy. For requesting parties, it means grasping the technology well enough to assess the completeness and effectiveness of your opponent's e-discovery efforts.

This paper seeks to equip the corporate counsel or trial lawyer with some of what's needed to meet the challenge of e-mail discovery in civil litigation. It's intended to be technical because technical knowledge is what's most needed and most lacking in continuing legal education today. Even if you went to law school because you had no affinity for matters technical, it's time to dig in and learn enough to stay in the fray.

Not Enough Eyeballs

Futurist Arthur C. Clarke said, "Any sufficiently advanced technology is indistinguishable from magic." E-mail, like electricity or refrigeration, is one of those magical technologies we use every day without knowing quite how it works. But, "It's magic to me, your Honor," won't help you when the e-mail pulls a disappearing act. Judges expect you to pull that e-mail rabbit out of your hat.

A lawyer managing electronic discovery is obliged to do more than just tell their clients to "produce the e-mail." You've got to make an effort to understand their systems and procedures and ask the right questions. Plus, you have to know when you aren't getting the right answers. Perhaps that's asking a lot, but well over 95% of all business documents are born digitally and only a tiny fraction are ever printed.[30] Hundreds of billions of e-mails traverse the Internet *daily,* far more than telephone and postal traffic combined,[31] and the average business person sends and receives between 50 and 150 e-mails *every business day.* E-mail contributes *500 times greater volume* to the Internet than web page content.

Think that's a lot? Then best not think about the fact that the volume is expected to nearly double by 2012,[32] and none of these numbers take into account the explosive growth in instant messaging, unified messaging or the next insanely great communication or collaboration technology that—starting next year and every year—we can hardly live without. The volume keeps increasing, and there's no end in sight. It's simply too easy, too quick and too cheap to expect anything else.

Neither should we anticipate a significant decline in users' propensity to retain their e-mail. Here again, it's too easy and, at first blush, too cheap to expect users to selectively dispose of e-mail

[30] Extrapolating from a 2003 updated study compiled by faculty and students at the School of Information Management and Systems at the University of California at Berkeley. http://www2.sims.berkeley.edu/research/projects/how-much-info-2003/

[31] http://www.radicati.com/?p=638 (visited 11/1/08)

[32] Id.

and still meet business, litigation hold and regulatory obligations. Our e-mail is so twisted up with our lives that to abandon it is to part with our personal history.

Another difficulty is that this startling growth isn't happening in just one locale. E-mail lodges on servers, cell phones, laptops, home systems, thumb drives and in "the cloud," a term ethereally denoting all the places we store information online, little knowing or caring about its physical location. Within the systems, applications and devices we use to store and access e-mail, most users and even many IT professionals don't know where messages lodge or how long they hang around.

In discovery, we overlook so much that we're obliged to consider, and with respect to what we do collect, it's increasingly infeasible to put enough pairs of trained eyes in front of enough computers to review every potentially responsive electronic document. Instead, we must employ shortcuts that serve as proxies for lawyer judgment. Here, too, our success hinges upon our understanding of the technologies we use to extend and defend our reach.

Test Your E.Q.

Suppose opposing counsel serves a preservation demand or secures an order compelling your client to preserve electronic messaging. Are you assured that your client can and will faithfully back up and preserve responsive data? Even if it's practicable to capture and set aside the current server e-mail stores of key custodians—and even if you hold onto backup tapes for a few significant points in time—are you *really* capturing all or even most of the discoverable communications? How much is falling outside your net, and how do you assess its importance?

Here are a dozen questions you should be able to confidently answer about your client's communication systems:

1. What messaging environment(s) does your client employ? Microsoft Exchange, Lotus Domino, Novell GroupWise or something else?
2. Do *all* discoverable electronic communications come in and leave via the company's e-mail server?
3. Is the e-mail system configured to support synchronization with local e-mail stores on laptops and desktops?
4. How long have the current e-mail client and server applications been used?
5. What are the message purge, dumpster, journaling and archival settings for each key custodian?
6. Can your client disable a specific custodian's ability to delete messages?
7. Does your client's backup or archival system capture e-mail stored on individual user's hard drives, including company-owned laptops?
8. Where are e-mail container files stored on laptops and desktops?
9. How should your client collect and preserve relevant web mail?

10. Do your clients' employees use home machines, personal e-mail addresses or browser-based e-mail services (like Gmail or Yahoo! Mail) for discoverable business communications?
11. Do your clients' employees use Instant Messaging on company computers or over company-owned networks?
12. How do your clients' voice messaging systems store messages, and how long are they retained?

If you are troubled that you can't answer some of these questions, you should be; but know you're not alone. Many other lawyers can't either. And don't delude yourself that these are exclusively someone else's issues, *e.g.,* your litigation support services vendor or IT expert. These are the inquiries that will soon be coming at *you* in court and when conferring with the other side. You do confer on ESI, right?

Staying Out of Trouble

Fortunately, the rules of discovery don't require you to do the impossible. All they require is diligence, reasonableness and good faith. To that end, you must be able to establish that you and your client acted swiftly, followed a sound plan, and took such action as reasonable minds would judge adequate to the task. It's also important to keep the lines of communication open with the opposing party and the court, seeking agreement with the former or the protection of the latter where fruitful. I'm fond of quoting Oliver Wendell Holmes' homily, "Even a dog knows the difference between being stumbled over and being kicked." Judges, too, have a keen ability to distinguish error from arrogance. There's no traction for sanctions when it is clear that the failure to produce electronic evidence occurred despite good faith and due diligence.

…And You Could Make Spitballs with It, Too

Paper discovery enjoyed a self-limiting aspect because businesses tended to allocate paper records into files, folders and cabinets according to persons, topics, transactions or periods of time. The space occupied by paper and the high cost to create, manage and store paper records served as a constant impetus to cull and discard them, or even to avoid creating them in the first place. By contrast, the ephemeral character of electronic communications, the ease of and perceived lack of cost to create, duplicate and distribute them and the very low direct cost of data storage have facilitated a staggering and unprecedented growth in the creation and retention of electronic evidence. At fifty e-mails per day, a company employing 100,000 people could find itself storing well over *1.5 billion* e-mails annually.

Did You Say *Billion*?

But volume is only part of the challenge. Unlike paper records, e-mail tends to be stored in massive data blobs. The single file containing my Outlook e-mail is over four gigabytes in size and contains tens of thousands of messages, many with multiple attachments covering virtually every aspect of my life and many other people's lives, too. In thousands of those e-mails, the subject line bears only a passing connection to the contents as "Reply to" threads strayed further and further from the original topic. E-mails meander through disparate topics or, by absent-minded clicks of the “Forward” button, lodge in my inbox dragging with them, like toilet paper on a wet shoe, the unsolicited detritus of other people’s business.

To respond to a discovery request for e-mail on a particular topic, I’d either need to skim/read countless messages or I’d have to naively rely on keyword search to flush out all responsive material. If the request for production implicated material I no longer kept on my current computer or web mail collections, I’d be forced to root around through a motley array of archival folders, old systems, obsolete disks, outgrown hard drives, ancient backup tapes (for which I currently have no tape reader) and unlabeled CDs. Ugh!

Net Full of Holes

I’m just one guy. What’s a company to do when served with a request for “all e-mail” on a particular matter in litigation? Surely, I mused, someone must have found a better solution than repeating, over and over again, the tedious and time-consuming process of accessing individual e-mail servers at far-flung locations along with the local drives of all key players’ computers?

For this article, I contacted colleagues in both large and small electronic discovery consulting groups, inquiring about “the better way” for enterprises, and was struck by the revelation that, if there was a better mousetrap, they hadn’t discovered it either. Uniformly, we recognized such enterprise-wide efforts were gargantuan undertakings fraught with uncertainty and concluded that counsel must somehow seek to narrow the scope of the inquiry—either by data sampling or through limiting discovery according to offices, regions, time span, business sectors or key players. Trying to capture *everything,* enterprise-wide, is trawling with a net full of holes.

New Tools

The market has responded in recent years with tools that either facilitate search of remote e-mail stores, including locally stored messages, from a central location (*i.e.,* enterprise search) or which agglomerate enterprise-wide collections of e-mail into a single, searchable repository (*i.e.,* e-mail archiving), often reducing the volume of stored data by so-called “single instance de-duplication,” rules-based journaling and other customizable features.

These tools, especially enterprise archival, promise to make it easier, cheaper and faster to search and collect responsive e-mail, but they’re costly and complex to implement. Neither established standards nor a leading product has emerged. Further, it remains to be seen whether the practical result of a serial litigant employing an e-mail archival system is that they—for all intents and purposes--end up keeping every message for every employee.

E-Mail Systems and Files

The corporate and government e-mail environment is dominated by two well-known, competitive product pairs: Microsoft Exchange Server and its Outlook e-mail client and IBM Lotus Domino server and its Lotus Notes client. A legacy environment called Novell GroupWise occupies a distant third place, largely among government users.

Per a 2008 study by Ferris Research,[33] Microsoft Exchange accounts for 65% of market share among all organizations, with significantly larger shares among businesses with fewer than 49 employees and those in the health care and telecommunications sectors. Lotus Notes was found to have just 10% of overall market share, but a much higher percentage base among manufacturers with at least 5,000 employees. GroupWise's share was termed "negligible," except in niches—notably organizations in the financial services and government sectors with 100 to 999 employees—where its share reached as high as 10-15%. Blackberry servers transmit a large percentage of e-mail as well, but these messages typically find their way to or through an Exchange or Lotus mail server.

Of course, when one looks at personal and small office/home office business e-mail, it's rare to encounter server-based Exchange or Domino systems. Here, the market belongs to Internet service providers (*e.g.,* AOL, the major cable and telephone companies and hundreds of smaller, local players) and web mail providers (*e.g.,* Gmail, Yahoo! Mail or Hot Mail). Users employ a variety of e-mail client applications, including Microsoft Outlook, Windows Mail (formerly Outlook Express), Eudora, Entourage (on Apple machines) and, of course, their web browser and webmail. This motley crew and the enterprise behemoths are united by common e-mail *protocols* that allow messages and attachments to be seamlessly handed off between applications, providers, servers and devices.

A Snippet about Protocols

Computer network specialists are always talking about this "protocol" and that "protocol." Don't let the geek-speak get in the way. An *application protocol* is a bit of computer code that facilitates communication between applications, *i.e.,* your e-mail client and a network like the Internet. When you send a snail mail letter, the U.S. Postal Service's "protocol" dictates that you place the contents of your message in an envelope of certain dimensions, seal it, add a defined complement of address information and affix postage to the upper right hand corner of the envelope adjacent to the addressee information. Only then can you transmit the letter through the Postal Service's network of post offices, delivery vehicles and postal carriers. Omit the address, the envelope or the postage—or just fail to drop it in the mail—and Grandma gets no Hallmark this year! Likewise, computer networks rely upon protocols to facilitate the transmission of information. You invoke a protocol—*Hyper Text Transfer Protocol*—every time you type *http://* at the start of a web page address.

Incoming Mail: POP, IMAP, MAPI and HTTP E-Mail

Although Microsoft Exchange Server rules the roost in enterprise e-mail, it's by no means the most common e-mail system for the individual and small business user. When you access your

[33] http://www.ferris.com/2008/01/31/email-products-market-shares-versions-deployed-migrations-and-software-cost/ visited 11/10/08.

personal e-mail from your own Internet Service Provider (ISP), chances are your e-mail comes to you from your ISP's e-mail server in one of three ways: POP3, IMAP or HTTP, the last commonly called web- or browser-based e-mail. Understanding how these three protocols work—and differ—helps in identifying where e-mail can be found.

POP3 (for Post Office Protocol, version 3) is the oldest and most common of the three approaches and the one most familiar (by function, if not by name) to users of the Windows Mail, Outlook Express and Eudora e-mail clients. Using POP3, you connect to a mail server, download copies of all messages and, unless you have configured your e-mail client to leave copies on the server, the e-mail is deleted on the server and now resides on the hard drive of the computer you used to pick up mail. Leaving copies of your e-mail on the server seems like a great idea as it allows you to have a back up if disaster strikes and facilitates easy access of your e-mail, again and again, from different computers. However, few ISPs afford unlimited storage space on their servers for users' e-mail, so mailboxes quickly become "clogged" with old e-mails, and the servers start bouncing new messages. As a result, POP3 e-mail typically resides only on the local hard drive of the computer used to read the mail and on the back up system for the servers which transmitted, transported and delivered the messages. In short, POP is locally-stored e-mail that supports some server storage.

IMAP (Internet Mail Access Protocol) functions in much the same fashion as most Microsoft Exchange Server installations in that, when you check your messages, your e-mail client downloads just the headers of e-mail it finds on the server and only retrieves the body of a message when you open it for reading. Else, the entire message stays in your account on the server. Unlike POP3, where e-mail is searched and organized into folders locally, IMAP e-mail is organized and searched on the server. Consequently, the server (and its back up tapes) retains not only the messages but also the way the user *structured* those messages for archival.

Since IMAP e-mail "lives" on the server, how does a user read and answer it without staying connected all the time? The answer is that IMAP e-mail clients afford users the ability to synchronize the server files with a local copy of the e-mail and folders. When an IMAP user reconnects to the server, local e-mail stores are updated (synchronized) and messages drafted offline are transmitted. So, to summarize, IMAP is server-stored e-mail, with support for synchronized local storage.

A notable distinction between POP3 and IMAP e-mail centers on where the "authoritative" collection resides. Because each protocol allows for messages to reside both locally ("downloaded") and on the server, it's common for there to be a difference between the local and server collections. Under POP3, the *local* collection is deemed authoritative whereas in IMAP the *server* collection is authoritative. But for e-discovery, the important point is that the contents of the local and server e-mail stores can and do *differ.*

MAPI (Messaging Application Programming Interface) is the e-mail protocol at the heart of Windows and Microsoft's Exchange Server applications. Simple MAPI comes preinstalled on Windows machines to provide basic messaging services for Windows Mail/Outlook Express. A substantially more sophisticated version of MAPI (Extended MAPI) is installed with Microsoft Outlook and Exchange. Like IMAP, MAPI e-mail is typically stored on the server and not necessarily on the client machine. The local machine may be configured to synchronize with the server mail stores and keep a copy of mail on the local hard drive (typically in an Offline Synchronization file with the extension .OST), but this is user- and client application-dependent. Though it's exceedingly rare (especially for laptops) for there to be no local e-mail stores for a MAPI machine, it's nonetheless possible, and e-mail won't be found on the local hard drive except to the extent fragments may turn up through computer forensic examination.

HTTP (Hyper Text Transfer Protocol) mail, or web-based/browser-based e-mail, dispenses with the local e-mail client and handles all activities on the server, with users managing their e-mail using their Internet browser to view an interactive web page. Although most browser-based e-mail services support local POP3 or IMAP synchronization with an e-mail client, users may have no local record of their browser-based e-mail transactions except for messages they've affirmatively saved to disk or portions of e-mail web pages which happen to reside in the browser's cache (*e.g.,* Internet Explorer's Temporary Internet Files folder). Gmail, AOL, Hotmail and Yahoo! Mail are popular examples of browser-based e-mail services, although many ISPs (including all the national providers) offer browser-based e-mail access in addition to POP and IMAP connections.

The protocol used to carry e-mail is not especially important in electronic discovery except to the extent that it signals the most likely place where archived and orphaned e-mail can be found. Companies choose server-based e-mail systems (*e.g.,* IMAP and MAPI) for two principal reasons. First, such systems make it easier to access e-mail from different locations and machines. Second, it's easier to back up e-mail from a central location. Because IMAP and MAPI systems store e-mail on the server, the back up system used to protect server data can yield a mother lode of server e-mail.

Depending upon the back up procedures used, access to archived e-mail can prove a costly and time-consuming task or a relatively easy one. The enormous volume of e-mail residing on back up tapes and the potentially high cost to locate and restore that e-mail makes discovery of archived e-mail from backup tapes a major bone of contention between litigants. In fact, most reported cases addressing cost-allocation in e-discovery seem to have been spawned by disputes over e-mail on server back up tapes.

Outgoing Mail: SMTP and MTA

Just as the system that brings water into your home works in conjunction with a completely different system that carries wastewater away, the protocol that delivers e-mail to you is completely different from the one that transmits your e-mail. Everything discussed in the preceding paragraph concerned the protocols used to *retrieve* e-mail from a mail server.

Yet another system altogether, called **SMTP** for *Simple Mail Transfer Protocol*, takes care of outgoing e-mail. SMTP is indeed a very simple protocol and doesn't even require authentication, in much the same way as anyone can anonymously drop a letter into a mailbox. A server that uses SMTP to route e-mail over a network to its destination is called an **MTA** for *Message Transfer Agent*. Examples of MTAs you might hear mentioned by IT professionals include Sendmail, Exim, Qmail and Postfix. Microsoft Exchange Server is an MTA, too. In simplest terms, an MTA is the system that carries e-mail between e-mail servers and sees to it that the message gets to its destination. Each MTA reads the code of a message and determines if it is addressed to a user in its domain and, if not, passes the message on to the next MTA after adding a line of text to the message identifying the route to later recipients. If you've ever set up an e-mail client, you've probably had to type in the name of the servers handling your outgoing e-mail (perhaps *SMTP.yourISP.com*) and your incoming messages (perhaps *mail.yourISP.com* or *POP.yourISP.com*).

Anatomy of an E-Mail Header

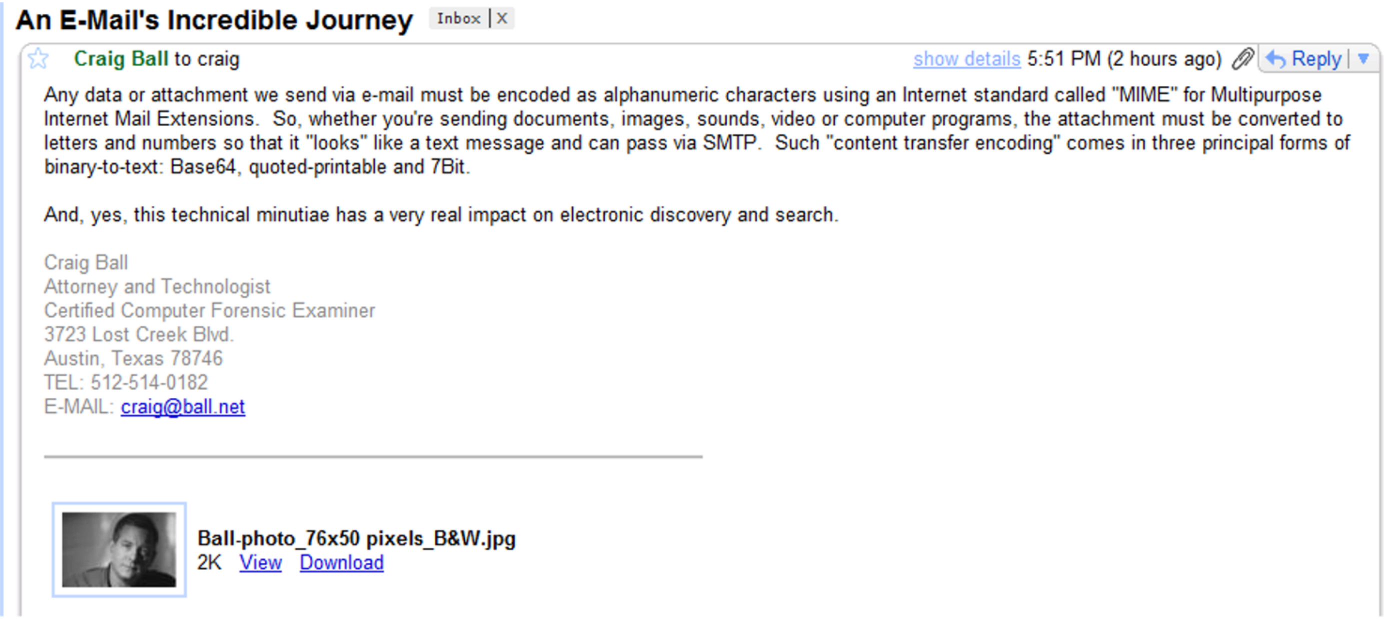
Now that we've waded through the alphabet soup of protocols managing the movement of an e-mail message, let's take a look inside the message itself. Considering the complex systems on which it lives, an e-mail is astonishingly simple in structure. The Internet protocols governing e-mail transmission require electronic messages to adhere to rigid formatting, making individual e-mails fairly easy to dissect and understand. The complexities and headaches associated with e-mail don't really attach until the e-mails are stored and assembled into databases and local stores.

An e-mail is just a plain text file. Though e-mail can be "tricked" into carrying non-text binary data like application files (*i.e.,* a Word document) or image attachments (*e.g.,* GIF or JPEG files), this piggybacking requires binary data be *encoded into text* for transmission. Consequently, even when transmitting files created in the densest computer code, *everything in an e-mail is plain text*.

Figure 1 is an e-mail I sent from one of my e-mail addresses to another with a small image attached. Transmitted and received in seconds using the same machine, the message was sliced-and-diced into two versions (plain text and HTML), and its image attachment was encoded into Base 64, restructured to comply with rigid Internet protocols. It then winged its way across several time zones and servers, each server prepending its own peculiar imprimatur.

Figure 1 is just one of a variety of different ways in which an e-mail client application (in this instance the webmail application, Gmail) may display a message. When you view e-mail onscreen or print it out, you're seeing just part of the data contained in the message and attachment. Moreover, the e-mail client may be interpreting the message data according to, *e.g.,* the time zone and daylight savings time settings of your machine or its ability to read embedded formatting information. What you don't see—or see accurately—may be of little import, or it may be critical evidence. You've got to know what lies beneath to gauge its relevance.

Figure 1: Exemplar E-Mail as Displayed in Client Application



An E-Mail's Incredible Journey

Craig Ball to craig — 5:51 PM (2 hours ago)

Any data or attachment we send via e-mail must be encoded as alphanumeric characters using an Internet standard called "MIME" for Multipurpose Internet Mail Extensions. So, whether you're sending documents, images, sounds, video or computer programs, the attachment must be converted to letters and numbers so that it "looks" like a text message and can pass via SMTP. Such "content transfer encoding" comes in three principal forms of binary-to-text: Base64, quoted-printable and 7Bit.

And, yes, this technical minutiae has a very real impact on electronic discovery and search.

Craig Ball
Attorney and Technologist
Certified Computer Forensic Examiner
3723 Lost Creek Blvd.
Austin, Texas 78746
TEL: 512-514-0182
E-MAIL: craig@ball.net

Ball-photo_76x50 pixels_B&W.jpg
2K

Figure 2 (opposite) shows the source code of the Figure 1 e-mail, sent using a browser-based Gmail account. The e-mail came from the account computerforensics@gmail.com and was addressed to craig@ball.net. A small photograph in JPEG format was attached.

Before we dissect the e-mail message in Figure 2, note that any e-mail can be divided into two parts, the header and body of the message. By design, the header details the journey taken by the e-mail from origin to destination; but be cautioned that it's a fairly simple matter for a hacker to spoof (falsify) the identification of all but the final delivery server. Accordingly, where the origin or origination date of an e-mail is suspect, the actual route of the message may need to be validated at each server along its path.

In an e-mail header, each line which begins with the word "Received:" represents the transfer of the message between or within systems. The transfer sequence is reversed chronologically such that those closest to the top of the header were inserted after those that follow, and the topmost line reflects delivery to the recipient's e-mail server. As the message passes through intervening hosts, each adds its own identifying information along with the date and time of transit.

E-Mail Autopsy: Tracing a Message's Incredible Journey

In this header, section **(A)** indicates the parts of the message designating the sender, addressee, recipient, date, time and subject line of the message. Importantly, the header also identifies the message as being formatted in MIME (MIME-Version: 1.0).[34] The **Content-Type: multipart/mixed** reference that follows indicates that the message holds both text and one or more attachments.

Though a message may be assigned various identification codes by the servers it transits in its journey (each enabling the administrator of the transiting e-mail server to track the message in the server logs), the message will contain one unique identifier assigned by the originating Message Transfer Agent. The unique identifier assigned to this message at **(B)** labeled "Message-ID:" is:

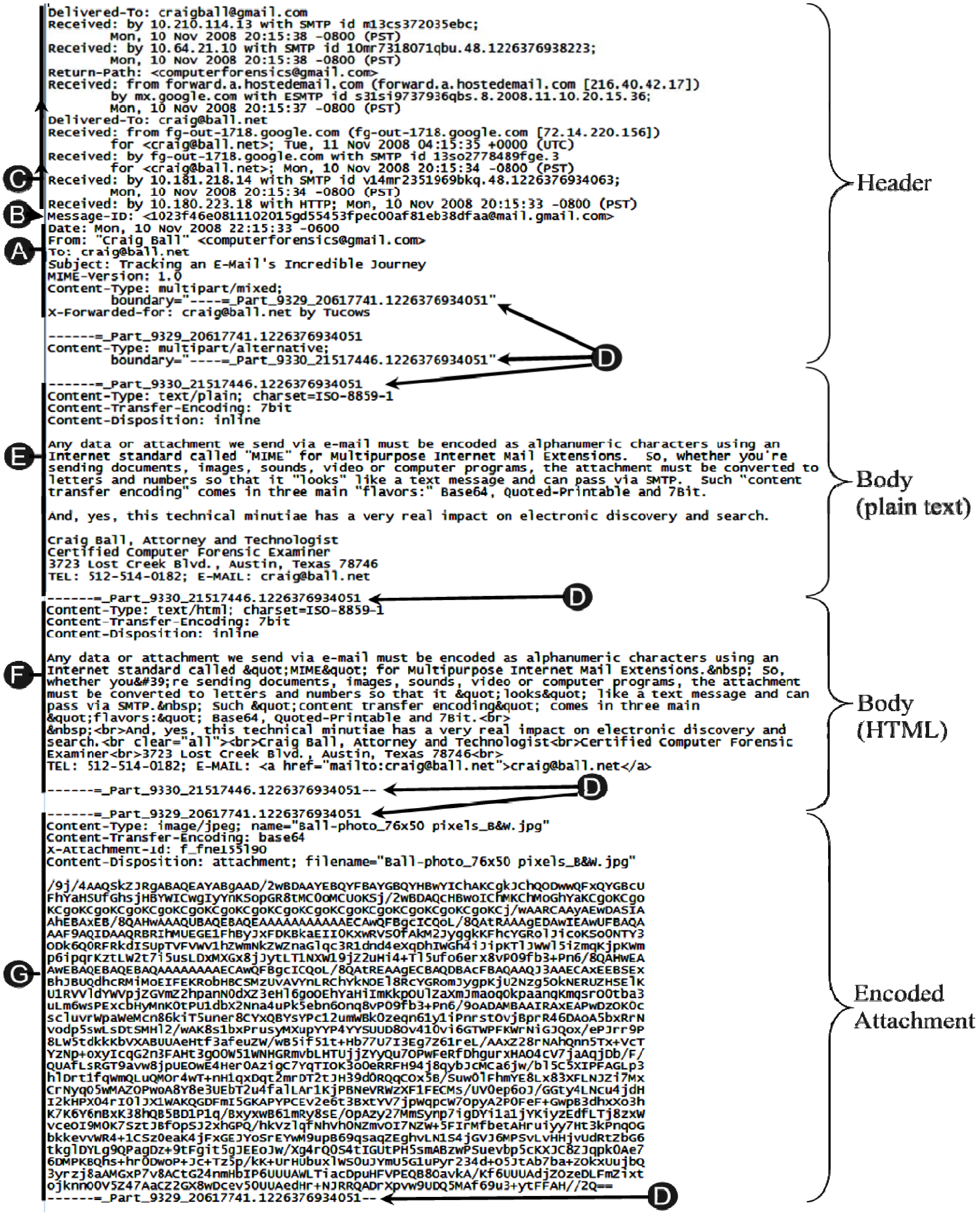
1023f46e0811102015gd55453fpec00af81eb38dfaa@mail.gmail.com.

In the line labeled "Date," both the date and time of transmittal are indicated. The time indicated is 22:15:33, and the "-0600" which follows denotes the time *difference* between the sender's local time (the system time on my computer in Austin, Texas in standard time) and Coordinated Universal Time (UTC), roughly equivalent to Greenwich Mean Time. As the offset from UTC is minus six hours on November 10, 2008, we deduce that the message was sent from a machine set to Central Standard Time, giving some insight into the sender's location. Knowing the originating computer's time and time zone can occasionally prove useful in demonstrating fraud or fabrication.

At **(A),** we see that the message was addressed to craig@ball.net from computerforensics@gmail.com; yet, the ultimate recipient of the message is (as seen at

[34] MIME, which stands for Multipurpose Internet Mail Extensions, is a seminal Internet standard that supports Non-US/ASCII character sets, non-text attachments (e.g., photos, video, sounds and machine code) and message bodies with multiple parts. Virtually all e-mail today is transmitted in MIME format.

Figure 2: Anatomy of an E-Mail



C
B
A
D
E
F
G
Header
Body (plain text)
Body (HTML)
Encoded Attachment

```
Delivered-To: craigball@gmail.com
Received: by 10.210.114.13 with SMTP id m13cs372035ebc;
        Mon, 10 Nov 2008 20:15:38 -0800 (PST)
Received: by 10.64.21.10 with SMTP id 10mr7318071qbu.48.1226376938223;
        Mon, 10 Nov 2008 20:15:38 -0800 (PST)
Return-Path: <computerforensics@gmail.com>
Received: from forward.a.hostedemail.com (forward.a.hostedemail.com [216.40.42.17])
        by mx.google.com with ESMTP id s31si9737936qbs.8.2008.11.10.20.15.36;
        Mon, 10 Nov 2008 20:15:37 -0800 (PST)
Delivered-To: craig@ball.net
Received: from fg-out-1718.google.com (fg-out-1718.google.com [72.14.220.156])
        for <craig@ball.net>; Tue, 11 Nov 2008 04:15:35 +0000 (UTC)
Received: by fg-out-1718.google.com with SMTP id 13so2778489fge.3
        for <craig@ball.net>; Mon, 10 Nov 2008 20:15:34 -0800 (PST)
Received: by 10.181.218.14 with SMTP id v14mr2351969bkq.48.1226376934063;
        Mon, 10 Nov 2008 20:15:34 -0800 (PST)
Received: by 10.180.223.18 with HTTP; Mon, 10 Nov 2008 20:15:33 -0800 (PST)
Message-ID: <1023f46e0811102015gd55453fpec00af81eb38dfaa@mail.gmail.com>
Date: Mon, 10 Nov 2008 22:15:33 -0600
From: "Craig Ball" <computerforensics@gmail.com>
To: craig@ball.net
Subject: Tracking an E-Mail's Incredible Journey
MIME-Version: 1.0
Content-Type: multipart/mixed;
        boundary="----=_Part_9329_20617741.1226376934051"
X-Forwarded-for: craig@ball.net by Tucows

------=_Part_9329_20617741.1226376934051
Content-Type: multipart/alternative;
        boundary="----=_Part_9330_21517446.1226376934051"

------=_Part_9330_21517446.1226376934051
Content-Type: text/plain; charset=ISO-8859-1
Content-Transfer-Encoding: 7bit
Content-Disposition: inline

Any data or attachment we send via e-mail must be encoded as alphanumeric characters using an
Internet standard called "MIME" for Multipurpose Internet Mail Extensions.  So, whether you're
sending documents, images, sounds, video or computer programs, the attachment must be converted to
letters and numbers so that it "looks" like a text message and can pass via SMTP.  Such "content
transfer encoding" comes in three main "flavors:" Base64, Quoted-Printable and 7Bit.

And, yes, this technical minutiae has a very real impact on electronic discovery and search.

Craig Ball, Attorney and Technologist
Certified Computer Forensic Examiner
3723 Lost Creek Blvd., Austin, Texas 78746
TEL: 512-514-0182; E-MAIL: craig@ball.net
------=_Part_9330_21517446.1226376934051
Content-Type: text/html; charset=ISO-8859-1
Content-Transfer-Encoding: 7bit
Content-Disposition: inline

Any data or attachment we send via e-mail must be encoded as alphanumeric characters using an
Internet standard called &quot;MIME&quot; for Multipurpose Internet Mail Extensions.&nbsp; So,
whether you&#39;re sending documents, images, sounds, video or computer programs, the attachment
must be converted to letters and numbers so that it &quot;looks&quot; like a text message and can
pass via SMTP.&nbsp; Such &quot;content transfer encoding&quot; comes in three main
&quot;flavors:&quot; Base64, Quoted-Printable and 7Bit.<br>
&nbsp;<br>And, yes, this technical minutiae has a very real impact on electronic discovery and
search.<br clear="all"><br>Craig Ball, Attorney and Technologist<br>Certified Computer Forensic
Examiner<br>3723 Lost Creek Blvd., Austin, Texas 78746<br>
TEL: 512-514-0182; E-MAIL: <a href="mailto:craig@ball.net">craig@ball.net</a>
------=_Part_9330_21517446.1226376934051--
------=_Part_9329_20617741.1226376934051
Content-Type: image/jpeg; name="Ball-photo_76x50 pixels_B&W.jpg"
Content-Transfer-Encoding: base64
X-Attachment-Id: f_fne155190
Content-Disposition: attachment; filename="Ball-photo_76x50 pixels_B&W.jpg"
```

```
/9j/4AAQSkZJRgABAQEAYABgAAD/2wBDAAYEBQYFBAYGBQYHBwYIChAKCgkJChQODwwQFxQYGBcU
FhYaHSUfGhsjHBYWICwgIyYnKSopGR8tMC0oMCUoKSj/2wBDAQcHBwoIChMKChMoGhYaKCgoKCgo
KCgoKCgoKCgoKCgoKCgoKCgoKCgoKCgoKCgoKCgoKCgoKCgoKCgoKCgoKCgoKCj/wAARCAAyAEwDASIA
AhEBAxEB/8QAHwAAAQUBAQEBAQEAAAAAAAAAAAECAwQFBgcICQoL/8QAtRAAAgEDAwIEAwUFBAQA
AAF9AQIDAAQRBRIhMUEGE1FhByJxFDKBkaEII0KxwRVS0fAkM2JyggkKFhcYGRolJicoKSo0NTY3
ODk6Q0RFRkdISUpTVFVWV1hZWmNkZWZnaGlqc3R1dnd4eXqDhIWGh4iJipKTlJWWl5iZmqKjpKWm
p6ipqrKztLW2t7i5usLDxMXGx8jJytLT1NXW19jZ2uHi4+Tl5ufo6erx8vP09fb3+Pn6/8QAHwEA
AwEBAQEBAQEBAQAAAAAAAAECAwQFBgcICQoL/8QAtREAAgECBAQDBAcFBAQAAQJ3AAECAxEEBSEx
BhJBUQdhcRMiMoEIFEKRobHBCSMzUVAVYnLRChYkNOEl8RcYGRomJygpKjU2Nzg5OkNERUZHSElK
U1RVVldYWVpjZGVmZ2hpanN0dXZ3eHl6goOEhYaHiImKkpOUlZaXmJmaoqOkpaanqKmqsrO0tba3
uLm6wsPExcbHyMnK0tPU1dbX2Nna4uPk5ebn6Onq8vP09fb3+Pn6/9oADAMBAAIRAxEAPwDzOK0c
scluvrWpaWeMcn86kiT5uner8CYxQBYsYPc12umWBk0zeqn61y1iPnrst0vjBprR46DAoA5bxRrN
vodp5swLsDtSMHl2/wAK8s1bxPrusyMXupYYP4YYSUUD8Ov410vi6GTWPFKWrNiGJQox/ePJrr9P
8LW5tdkkKbVXABUUAeHtf3afeuZW/wB5if51t+Hb77U7I3Eg7Z61reL/AAxZ28rNAhQnn5Tx+VcT
YzNp+oxyIcqG2n3FAHt3gOOW51WNHGRmvbLHTUjjZYyQu7OPWFeRfDhgurxHAO4cV7jaAqjDb/F/
QUAfLsRGT9avw8jpUEOwE4Her0AzigC7YqTIOK3oO0eRRFH94j8qybJcMCa6jw/bl5C5XIPFAGLp3
hlDrt1fqwmQLuQMOr4wT+nH1qxDqt2mrDT2tJH39d0RQqCox5B/Suw0lFhmYE8LxFLNJZi7Mx
CrNyq05wMAZOPwoA8Y8e3UEbT2u4falLAr1KjPBNeVRWzXF1FECMs/VRGQ4VGtmCvbt3pGrG0jdH
I2kHPX04rI0lJX1WAKQGDFmI5GKAPYPCEv2e6t3BxtYV7jpWqpcW7OpyA2POFoF0FeF+GwpB3dhxXo3h
K7K6Y6nBxK38hQB5BD1P1q/BxyxwB61mRy8sE/OpAzy27MmSynp7igDYi1a1jYKiyzEdfLTj8zx
vceOI9MOK7SztJBfopSJ2xhGPQ/hkvzlqfNhvh0NZmvOI7NZW+5FIrEYBbJEYoJw/Xg4rQ054tIGUtPH5smABzwPSuevbp5cKXJC8zJqpk0Ae7
tkglDYLg9QPagDz+9tFgit5gJEEoJw/Xg4rQ054tIGUtPH5smABzwPSuevbp5cKXJC8zJqpk0Ae7
6DMPKBQhs+hr0DwoP+Jc+Tz5p/kK+UrHUbux1wS0uJYmU5G1uPyr234d+o5JtAb7ba+zOkxUujbQ
3yrzj8aAMGxP7v8ACtG24nmHbIP6UUUAWLTiacDpuHFVPEQB80avkA/Kf6UUUAdjZ0ozeDLFmZixt
ojknn00V5Z47AaCZ2GX8wDcev50UUAedHr+NJRRQADrXpvw9UhXLANu9UUQ5MAf69u3+ytFFAH//2Q==
```

```
------=_Part_9329_20617741.1226376934051--
```

the very top of the page) craigball@gmail.com. How this transpired can be deciphered from the header data, read from the bottom up.

The message was created and sent using Gmail web interface; consequently the first hop **(C)** indicates that the message was transmitted using HTTP and first received by IP (Internet Protocol) address 10.180.223.18 at 20:15:33 -0800 (PST). Note that the server marks time in Pacific Standard Time, suggesting it may be located on the West Coast. The message is immediately handed off to another IP address 10.181.218.14 using Simple Mail Transfer Protocol, denoted by the initials SMTP. Next, we see another SMTP hand off to Google's server named "fg-out-1718.google.com" (IP address 72.14.220.156), which immediately transmits the message to a server with the IP address 216.40.42.17 and keeping time in UTC. A check of that IP address reveals that it's registered to Tucows International in Toronto, Canada.

Tucows is the host of my craig@ball.net address, which is configured to forward incoming messages to my other Gmail address, craigball@gmail.com. The forwarding is handled by a server called *forward.a.hostedemail.com*, and we then see the message received by server *MX.google.com*, transferred via SMTP to a server at IP address 10.64.21.10, then finally come to rest, delivered via SMTP to my craigball@gmail.com address via a server at 10.210.114.

As we examine the structure of the e-mail, we see that it contains content boundaries separating its constituent parts **(D).** These content boundary designators serve as delimiters; that is, sequences of one or more characters used to specify the boundary between text or data streams.[35] In order to avoid confusion of the boundary designator with message text, a complex sequence of characters is generated to serve as the two boundary designators used in this message. The first, called "_Part_9329_20617741.1226376934051," serves to separate the message header from the message body and signal the end of the message. The second delimiter, called "----=_Part_9330_21517446.1226376934051," denotes the boundaries between the segments of the message body: here, plain text content **(E)**, HTML content **(F)** and the encoded attachment **(G)**.

I didn't draft the message in *both* plain text and HTML formats, but my e-mail client thoughtfully did so to insure that my message won't confuse recipients using e-mail clients unable to display the richer formatting supported by HTML. For these recipients, there is a plain text version, too (albeit without the bolding, italics, hyperlinks and other embellishments of HTML). That the message carries alternative versions of the text is flagged by the designation at the break between header and message body stating: "**Content-Type: multipart/alternative**."

Looking more closely at the message boundaries, we see that each boundary delimiter is followed by Content-Type and Content-Transfer-Encoding designations. The plain text version

[35] The use of delimiters should be a familiar concept to those accustomed to specifying load file formats to accompany document image productions employed in e-discovery, where commas typically serve as field delimiters. Hence, these load files are sometimes referred to as CSV files (for comma-separated values).

of the message **(E)** begins: "**Content-Type: text/plain; charset=ISO-8859-1**," followed by "**Content-Transfer-Encoding: 7bit.**" The first obviously denotes plain text content using the very common ISO-8859-1 character encoding more commonly called "Latin 1."[36] The second signals that the content that follows consists of standard ASCII characters which historically employ 7 bits to encode 128 characters.

Not surprisingly, the boundary for the HTML version uses the Content-Type designator "text/html."

The most interesting and complex part of the message **(F)** starts after the second to last boundary delimiter with the specifications:
Content-Type: image/jpeg; name="Ball-photo_76x50 pixels_B&W.jpg"
Content-Transfer-Encoding: base64

The content type is self explanatory: an image in the JPEG format common to digital photography. The "name" segment obviously carries the name to be re-assigned to the attached photograph when decoded at its destination. But where, exactly, is the photograph?

Recall that to travel as an e-mail attachment, binary content (like photos, sound files, video or machine codes) must first be converted to plain text characters. Thus, the photograph has been encoded to a format called Base64, which substitutes 64 printable ASCII characters (A–Z, a–z, 0–9, + and /) for any binary data or for foreign characters, like Cyrillic or Chinese, that can be represented by the Latin alphabet.[37]

Accordingly, the attached JPEG photograph with the filename "Ball-photo_76x50 pixels_B&W.jpg," has been encoded from non-printable binary code into those 26 lines of gibberish comprising nearly 2,000 plain text characters **(G) and Figure 3.** It's now able to traverse the network as an e-mail, yet easily be converted back to binary data when the message reaches its destination.

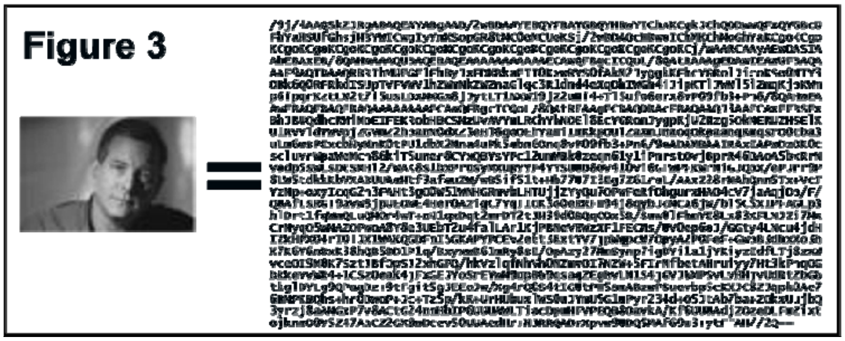


Figure 3

[36] In simplest terms, a character set or encoding pairs a sequence of characters (like the Latin alphabet) with numbers, byte values or other signals in much the same way as Morse code substitutes particular sequences of dots and dashes for letters. It's the digital equivalent of the Magic Decoder Rings once found in boxes of Cracker Jacks.

[37] A third common transfer encoding is called "quoted-printable" or "QP encoding." It facilitates transfer of non-ASCII 8-bit data as 7-bit ASCII characters using three ASCII characters (the "equals" sign followed by two hexadecimal characters: 0-9 and A-F) to stand in for a byte of data Quoted-printable is employed where the content to be encoded is predominantly ASCII text coupled with some non-ASCII items. Its principal advantage is that it allows the encoded data to remain largely intelligible to readers.

Clearly, e-mail clients don’t display all the information contained in a message’s source but instead parse the contents into the elements we most want to see: To, From, Subject, body, and attachment. If you decide to try a little digital detective work on your own e-mail, you’ll find that some e-mail client software doesn’t make it easy to see complete header information. Microsoft’s Outlook mail client makes it difficult to see the complete message source; however, you can see message headers for individual e-mails by opening the e-mail, then selecting “View” followed by “Options” until you see the “Internet headers” window on the Message Option menu. In Microsoft Outlook Express (now Windows Mail), highlight the e-mail item you want to analyze and then select “File” from the Menu bar, then “Properties,” then click the “Details” tab followed by the “Message Source” button. For Gmail, select “Show Original” from the Reply button pull-down menu.

The lesson from this is that what you see displayed in your e-mail client application isn’t really the e-mail. It’s an *arrangement* of selected *parts* of the message, frequently modified in some respects from the native message source that traversed the network and Internet and, as often, supplemented by metadata (like message flags, contact data and other feature-specific embellishments) unique to your software and setup. What you see handily displayed as a discrete attachment is, in reality, encoded into the message body. The time assigned to message is calculated relative to your machine’s time and DST settings. Even the sender’s name may be altered based upon the way your machine and contact’s database is configured. What you see is not always what you get (or got).

Hashing and Deduplication

Hashing is the use of mathematical algorithms to calculate a unique sequence of letters and numbers to serve as a “fingerprint” for digital data. These fingerprint sequences are called “message digests” or, more commonly, “hash values.”

The ability to “fingerprint” data makes it possible to identify identical files without the necessity of examining their content. If the hash values of two files are identical, the files are identical. This file-matching ability allows hashing to be used to de-duplicate collections of electronic files before review, saving money and minimizing the potential for inconsistent decisions about privilege and responsiveness for identical files.

Although hashing is a useful and versatile technology, it has a few shortcomings. Because the tiniest change in a file will alter that file’s hash value, hashing is of little value in comparing files that have any differences, even if those differences have no bearing on the substance of the file. Applied to e-mail, we understand from our e-mail “autopsy” that messages contain unique identifiers, time stamps and routing data that would frustrate efforts to compare one complete message to another using hash values. Looking at the message as a whole, multiple recipients of the same message have different versions insofar as their hash values.

Consequently, deduplication of e-mail messages is accomplished by calculating hash values for selected segments of the messages and comparing those segment values. Thus, hashing e-mails for deduplication will omit the parts of the header data reflecting, *e.g.,* the message identifier and the transit data. Instead, it will hash just the data seen in, *e.g.,* the To, From, Subject and Date lines, message body and encoded attachment. If these match, the message can be said to be *practically* identical.

For example, a deduplication application might hash only segments **(A)**, **(E)** and **(G)** of Figure 2. If the hash values of these segments match the hash values of the same segments of another message, can we say they are the same message? Probably, but it could also be important to evaluate emphasis added by HTML formatting (*e.g.,* text in red or underlined) or information about blind carbon copy recipients. The time values or routing information in the headers may also be important to reliably establishing authenticity, reliability or sequence.

By hashing particular segments of messages and selectively comparing the hash values, it's possible to gauge the *relative* similarity of e-mails and perhaps eliminate the cost to review messages that are *inconsequentially* different. This concept is called "near deduplication." It works, but it's important to be aware of exactly what it's excluding and why. It's also important to advise your opponents when employing near deduplication and ascertain whether you're mechanically excluding evidence the other side deems relevant and material.

Hash deduplication of e-mail is tricky. Time values may vary, along with the apparent order of attachments. These variations, along with minor formatting discrepancies, may serve to prevent the exclusion of items defined as duplicates. When this occurs, be certain to delve into the reasons *why* apparent duplicates aren't deduplicating, as such errors may be harbingers of a broader processing problem.

Local E-Mail Storage Formats and Locations

Suppose you're faced with a discovery request for a client's e-mail and there's no budget or time to engage an e-discovery service provider or ESI expert?

Where are you going to look to find stored e-mail, and what form will it take?

"Where's the e-mail?" It's a simple question, and one answered too simply and often wrongly by, "It's on the server" or "The last 60 days of mail is on the server and the rest is purged." Certainly, much e-mail will reside on the server, but most e-mail is elsewhere; and it's never all gone in practice, notwithstanding retention policies. The true location and extent of e-mail depends on systems configuration, user habits, backup procedures and other hardware, software and behavioral factors. This is true for mom-and-pop shops, for large enterprises and for everything in-between.

Going to the server isn't the wrong answer. It's just not the whole answer. In a matter where I was tasked to review e-mails of an employee believed to have stolen proprietary information, I went first to the company's Microsoft Exchange e-mail server and gathered a lot of unenlightening e-mail. Had I stopped there, I would've missed the Hotmail traffic in the Temporary Internet Files folder and the Short Message Service (SMS) exchanges in the PDA synchronization files. I'd have overlooked the Microsoft Outlook archive file (archive.pst) and offline synchronization file (Outlook.ost) on the employee's laptop, collectively holding thousands more e-mails, including some "smoking guns" absent from the server. These are just some of the many places e-mails without counterparts on the server may be found. Though an exhaustive search of every nook and cranny may not be required, you need to know your options in order to assess feasibility, burden and cost.

E-mail resides in some or all of the following venues, grouped according to relative accessibility:

Easily Accessible:

- **E-Mail Server:** Online e-mail residing in active files on enterprise servers: MS Exchange e.g., (.edb, .stm, .log files), Lotus Notes (.nsf files), Novell GroupWise (.db files)
- **File Server:** E-mail saved as individual messages or in container files on a user's network file storage area ("network share").
- **Desktops and Laptops**: E-mail stored in active files on local or external hard drives of user workstation hard drives (*e.g.,* .pst, .ost files for Outlook and .nsf for Lotus Notes), laptops (.ost, .pst, .nsf), mobile devices, and home systems, particularly those with remote access to networks.
- OLK system subfolders holding viewed attachments to Microsoft Outlook messages, *including deleted messages*.
- Nearline e-mail: Optical "juke box" devices, backups of user e-mail folders.
- Archived or journaled e-mail: *e.g.,* Autonomy Zantaz Enterprise Archive Solution, EMC EmailXtender, Mimosa NearPoint, Symantec Enterprise Vault.

Accessible, but Often Overlooked:

- E-mail residing on non-party servers: ISPs (IMAP, POP, HTTP servers), Gmail, Yahoo! Mail, Hotmail, etc.
- E-mail forwarded and cc'd to external systems: Employee forwards e-mail to self at personal e-mail account.
- E-mail threaded as text behind subsequent exchanges.
- Offline local e-mail stored on removable media: External hard drives, thumb drives and memory cards, optical media: CD-R/RW, DVD-R/RW, floppy drives, zip drives.
- Archived e-mail: Auto-archived or saved under user-selected filename.
- Common user "flubs": Users experimenting with export features unwittingly create e-mail archives.
- Legacy e-mail: Users migrate from e-mail clients "abandoning" former e-mail stores. Also, e-mail on mothballed or re-tasked machines and devices.

- E-mail saved to other formats: PDF, .tiff, .txt, .eml, .msg, etc.
- E-mail contained in review sets assembled for other litigation/compliance purposes.
- E-mail retained by vendors or third- parties (*e.g.,* former service provider or attorneys)
- Paper print outs.

Less Accessible:

- Offline e-mail on server backup tapes and other media.
- E-mail in forensically accessible areas of local hard drives and re-tasked/reimaged legacy machines: deleted e-mail, internet cache, unallocated clusters.

The levels of accessibility above speak to practical challenges to ease of access, not to the burden or cost of review. The burden continuum isn't a straight line. That is, it may be less burdensome or costly to turn to a small number of less accessible sources holding relevant data than to broadly search and review the contents of many accessible sources. Ironically, it typically costs much more to process and review the contents of a mail server than to undertake forensic examination of a key player's computer; yet, the former is routinely termed "reasonably accessible" and the latter not.

The issues in the case, key players, relevant time periods, agreements between the parties, applicable statutes, decisions and orders of the court determine the extent to which locations must be examined; however, the failure to diligently identify relevant e-mail carries such peril that caution should be the watchword. Isn't it wiser to invest more effort to know exactly what the client has—even if it's not reasonably accessible and will not be searched or produced—than concede at the sanctions hearing the client failed to preserve and produce evidence it didn't know it because no one looked?

Looking for E-Mail 101

Because an e-mail is just a text file, individual e-mails could be stored as discrete text files. But that's not a very efficient or speedy way to manage a large number of messages, so you'll find that most e-mail client software doesn't do that. Instead, e-mail clients employ proprietary database files housing e-mail messages, and each of the major e-mail clients uses its own unique format for its database. Some programs encrypt the message stores. Some applications merely display e-mail housed on a remote server and do not store messages locally (or only in fragmentary way). The only way to know with certainty if e-mail is stored on a local hard drive is to look for it.

Merely checking the e-mail client's settings is insufficient because settings can be changed. Someone not storing server e-mail today might have been storing it a month ago. Additionally, users may create new identities on their systems, install different client software, migrate from other hardware or take various actions resulting in a cache of e-mail residing on their systems without their knowledge*. If they don't know it's there, they can't tell you it's not*. On local hard drives, you've simply got to know what to look for and where to look…*and then you've got to look for it.*

For many, computer use is something of an unfolding adventure. One may have first dipped her toes in the online ocean using browser-based e-mail or an AOL account. Gaining computer-savvy, she may have signed up for broadband access or with a local ISP, downloading e-mail with Netscape Messenger or Microsoft Outlook Express. With growing sophistication, a job change or new technology at work, the user may have migrated to Microsoft Outlook or Lotus Notes as an e-mail client. Each of these steps can orphan a large cache of e-mail, possibly unbeknownst to the user but still fair game for discovery. Again, you've simply got to know what to look for and where to look.

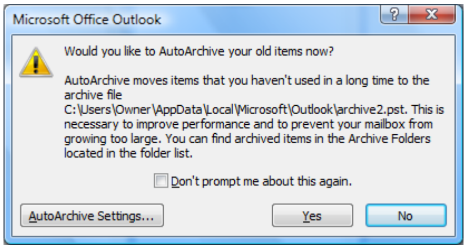
One challenge you'll face when seeking stored e-mail is that every user's storage path is different. This difference is not so much the result of a user's ability to specify the place to store e-mail—which few do, but which can make an investigator's job more difficult when it occurs—but more from the fact that operating systems are designed to support multiple users and so must assign unique identities and set aside separate storage areas for different users. Even if only one person has used a Windows computer, the operating system will be structured at the time of installation so as to make way for others. Thus, finding e-mail stores will hinge on your knowledge of the User's Account Name or Globally Unique Identifier (GUID) string assigned by the operating system. This may be as simple as the user's name or as obscure as the 128-bit hexadecimal value {721A17DA-B7DD-4191-BA79-42CF68763786}. Customarily, it's both.

***Caveat**: Before you or anyone on your behalf "poke around" on a computer system seeking a file or folder, recognize that absent the skilled use of specialized tools and techniques, such activity <u>will</u> result in changing data on the drive. Some of the changed data may be forensically significant (such as file access dates) and could constitute <u>spoliation of evidence</u>. If, under the circumstances of the case or matter, your legal or ethical obligation is to preserve the integrity of electronic evidence, then you and your client may be obliged to entrust the search only to qualified persons*

Finding Outlook E-Mail

PST: Microsoft Outlook is by far the most widely used e-mail client in the business environment. Despite the confusing similarity of their names, Outlook is a much different and substantially more sophisticated application than Outlook Express (now called Windows Mail). One of many important differences is that where Outlook Express stores messages in plain text, Outlook encrypts and compresses messages. But the most significant challenge Outlook poses in discovery is the fact that all of its message data and folder structure, along with all other information managed by the program (except the user's Contact data), is stored within a single, often massive, database file with the file extension .pst. The Outlook PST file format is proprietary and its structure poorly documented, limiting your options when trying to view or process its contents to Outlook itself or one of a handful of PST file reader programs available for purchase and download via the Internet.

OST: While awareness of the Outlook PST file has grown, even many lawyers steeped in e-discovery fail to consider a user's Outlook .ost file. The OST or offline synchronization file is commonly encountered on laptops configured for Exchange Server environments. It exists for the purpose of affording access to messages when the user has no active network connection. Designed to allow work to continue on, *e.g.,* airplane flights, local OST files often hold messages purged from the server—at least until re-synchronization. It's not unusual for an OST file to hold e-mail unavailable from any other comparably-accessible source.

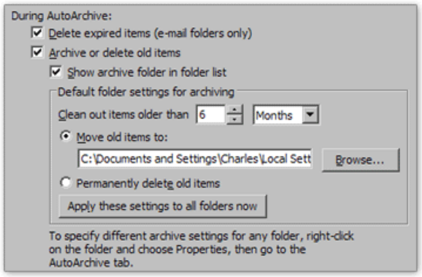


Microsoft Office Outlook

Would you like to AutoArchive your old items now?

AutoArchive moves items that you haven't used in a long time to the archive file C:\Users\Owner\AppData\Local\Microsoft\Outlook\archive2.pst. This is necessary to improve performance and to prevent your mailbox from growing too large. You can find archived items in the Archive Folders located in the folder list.

☐ Don't prompt me about this again.



During AutoArchive:

☑ Delete expired items (e-mail folders only)

☑ Archive or delete old items

☑ Show archive folder in folder list

Default folder settings for archiving

Clean out items older than 6 Months

● Move old items to: C:\Documents and Settings\Charles\Local Sett

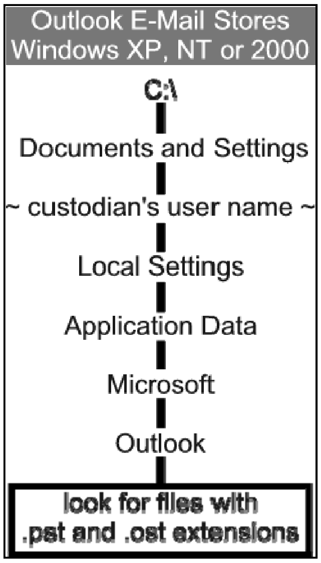
○ Permanently delete old items

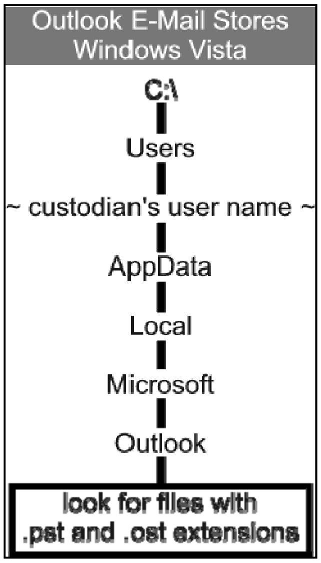
To specify different archive settings for any folder, right-click on the folder and choose Properties, then go to the AutoArchive tab.

Archive.pst: Another file to consider is one customarily called, "archive.pst." As its name suggests, the archive.pst file holds older messages, either stored automatically or by user-initiated action. If you've used Outlook without manually configuring its archive settings, chances are the system periodically asks whether you'd like to auto archive older items. Every other week (by default), Outlook 2003 seeks to auto archive any Outlook items older than six months (or for Deleted and Sent items older than two months for Outlook 2007). Users can customize these intervals, turn archiving off or instruct the application to permanently delete old items.

Outlook Mail Stores Paths

To find the Outlook message stores on machines running Windows XP/NT/2000 or Vista, drill down from the root directory (C:\ for most users) according to the path diagram on the right for the applicable operating system. The default filename of Outlook.pst/ost may vary if a user has opted to select a different designation or maintains multiple e-mail stores; however, it's rare to see users depart from the default settings. Since the location of the PST and OST files can be changed by the user, it's a good idea to do a search of all files and folders to identify any files ending with the .pst and .ost extensions.

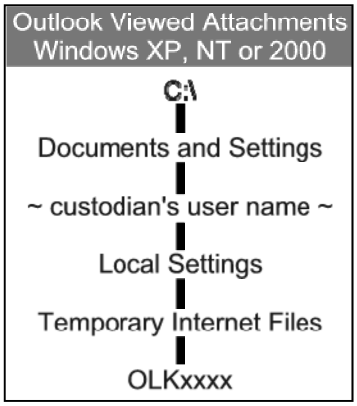


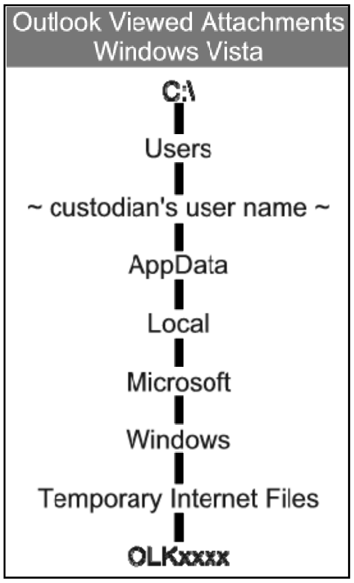


"Temporary" OLK Folders

Note that by default, when a user opens an attachment to a message from within Outlook (as opposed to saving the attachment to disk and then opening it), Outlook stores a copy of the attachment in a “temporary” folder. But don’t be misled by the word “temporary.” In fact, the folder isn’t going anywhere and its contents—sometimes voluminous--tend to long outlast the messages that transported the attachments. Thus, litigants should be cautious about representing that Outlook e-mail is “gone” if the e-mail’s attachments are not.

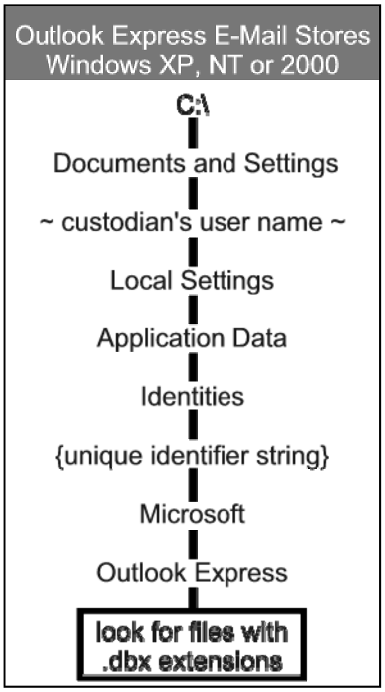
The Outlook viewed attachment folder will have a varying name for every user and on every machine, but it will always begin with the letters “OLK” followed by several randomly generated numbers and uppercase letters (e.g., OLK943B, OLK7AE, OLK167, etc.). To find the OLKxxxx viewed attachments folder on machines running Windows XP/NT/2000 or Vista, drill down from the root directory according to the path diagrams on the right for the applicable operating system.[38]

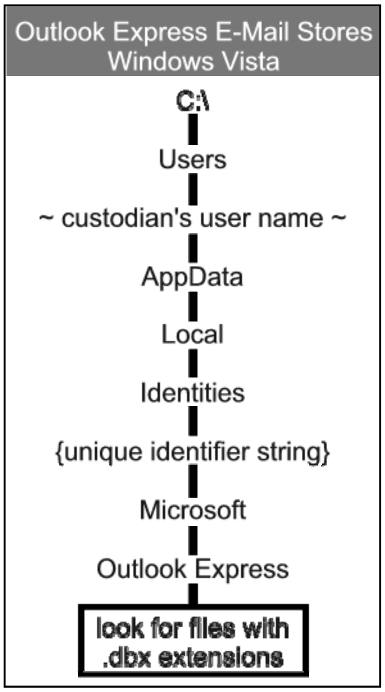




Finding Outlook Express E-Mail

Outlook Express has been bundled with every Windows operating system for about fifteen years, so you are sure to find at least the framework of an e-mail cache created by the program. Beginning with the release of Microsoft Vista, the Outlook Express application was renamed Windows Mail and the method of message storage was changed from a database format to storage as individual messages. More recently, Microsoft has sought to replace both Outlook Express on Windows XP and Windows Mail on Windows Vista with a freeware application called Windows Live Mail.





Outlook Express places e-mail in database files with the extension .dbx.

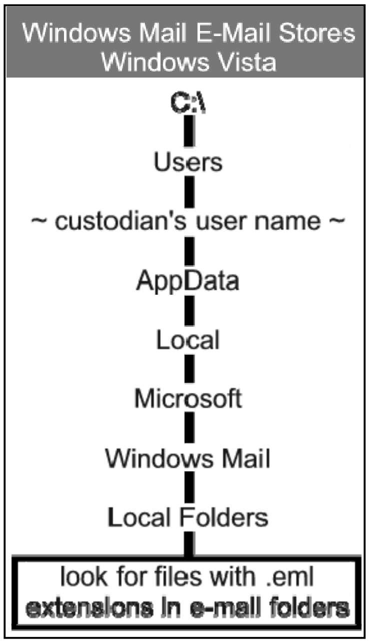
[38] By default, Windows hides system folders from users, so you may have to first make them visible. This is accomplished by starting Windows Explorer, then selecting ‘Folder Options’ from the Tools menu in Windows XP or ‘Organize>Folder and Search Options’ in Vista. Under the 'View' tab, scroll to ‘Files and Folders' and check 'Show hidden files and folders' and uncheck 'Hide extensions for known file types' and 'Hide protected operating system files. Finally, click ‘OK.’

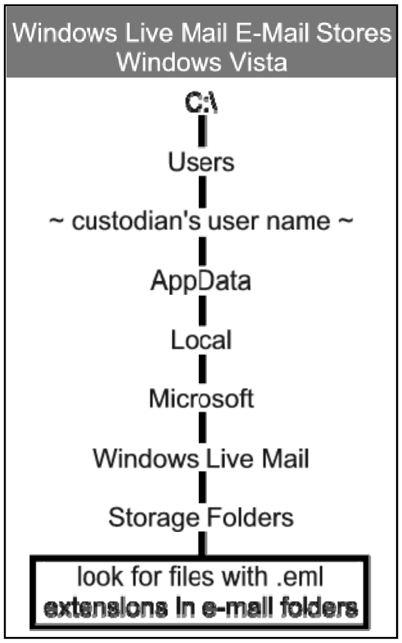
The program creates a storage file for each e-mail storage folder that it displays, so expect to find at least Inbox.dbx, Outbox.dbx, Sent Items.dbx and Deleted Items.dbx. If the user has created other folders to hold e-mail, the contents of those folders will reside in a file with the structure *foldername*.dbx. Typically on a Windows XP/NT/2K system, you will find Outlook Express .dbx files in the path shown in the diagram at near right on the preceding page. Though less frequently encountered on a Windows Vista machine, the .dbx files would be found in the default location path shown at far right on preceding page. Multiple identifier strings (Globally Unique Identifiers) string listed in the Identities subfolder may be an indication of multiple e-mail stores and/or multiple users of the computer. You will need to check each Identity's path. Another approach is to use the Windows Search function (if under windows XP) to find all files ending .dbx, but be very careful to enable all three of the following Advanced Search options before running a search: Search System Folders, Search Hidden Files and Folders, and Search Subfolders. If you don't, you won't find any—or at least not all—Outlook Express e-mail stores. Be certain to check the paths of the files turned up by your search as it can be revealing to know whether those files turned up under a particular user identity, in Recent Files or even in the Recycle Bin.

Finding Windows Mail and Windows Live Mail E-Mail Stores

You'll encounter Windows Mail on a machine running Windows Vista. By default, Windows Mail messages will be stored in oddly named individual files with the extension .eml and these housed in standard (*i.e.,* Inbox, Outbox, Sent Items, deleted Items, etc.) and user-created folders under the path diagrammed at near right.

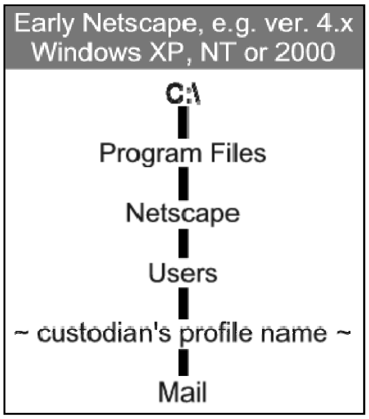
Similarly, Windows Live Mail running on Vista will store messages as oddly named individual files with the extension .eml, within standard and user-created folders under the path seen at far right.



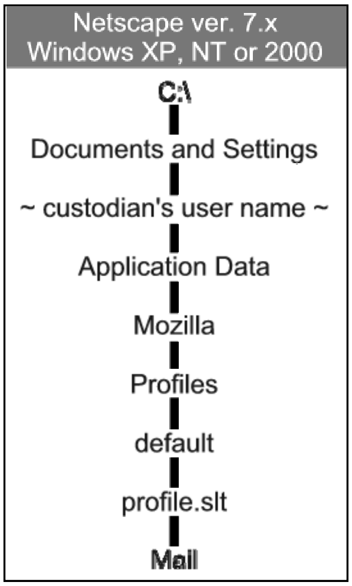


When collecting mail from these mail stores, it's important to capture both the message and the folder structure because, unlike the structured container seen in, *e.g.,* Outlook PST or OST files, the user's folder structure is not an integral part of the message storage scheme in Windows Mail or Live Mail.

Finding Netscape E-Mail



Though infrequently seen today, Netscape and its Mozilla e-mail client ruled the Internet before the browser wars left it crippled and largely forgotten. If you come across a Netscape e-mail client installation, keep in mind that the location of its e-mail stores will vary depending upon the version of the program installed. If it is an older version of the program, such as Netscape 4.x and a default installation, you will find the e-mail stores by drilling down the path depicted at right. Expect to find two files for each mailbox folder, one containing the message text with no extension (*e.g.,* Inbox) and another which serves as an index file with a .snm extension (*e.g.,* Inbox.snm).



In the last version of Netscape to include an e-mail client (Netscape 7.x), both the location and the file structures/names were changed. Drill down using the default path shown at right and locate the folder for the e-mail account of interest, usually the name of the e-mail server from which messages are retrieved. If you don’t see the Application Data folder, go to the Tools Menu, pull down to Folder Options, click on the View tab, and select "Show Hidden Files and Folders," then click “OK.” You should find two files for each mailbox folder, one containing the message text with no extension (*e.g.,* Sent) and another which serves as an index file with a .msf extension (*e.g.,* Sent.msf). If you can’t seem to find the e-mail stores, you can either launch a Windows search for files with the .snm and .msf extensions (*e.g.* *.msf) or, if you have access to the e-mail client program, you can check its configuration settings to identify the path and name of the folder in which e-mail is stored.

Microsoft Exchange Server

About 200 million people get their work e-mail via a Microsoft product called Exchange Server. It’s been sold for about a dozen years and its latest version was introduced in 2007; although, most users continue to rely on the 2003 version of the product.

The key fact to understand about an e-mail server is that it’s a *database* holding the messages (and calendars, contacts, to-do lists, journals and other datasets) of multiple users. E-mail servers are configured to maximize performance, stability and disaster recovery, with little consideration given to compliance and discovery obligations. If anyone anticipated the role e-mail would play in virtually every aspect of business today, their prescience never influenced the design of e-mail systems. E-mail evolved largely by accident, absent the characteristics of competent records management, and only lately are tools emerging that are designed to catch up to legal and compliance duties.

The other key thing to understand about enterprise e-mail systems is that, unless you administer the system, it probably doesn't work the way you imagine. The exception to that rule is if you can distinguish between Local Continuous Replication (LCR), Clustered Continuous Replication (CCR), Single Copy Cluster (SCC) and Standby Continuous Replication (SCR). In that event, I should be reading *your* paper!

But to underscore the potential for staggering complexity, appreciate that the latest Enterprise release of Exchange Server 2007 supports up to 50 storage groups per server of up to 50 message stores per group, for a database size limit of 16 terabytes. If there is an upper limit on how many users can share a single message store, I couldn't ascertain what it might be!

Though the preceding pages dealt with finding e-mail stores on local hard drives, in disputes involving medium- to large-sized enterprises, the e-mail server is likely to be the initial nexus of electronic discovery efforts. The server is a productive venue in electronic discovery for many reasons, among them:

- The periodic backup procedures which are a routine part of prudent server management tend to shield e-mail stores from those who, by error or guile, might delete or falsify data on local hard drives.
- The ability to recover deleted mail from archival server backups may obviate the need for costly and unpredictable forensic efforts to restore deleted messages.
- Data stored on a server is often less prone to tampering by virtue of the additional physical and system security measures typically dedicated to centralized computer facilities as well as the inability of the uninitiated to manipulate data in the more-complex server environment.
- The centralized nature of an e-mail server affords access to many users' e-mail and may lessen the need for access to workstations at multiple business locations or to laptops and home computers.
- Unlike e-mail client applications, which store e-mail in varying formats and folders, e-mail stored on a server can usually be located with relative ease and adhere to common file formats.
- The server is the crossroads of corporate electronic communications and the most effective chokepoint to grab the biggest "slice" of relevant information in the shortest time, for the least cost.

Of course, the big advantage of focusing discovery efforts on the mail server (*i.e.,* it affords access to thousands or millions of messages) is also its biggest disadvantage (someone has to *collect and review* thousands or millions of messages). Absent a carefully-crafted and, ideally, agreed-upon plan for discovery of server e-mail, both requesting and responding parties run the risk of runaway costs, missed data and wasted time.

E-mail originating on servers is generally going to fall into two realms, being online “live” data, which is deemed reasonably accessible, and offline “archival” data, routinely deemed inaccessible based on considerations of cost and burden.[39] Absent a change in procedure, “chunks” of data routinely migrate from accessible storage to less accessible realms—on a daily, weekly or monthly basis—as selected information on the server is replicated to backup media and deleted from the server’s hard drives.

The ABCs of Exchange

Because it’s unlikely most readers will be personally responsible for collecting e-mail from an Exchange Server and mail server configurations can vary widely, the descriptions of system architecture here are offered only to convey a rudimentary understanding of common Exchange architecture.

The 2003 version of Exchange Server stores data in a Storage Group containing a Mailbox Store and a Public Folder Store, each composed of two files: an .edb file and a .stm file. Mailbox Store, Priv1.edb, is a rich-text database file containing user’s email messages, text attachments and headers. Priv1.stm is a streaming file holding SMTP messages and containing multimedia data formatted as MIME data. Public Folder Store, Pub1.edb, is a rich-text database file containing messages, text attachments and headers for files stored in the Public Folder tree. Pub1.stm is a streaming file holding SMTP messages and containing multimedia data formatted as MIME data. Exchange Server 2007 did away with STM files altogether, shifting their content into the EDB database files.

Storage Groups also contain system files and transaction logs. Transaction logs serve as a disaster recovery mechanism that helps restore an Exchange after a crash. Before data is written to an EDB file, it is first written to a transaction log. The data in the logs can thus be used to reconcile transactions after a crash.

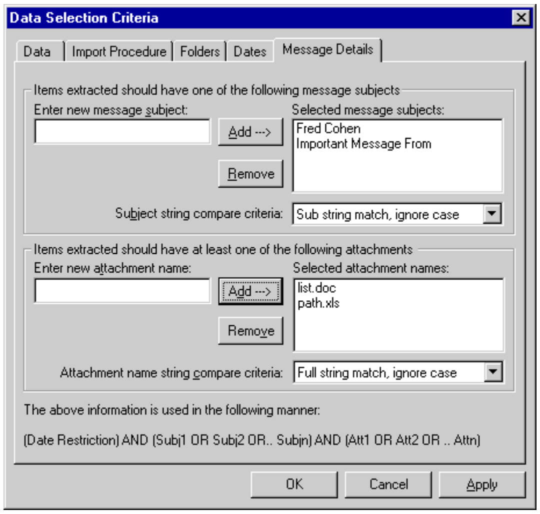
By default, Exchange data files are located in the path **X:\Program files\Exchsrvr\MDBDATA**, where X: is the server’s volume root. But, it’s common for Exchange administrators to move the mail stores to other file paths.

Recovery Storage Groups and ExMerge

Two key things to understand about Microsoft Exchange are that, since 2003, an Exchange feature called **Recovery Storage Group** supports collection of e-mail from the server without

[39] Lawyers and judges intent on distilling the complexity of electronic discovery to rules of thumb are prone to pigeonhole particular ESI as “accessible’ or ‘inaccessible” based on the media on which it resides. In fact, ESI’s storage medium is just one of several considerations that bear on the cost and burden to access, search and produce same. Increasingly, backup tapes are less troublesome to search and access while active data on servers or strewn across many “accessible” systems and devices is a growing challenge.

any need to interrupt its operation or restore data to a separate recovery computer. The second key thing is that Exchange includes a simple utility for exporting the server-stored e-mail of individual custodians to separate PST container files. This utility, officially the Exchange Server Mailbox Merge Wizard but universally called **ExMerge** allows for rudimentary filtering of messages for export, including (right) by message dates, folders, attachments and subject line content.



Items extracted should have one of the following message subjects

Enter new message subject:

Selected message subjects: Fred Cohen
Important Message From

Subject string compare criteria: Sub string match, ignore case

Items extracted should have at least one of the following attachments

Enter new attachment name:

Selected attachment names: list.doc
path.xls

Attachment name string compare criteria: Full string match, ignore case

The above information is used in the following manner:

(Date Restriction) AND (Subj1 OR Subj2 OR.. Subjn) AND (Att1 OR Att2 OR .. Attn)

ExMerge also plays a crucial role in recovering e-mails "double deleted" by users if the Exchange server has been configured to support a "dumpster retention period." When a user deletes an e-mail, it's automatically relegated to a "dumpster" on the Exchange Server. The dumpster holds the message for 30 days by default or until a full backup of your Exchange database is run, whichever comes first. The retention interval can be customized for a longer or shorter interval.

Journaling, Archiving and Transport Rules

Journaling is the practice of copying all e-mail to and from all users or particular users to one or more repositories inaccessible to most users. Journaling serves to preempt ultimate reliance on individual users for litigation preservation and regulatory compliance. Properly implemented, it should be entirely transparent to users and secured in a manner that eliminates the ability to alter the journaled collection.

Exchange Server supports three types of journaling: **Message-only journaling** which does not account for blind carbon copy recipients, recipients from transport forwarding rules, or recipients from distribution group expansions; **Bcc journaling**, which is identical to Message-only journaling except that it captures Bcc addressee data; and **Envelope Journaling** which captures all data about the message, including information about those who received it. Envelope journaling is the mechanism best suited to e-discovery preservation and regulatory compliance.

Journaling should be distinguished from **e-mail archiving**, which may implement only selective, rules-based retention and customarily entails removal of archived items from the server for offline or near-line storage, to minimize strain on IT resources and/or implement electronic records management. However, Exchange journaling also has the ability to implement rules-based storage, so each can conceivably be implemented to play the role of the other.

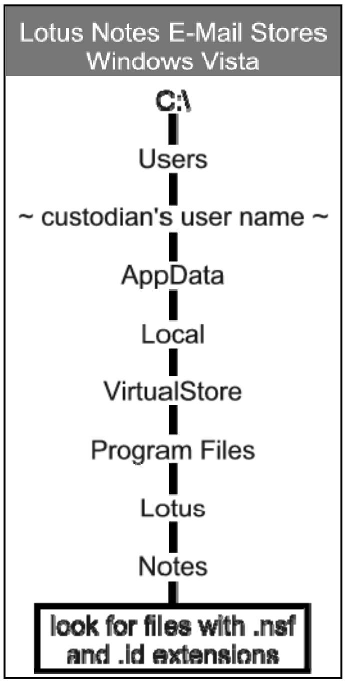
A related concept is the use of **Transport Rules** in Exchange, which serve, *inter alia*, to implement "Chinese Walls" between users or departments within an enterprise who are ethically or legally obligated not to share information, as well as to guard against dissemination of confidential information. In simplest terms, software called *transport rules agents* “listen” to e-mail traffic, compare the content or distribution to a set of rules (conditions, exceptions and actions) and if particular characteristics are present, intercedes to block, route, flag or alter suspect communications.

Lotus Domino Server and Notes Client

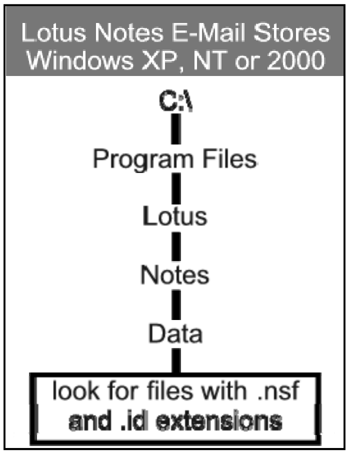
Though Microsoft’s Exchange and Outlook e-mail products have a greater overall market share, IBM’s Lotus Domino and Notes products hold powerful sway within the world’s largest corporations, especially giant manufacturing concerns and multinationals. IBM boasts of 140 million Notes licenses sold to data worldwide.

Lotus Notes can be unhelpfully described as a “cross-platform, secure, distributed document-oriented database and messaging framework and rapid application development environment.” The main takeaway with Notes is that, unlike Microsoft Exchange, which is a purpose-built application designed for messaging and calendaring, Lotus Notes is more like a toolkit for *building* whatever capabilities you need to deal with documents—mail documents, calendaring documents and any other type of document used in business. Notes wasn’t *designed* for e-mail—e-mail just happened to be one of the things it was tasked to do.[40] Notes is database driven and distinguished by its replication and security.

Lotus Notes is all about copies. Notes content, stored in Notes Storage facility or **NSF** files, are constantly being replicated (synchronized) here and there across the network. This guards against data loss and enables data access when the network is unavailable, but it also means that there can be many versions of Notes data stashed in various places within an enterprise. Thus, discoverable Notes mail may not be gone, but lurks within a laptop that hasn’t connected to the network since the last business trip.


Lotus Notes E-Mail Stores
Windows Vista
C:\
Users
~ custodian's user name ~
AppData
Local
VirtualStore
Program Files
Lotus
Notes
look for files with .nsf and .id extensions

By default, local iterations of users’ NSF and ID files will be found on desktops and laptops in the paths shown in the diagrams at right. It’s imperative to collect the user’s .id file along with the .nsf message container or you may find yourself locked out of encrypted content. It’s also important to secure each custodian’s Note’s password. It’s common for Notes to be installed in ways other than the default configuration, so search by


Lotus Notes E-Mail Stores
Windows XP, NT or 2000
C:\
Program Files
Lotus
Notes
Data
look for files with .nsf and .id extensions

[40] Self-anointed “Technical Evangelist,” Jeff Atwood describes Lotus Notes this way: “It is death by a thousand tiny annoyances -- the digital equivalent of being kicked in the groin upon arrival at work every day.” http://blogs.vertigosoftware.com/jatwood/archive/2005/08/11/1366.aspx. In fairness, Lotus Notes has been extensively overhauled since he made that observation.

extension to insure that .nsf and .id files are not also found elsewhere. Also, check the files' last modified date to assess whether the date is consistent with expected last usage. If there is a notable disparity, look carefully for alternate file paths housing later replications.

Local replications play a significant role in e-discovery of Lotus Notes mail because, built on a database and geared to synchronization of data stores, deletion of an e-mail within Lotus "broadcasts" the deletion of the same message system wide. Thus, it's less common to find undeleted iterations of messages in a Lotus environment unless you resort to backup media or find a local iteration that hasn't been synchronized after deletion.

Novell GroupWise

Experienced lawyers—that sound better than "older"--probably remember GroupWise. It originated as a WordPerfect virtual desktop product for messaging and calendaring called "WordPerfect Library," then became "WordPerfect Office." It changed to GroupWise when WordPerfect was acquired in 1993 by another deposed tech titan, Novell. GroupWise is alive (some might say "alive and well") in a handful of niche sectors, particularly government; but GroupWise's market share been so utterly eclipsed by its rivals as to make it seem almost extinct.

GroupWise is another tool thought of as "just an e-mail application" when it's really a Swiss army knife of data management features that happens to do e-mail, too. Because it's not a standalone e-mail server and client and because few vendors and experts have much recent experience with GroupWise, it's presents greater challenges and costs in e-discovery.

GroupWise is built on a family of databases which collectively present data comprising messages to users. That's an important distinction. Messages are less like discrete communications than reports *about* the communication, queried from a series of databases and presented *in the form of* an e-mail. User information is pulled from one database (ofuser), message content emerges from a second (ofmsg) and attachments are managed by a third database (offiles). When a user sends a GroupWise e-mail, the message is created in the user's message database and pointers to that message go to the user's Sent Items folder and the Recipients' Inboxes. Attachments go to the offiles database and pointers to attachments go out. Naturally, a more traditional method must be employed when message are sent beyond the GroupWise environment.

The prevailing practice in dealing with GroupWise e-mail is to convert messages to Outlook PST formats. The sole rationale for this seems to be that most e-discovery service providers are equipped to deal with PSTs and not native GroupWise data. Thus, the decision is driven by ignorance not evidence. Accordingly, a cottage industry has emerged dedicated to converting GroupWise ESI to other formats, but a few vendors tout their ability to work natively with GroupWise data. As often as not, conversion is a costly but harmless hurdle; but recognize that some data won't survive the leap between formats and, in choosing whether to deal with

GroupWise data by conversion, you must assess whether the data sacrificed to the conversion process may be relevant and material.

Webmail

An estimated 1.2 billion people use webmail worldwide.41 Ferris Research puts the number of business e-mail users in 2007 at around 780 million, accounting for some 6 *trillion* non-spam e-mails in sent in 2006. In April 2008, *USA Today*[42] reported the leading webmail providers' market share as:

Microsoft webmail properties:	256.2 million users
Yahoo:	254.6 million users
Google:	91.6 million users
AOL webmail properties:	48.9 million users

Any way you slice it, webmail can't be ignored in e-discovery. Webmail holding discoverable ESI presents legal, technical and practical challenges, but the literature is nearly silent about how to address them.

The first hurdle posed by webmail is the fact that it's stored "in the cloud" and off the company grid. Short of a subpoena or court order, the only legitimate way to access and search employee web mail is with the employee's cooperation, and that's not always forthcoming. Courts nonetheless expect employers to exercise control over employees and insure that relevant, non-privileged webmail isn't lost or forgotten.

One way to assess the potential relevance of webmail is to search server e-mail for webmail traffic. If a custodian's Exchange e-mail reveals that it was the custodian's practice to e-mail business documents to or from personal webmail accounts, the webmail accounts may need to be addressed in legal hold directives and vetted for responsive material.

A second hurdle stems from the difficulty in collecting responsive webmail. How do you integrate webmail content into your review and production system? Where a few pages might be "printed" to searchable Adobe Acrobat PDF formats or paper, larger volumes require a means to dovetail online content and local collections. The most common approach is to employ a POP3 client application to download messages from the webmail account. All of the leading webmail providers support POP3 transfer, and with the user's cooperation, it's simple to configure a clean installation of any of the client applications already discussed to capture online message stores. Before proceeding, the process should be tested against accounts that don't evidence to determine what metadata values may be changed, lost or introduced by POP3 collection.

[41] October 2007 report by technology market research firm The Radicati Group, expected to rise to 1.6 billion by 2011.

[42] http://www.usatoday.com/tech/products/2008-04-15-google-gmail-webmail_N.htm

Keep in mind that webmail content can be fragile compared to server content. Users rarely employ a mechanism to back up webmail messages (other than the POP3 retrieval just discussed) and webmail accounts may purge content automatically after periods of inactivity or when storage limits are exceeded. Further, users tend to delete embarrassing or incriminating content more aggressively on webmail, perhaps because they regard webmail content as personal property or the evanescent nature of account emboldens them to believe spoliation will be harder to detect and prove.

Computer Forensics

Virtually any information that traverses a personal computer or other device has the potential to leave behind content that can be recovered in an examination of the machine or device by a skilled computer forensic examiner. Even container files like Outlook PST or OST files have a propensity to hold a considerable volume of recoverable information long after the user believes such data has been deleted.

Though the scope and methodology of a thorough computer forensic examination for hidden or deleted e-mail is beyond the scope of this paper,[43] readers should be mindful that a computer's operating system or **OS** (e.g., Windows or Vista, Mac or Linux) and installed software (**applications**) generate and store much more information than users realize. Some of this unseen information is **active data** readily accessible to users, but requiring skilled interpretation to be of value in illuminating human behavior. Examples include the data *about* data or **metadata** tracked by the OS and applications, but not displayed onscreen. For example, Microsoft Outlook records the date a Contact is created, but few of us customize the program to display that "date created" information.

Other active data reside in obscure locations or in coded formats less readily accessible to users, but enlightening when interpreted and correlated. Log files, hidden system files and information recorded in non-text formats are examples of **encoded data** that may reveal information about user behavior. As discussed, e-mail attachments and the contents of OST, PST and NSF files are all encoded data.

Finally, there are vast regions of hard drives and other data storage devices that hold **forensic data** even the operating systems and applications can't access. These "data landfills," called **unallocated clusters** and **slack space**, contain much of what a user, application or OS discards over the life of a machine. Accessing and making sense of these vast, unstructured troves demands specialized tools, techniques and skill.

[43] For further reading on computer forensics, see Ball, *Five on Forensics*, http://www.craigball.com/cf.pdf and Ball, *What Judges Should Know About Computer Forensics*, published by the Federal Judicial Center and available at http://www.craigball.com/What_Judges_Computer_Forensics-200807.pdf

Computer forensics is the expert acquisition, interpretation and presentation of the data within these three categories (**Active**, **Encoded** and **Forensic** data), along with its juxtaposition against other available information (e.g., e-mail, phone records and voice mail, credit card transactions, keycard access data, documents and instant message communications).

Most cases require no forensic-level computer examination, so courts and litigants should closely probe whether a request for access to an opponent's machines to recover e-mail is grounded on a genuine need or is simply a fishing expedition. Except in cases involving, e.g., data theft, forgery or spoliation, computer forensics will usually be an effort of last resort for identification and production of e-mail.

The Internet has so broken down barriers between business and personal communications that workplace computers are routinely peppered with personal, privileged and confidential communications, even intimate and sexual content, and home computers normally contain some business content. Further, a hard drive is more like one's office than a file drawer. It may hold data about the full range of a user's daily activity, including private or confidential information about others.

Accordingly, computer forensic examination should be governed by an agreed or court-ordered protocol to protect unwarranted disclosure of privileged and confidential information. Increasingly, courts appoint neutral forensic examiners to serve as Rule 53 Special Masters for the purpose of performing the forensic examination *in camera*. To address privilege concerns, the information developed by the neutral is first tendered to counsel for the party proffering the machines for examination, which party generates a privilege log and produces non-privileged, responsive data.[44]

Whether an expert or court-appointed neutral conducts the examination, the order or agreed protocol granting forensic examination of ESI should provide for handling of confidential and privileged data and narrow the scope of examination by targeting specific objectives. The examiner needs clear direction in terms of relevant keywords and documents, as well as pertinent events, topics, persons and time intervals. A common mistake is for parties to agree upon a search protocol or secure an agreed order without consulting an expert to determine feasibility, complexity or cost.

There is no more a "standard" protocol for forensic examination than there is a "standard" set of deposition questions. In either case, a good examiner tailors the inquiry to the case, follows the evidence as it develops and remains flexible enough to adapt to unanticipated discoveries. Consequently, it is desirable for a court-ordered or agreed protocol to afford the examiner discretion to adapt to the evidence and apply their expertise.

[44] For further discussion of forensic examination protocols, see Ball in Your Court, *Problematic Protocols*, November 2008, Law Technology News; http://www.lawtechnews.com/r5/showkiosk.asp?listing_id=2756144&pub_id=5173&category_id=27902

Why Deleted Doesn't Mean Gone

A computer manages its hard drive in much the same way that a librarian manages a library. The files are the "books" and their location is tracked by an index. But there are two key differentiators between libraries and computer file systems. Computers employ no Dewey decimal system, so electronic "books" can be on any shelf. Further, electronic "books" may be split into chapters, and those chapters stored in multiple locations across the drive. This is called "**fragmentation**." Historically, libraries tracked books by noting their locations on index card in a card catalog. Computers similarly employ directories (often called "**file tables**") to track files and fragmented portions of files.

When a user hits "Delete," nothing happens to the actual file targeted for deletion. Instead, a change is made to the file table that keeps track of the file's location. Thus, akin to tearing up a card in the card catalogue, the file, like its literary counterpart, is still on the "shelf," but now…without a locator in the file table…our file is a needle in a haystack, lost among millions of other unallocated clusters.

To recover the deleted file, a computer forensic examiner employs three principal techniques:

File Carving by Binary Signature

Because most files begin with a unique digital signature identifying the file type, examiners run software that scans each of the millions of unallocated clusters for particular signatures, hoping to find matches. If a matching file signature is found and the original size of the deleted file can be ascertained, the software copies or "carves" out the deleted file. If the size of the deleted file is unknown, the examiner designates how much data to carve out. The carved data is then assigned a new name and the process continues.

Unfortunately, deleted files may be stored in pieces as discussed above, so simply carving out contiguous blocks of fragmented data grabs intervening data having no connection to the deleted file and fails to collect segments for which the directory pointers have been lost. Likewise, when the size of the deleted file isn't known, the size designated for carving may prove too small or large, leaving portions of the original file behind or grabbing unrelated data. Incomplete files and those commingled with unrelated data are generally corrupt and non-functional. Their evidentiary value is also compromised.

File signature carving is frustrated when the first few bytes of a deleted file are overwritten by new data. Much of the deleted file may survive, but the data indicating what type of file it was, and thus enabling its recovery, is gone.

File signature carving requires that each unallocated cluster be searched for each of the file types sought to be recovered. When the parties or a court direct that an examiner "recover all deleted files," that's an exercise that could take weeks, followed by countless hours spent culling corrupted files. Instead, the protocol should, as feasible, specify the *particular* file types of interest (i.e., e-mail and attachments) based upon how the machine's was used and the facts and issues in the case.

File Carving by Remnant Directory Data

In some file systems, residual file directory information revealing the location of deleted files may be strewn across the drive. Forensic software scans the unallocated clusters in search of these lost directories and uses this data to restore deleted files.

Search by Keyword

Where it's known that a deleted file contained certain words or phrases, the remnant data may be found using keyword searching of the unallocated clusters and slack space. Keyword search is a laborious and notoriously inaccurate way to find deleted files, but its use is necessitated in most cases by the enormous volume of ESI. When keywords are not unique or less than about 6 letters long, many false positives ("**noise hits**") are encountered. Examiners must painstakingly look at each hit to assess relevance and then manually carve out responsive data. This process can take days or weeks for a single machine.

Keyword searching for e-mail generally involves looking for strings invariably associated with messages (e.g., e-mail addresses) or words or phrases known or expected to be seen in deleted messages (e.g., subject lines, signatures or header data).

Because e-mail is commonly encoded, encrypted and/or compressed, and because it customarily resides in container files structured more like databases than discrete messages, computer forensic analysis for e-mail recovery is particularly challenging. On the other hand, e-mail tends to lodge in so many places and formats; it's the rare case where at least some responsive e-mail cannot be found.

As relevant, a forensic protocol geared to e-mail should include a thorough search for orphaned message collections, looking for any of the varied formats in which e-mail is stored (e.g., PST, OST, NSF, MSG, EML, MHT, DBX, IDX) and of unallocated clusters for binary signatures of deleted container files. Container files themselves should be subjected to processes that allow for recovery of double deleted messages that remain lodged within uncompacted containers.[45]

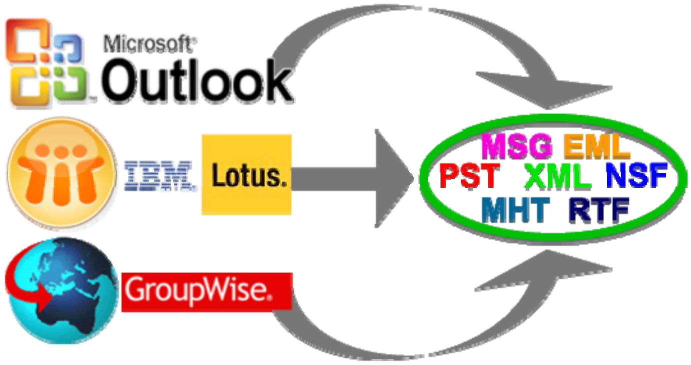
[45] A common technique used on PST containers is to corrupt the file header on a copy of the container file and use Microsoft's free Scanpst utility to repair it. This process sometimes recovers double deleted messages as these remain in the container until periodically compacted by Outlook. Scanpst can also be run against chunks of the unallocated clusters to ferret out deleted PSTs.

Webmail can often be found in the Internet cache (Temporary Internet Files) as well as within unallocated clusters and swap files. Desktop search and indexing programs (like Google Desktop) may also hold the full text of deleted e-mail. Moreover, devices like smart phones and PDAs employ synchronization files to store and transfer e-mail. Finally, e-mail clients like Outlook can themselves hold messages (e.g., corrupted drafts and failed transmissions) along with metadata unseen by users.

Forms of Production

As discussed above, what users see presented onscreen as e-mail is a selective presentation of information from the header, body and attachments of the source message, determined by the capabilities and configuration of their e-mail client and engrafted with metadata supplied by that client. Meeting the obligation to produce comparable data of similar utility to the other side in discovery is no mean feat, and one that hinges on choosing suitable forms of production.

Requesting parties often demand "native production" of e-mail; but, electronic mail is rarely produced natively in the sense of supplying a duplicate of the source container file. That is, few litigants produce the entire Exchange database EDB file to the other side. Even those that produce mail in the format employed natively by the application (e.g., as a PST file) aren't likely to produce the source file but will fashion a reconstituted PST file composed of selected messages deemed responsive and non-privileged.



As applied to e-mail, "native production" instead signifies production in a form or forms that most closely approximate the contents and usability of the source. Often, this will be an form of production identical to the original (e.g., PST or NSF) or a form (like MSG or EML) that shares many of the characteristics of the source and can deliver comparable usability when paired with additional information (e.g., information about folder structures).[46]

Similarly, producing parties employ imaged production and supply TIFF image files of messages, but in order to approximate the usability of the source must also create and produce accompanying load files carrying the metadata and full text of the source message keyed to its images. Collectively, the load files and image data permit recipients with compatible software (e.g., Summation, Concordance) to view and search the messages. Selection of Adobe PDF

[46] When e-mail is produced as individual messages, the folder structure may be lost and with it, important context. Additionally, different container formats support different complements of metadata applicable to the message. For example, a PST container may carry information about whether a message was opened, flagged or linked to a calendar entry.

documents as the form of production allows producing parties to dispense with the load files because much of the same data can be embedded in the PDF. PDF also has the added benefit of not requiring the purchase of review software.

Some producing parties favor imaged production formats in a mistaken belief that they are more secure than native production and out of a desire to emboss Bates numbers or other text (i.e., protective order language) to the face of each image. Imaged productions are more expensive than native or quasi-native productions, but, as they hew closest to the document review mechanisms long employed by law firms, they require little adaption. It remains to be seen if clients will continue to absorb higher costs solely to insulate their counsel from embracing more modern and efficient tools and techniques.

Other possible format choices include XML[47] and MHT,[48] as well as Rich Text Format (RTF)--essentially plain text with improved formatting—and, for small collections, paper printouts.
There is no single, “perfect” form of production for e-mail, though the “best” format to use is the one on which the parties agree. Note also that there’s likely not a single production format that lends itself to *all* forms of ESI. Instead, *hybrid productions* match the form of production to the characteristics of the data being produced. In a hybrid production, images are used where they are most utile or cost-effective and native formats are employed when they offer the best fit or value.

As a rule of thumb to maximize usability of data, hew closest to the format of the source data (i.e., PST for Outlook mail and NSF for Lotus Notes), but keep in mind that whatever form is chosen should be one that the requesting party has the tools and expertise to use.

Though there is no ideal form of production, we can be guided by certain ideals in selecting the forms to employ. Absent agreement between the parties or an order of the Court, the forms of production employed for electronic mail should be either the mail’s native format or a form that will:

1. Enable the complete and faithful reproduction of all information available to the sender and recipients of the message, including layout, bulleting, tabular formats, colors, italics, bolding, underlining, hyperlinks, highlighting, embedded images and other non-textual ways we communicate and accentuate information in e-mail messages.
2. Support accurate electronic searchability of the message text and header data;

[47] XML is eXtensible Markup Language, an unfamiliar name for a familiar technology. Markup languages are coded identifiers paired with text and other information. They can define the appearance of content, like the Reveal Codes screen of Corel Inc.'s WordPerfect documents. They also serve to tag content to distinguish whether 09011957 is a birth date (09/01/1957), a phone number (0-901-1957) or a Bates number. Plus, markup languages allow machines to talk to each other in ways humans understand. For further information about the prospects for XML in e-discovery, see Ball in Your Court, *Trying to Love XML*, March 2008, Law Technology News;
http://www.lawtechnews.com/r5/showkiosk.asp?listing_id=1929884

[48] MHT is a shorthand reference for MHTML or MIME Hypertext markup Language. HTML is the markup language used to create web pages and rich text e-mails. MHT formats mix HTML and encoded MIME data(see prior discussion of MIME at page to represent the header, message body and attachments of an e-mail.

3. Maintain the integrity of the header data (To, From, Cc, Bcc, Subject and Date/Time) as discrete fields to support sorting and searching by these data;
4. Preserve family relationships between messages and attachments;
5. Convey the folder structure/path of the source message;
6. Include message metadata responsive to the requester's legitimate needs;
7. Facilitate redaction of privileged and confidential content and, as feasible, identification and sequencing akin to Bates numbering; and
8. Enable reliable date and time normalization across the messages produced.[49]

Conclusion

By now, you're wishing you'd taken my advice on page one and not begun. It's too late. You know too much about e-mail to ever again trot out the "I dunno" defense.

As I look back over the preceding discussion of the nerdy things that lawyers need to know about e-mail, I'm struck by how much *more* there is to cover. We've barely touched on e-mail backup systems, review platforms, visual analytics, e-mail archival, cloud computing, search and sampling, message conversion tools, unified messaging and a host of other exciting topics.

I hope you've gleaned something useful from this paper. I invite and appreciate your suggestions for corrections and improvements. Please e-mail them to craig@ball.net.

[49] E-mails carry multiple time values depending upon, e.g., whether the message was obtained from the sender or recipient. Moreover, the times seen in an e-mail may be offset according to the time zone settings of the originating or receiving machine as well as for daylight savings time. When e-mail is produced as TIFF images or as text embedded in threads, these offsets may produce hopelessly confusing sequences. For further discussion of date/time normalization to UTC, see Ball in Your Court, *SNAFU*, September 2008, Law Technology News; http://www.lawtechnews.com/r5/showkiosk.asp?listing_id=2217760

Technology Primer: Backups in Civil Discovery

Craig Ball

Technology Primer: Backups in Civil Discovery
By Craig Ball
© 2009

E-discovery lawyers think they know all they need to know about backup: "It's tapes, right? You send 'em to a vendor, you look at what they send back, you bill the client. Simple."

Backup is the Rodney Dangerfield of the e-discovery world. It don't get no respect. Or, maybe it's more like Milton, the sad sack with the red stapler from the movie, *Office Space*. Backup is pretty much ignored...until headquarters burns to the ground or it turns out the old tapes in the basement hold the only copy of the all-important TPS reports demanded in discovery.

Would you be surprised to learn that backup is the hottest, fastest moving area of information technology? Consider the:

- Migration of data to the "cloud" *(Minsk! Why's our data in Minsk?);*
- Explosive growth in hard drive capacities *(Two terabytes! On a desktop?);*
- Ascendency of virtual machines *(Isn't that the title of the next Terminator movie?);* and
- Increased reliance on replication *(D2D2T? That's the cute Star Wars droid, right?).*

If you don't fully understand how backup systems work, you can't reliably assess whether discoverable data exists or how much it will cost in terms of sweat and coin to access, search and recover that data.

The Good and Bad of Backups

Ideally, the contents of a backup system would be entirely cumulative of the active "online" data on the servers, workstations and laptops that make up a network. But because businesses entrust the power to destroy data to every computer user--including those motivated to make evidence disappear—and because companies configure systems to purge electronically stored information as part of records retention programs, backup tapes may be the only evidence containers beyond the reach of those who've failed to preserve evidence and those with an incentive to destroy or fabricate it. Going back as far as Col. Oliver North's deletion of e-mail subject to subpoena in the Iran-Contra affair, it's long been backup systems that ride to truth's rescue with "smoking gun" evidence.

Backup tapes can also be fodder for pointless fishing expeditions mounted without regard for the cost and burden of turning to backup media, or targeted prematurely in discovery, before more accessible data sources have been exhausted.

Grappling with Backup Tapes

Backup tapes are made for ***disaster recovery***, i.e., picking up the pieces of a damaged or corrupted data storage system. Some call backups "snapshots" of data, and like a photo, backup tapes capture only what's in focus. To save time and space, backups typically ignore commercial software programs that can be reinstalled in the event of disaster, so ***full backups*** typically focus on all *user created* data. ***Incremental backups*** grab just what's been created or changed since the last full or incremental backup. Together, they put Humpty-Dumpty back together again in a process called ***tape restoration*.**

Tape is cheap, durable and portable, the last important because backups need to be stored away from the systems at risk. Tape is also slow and cumbersome, downsides discounted because it's so rarely needed for restoration.

Because backup systems have but one legitimate purpose--being the retention of data required to get a business information system "back up" on its feet after disaster--a business only needs recovery data covering a brief interval. No business wants to replicate its systems as they existed six months or even six weeks before a crash. Thus, *in theory*, older tapes are supposed to be recycled by overwriting them in a practice called ***tape rotation*.**

<u>Jargon Watch</u>

disaster recovery
full backup
incremental backup
tape restoration
tape rotation
legacy tapes
replication
drive imaging
backup set
backup catalog
tape log
linear serpentine
helical recording
virtual tape library
D2D2T
RAID
striping
parity
hash value
single-instance storage
non-native restoration

But, as theory and practice are rarely on speaking terms, companies may keep backup tapes long past (sometimes *years* past) their usefulness for disaster recovery and often beyond the companies' ability to access tapes created with obsolete software or hardware. These ***legacy tapes*** are business records—sometimes the last surviving copy—but are afforded little in the way of *records management*. Even businesses that overwrite tapes every two weeks replace their tape sets from time to time as faster, bigger options hit the market. The old tapes are frequently set aside and forgotten in offsite storage or a box in the corner of the computer room.

Like the Delorean in "Back to the Future," legacy tapes allow you to travel back in time. It doesn't take 1.2 million gigawatts of electricity, just lots of cabbage.

Duplication, Replication and Backup

We save data from loss or corruption via one of three broad measures: duplication, replication and backup.

Duplication is the most familiar--protecting the contents of a file by making a copy of the file to another location. If the copy is made to another location on the same medium (e.g., another folder on the hard drive), the risk of corruption or overwriting is reduced. If the copy is made to another medium (another hard drive), the risk of loss due to media failure is reduced. If the copy is made to a distant physical location, the risk of loss due to physical catastrophe is reduced.

You may be saying, “Wait a second. Isn’t backup just a form of duplication?” To some extent, it is, and certainly, it’s the most common “backup” method used on a standalone machine. But true enterprise backup injects other distinctive elements, the foremost being that backups are not user-initiated but occur systematically, untied to the whims and preferences of individual users.

Replication is duplication without discretion. That is, the contents of one storage medium are periodically or continuously mirrored to another storage medium. Replication may be as simple as RAID 1 mirroring of two local hard drives or as elaborate as employing a distant data recovery center ready to roll in the event of a catastrophe.

Unlike duplication and replication, backup involves (reversible) alteration of the data and logging and cataloging of the stored data. Typically, backup entails the use of software or hardware that compresses and encrypts data. Further, backup systems are designed to support iteration, e.g., they manage the scheduling and scope of backup, track the content and timing of backup “sets” and record the allocation of backup volumes across multiple devices or media.

Major Elements of Backup Systems

Understanding backups requires an appreciation of the three major elements of a backup system: the source data, the target data (“backup set”) and the catalog.

1. Source Data (Logical or Physical) Though users tend to think of the source data as a collection of files, backup may instead be drawn from the broader, logical divisions of a storage medium—“partitions,” “volumes” and “folders” in the parlance of hard drive organization. ***Drive imaging,*** a specialized form of backup employed by IT specialists and computer forensic examiners, may draw from below the logical hierarchy of a drive, collecting a “bitstream” of the drive’s contents reflecting the contents of the medium at the physical level. The bitstream of the medium may be stored in a single large file, but more often is broken into manageable, like-sized “chunks” of data to facilitate more flexible storage.

2. Backup Set (Physical or Logical, Full or Changed-File) A ***backup set*** may refer to a *physical* collection of *media* housing backed up data, i.e., the collective group of magnetic tape cartridges required to hold the data, or the “set” may reference the *logical* grouping of *files* (and associated catalog) which collectively comprise the backed up data.

Backup sets further divide between what can be termed “full backups” and “changed-file backups.” As you might expect, full backups tend to copy everything present on the source (or at least “everything” as defined in the full backup set) where changed-file backups duplicate items that have been added or altered since a full backup. The changed-file components further subdivide into incremental backups, differential backups and delta block backups. The first two identify changed files based on either the status of a file's archive bit or a file's created and modified date values. The delta block method examines the contents of a file and stores only the *difference*s between the version of the file contained in the full backup and the modified version. This approach is trickier, but it permits the creation of more compact backup sets and accelerates backup and restoration.

3. Backup Catalog vs. Tape Log Unlike duplication and replication, where generally no record is kept of the files moved or their characteristics, the creation and maintenance of a catalog is a key element of backup. The ***backup catalog*** tracks*, inter alia,* the source and metadata of each file or component of the backup set as well as the location of the element within the set. The catalog delineates the quantity of target media and identifies and sequences each tape or disk required for restoration. Without a catalog setting out the logical organization of the data as stored, it would be impossible to distinguish between files from different sources having the same names or to extract selected files without restoration of all of the backed up data.

Equally important is the catalog's role in facilitating single instance backup of identical files. Multiple computers—especially those within the same company—store many files with identical names, content and metadata. It's a waste of time and resources to backup multiple iterations of identical data, so the backup catalog makes it possible to store just a single instance of such files and employ placeholder "stubs" or pointers to track all locations to which the file should be restored.

Obviously, *lose* the catalog, and it's tough to put Humpty Dumpty back together again.

It's important to distinguish the catalog--a detailed digital record that, if printed, would run to hundreds of pages or more--from the ***tape log,*** which is typically a simple listing of backup events and dates, machines and tape identifier. *See*, *e.g.,* the sample page of a tape log attached as Appendix A.

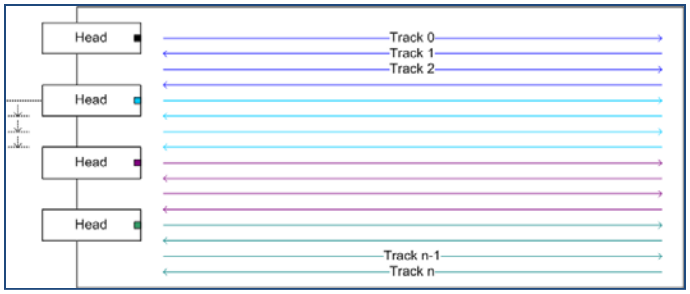
Backup Media: Tape and Disk-to-Disk

Tape Backup

Though backup tape seems almost antique, tape technology has adapted well to modern computing environments. The IBM 3420 reel-to-reel backup tapes that were a computer room staple in the 1970s and '80s employed 240 feet of half-inch tape on 10.5-inch reels. These tapes were divided into 9 tracks of data and held a then-impressive 100 megabytes of information traveling at 1.2 megabytes per second. Today's common LTO-4 tapes are housed in a 4-inch square LTO cartridge less than an inch thick and feature 2600 feet of half-inch tape divided into 896 tracks holding 800 gigabytes of information traveling at 120 megabytes per second.

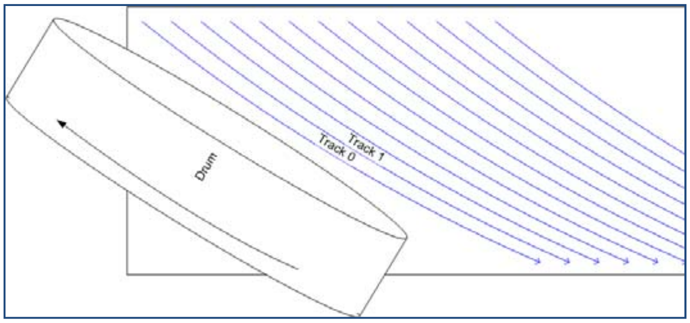
That's 100 times as many tracks, 100 times faster data transfer and *8,000 times greater* data storage capability.

Some readers may recall "auto-reverse" tape transport mechanisms, which eliminated the need to eject and turn over an audiocassette to play the other side. Many modern backup tapes use a scaled-up version of that back-and-forth or ***linear serpentine*** recording scheme. "Linear" because it stores data in parallel tracks running the length of the tape, and "serpentine" because its path snakes back-and-forth like a mountain road.[50] Sixteen of the LTO-4 cartridge's 896 tracks are read or written as the tape moves past the heads, so it takes *56 back-and-forth passes* or "wraps" to read or write the full contents of a single LTO-4 cartridge.



That's about *28 miles* of tape passing the heads!

An alternate recording scheme employed by SAIT-2 tape systems employs a ***helical recording*** system that writes data in parallel tracks running diagonally across the tape, much like a household VCR. Despite a slower transfer rate, helical recording also achieves 800GB of storage capacity on 755 feet of 8mm tape housed in a compact cartridge like that used in handheld video cameras.



Why is Tape So Slow?

Clearly, tape is a pretty remarkable technology that's seen great leaps in speed and capacity.

Still, there are those pesky laws of physics.

All that serpentine shuttling back and forth over 28 miles of tape is a mechanical process. It occurs at a glacial pace relative to the speed with which computer circuits or even hard drives move data.

Further, backup restoration is often an incremental process. Reconstructing reliable data sets may require data from multiple tapes to be combined. Add to the mix the fact

[50] Or, if you prefer, "Serpentine!" like the evasive action to avoid gunfire Peter Falk urges on Alan Arkin in the 1979 screwball comedy, "The In-Laws."

that as hard drive capacities have exploded, tape must store more and more information to keep pace. Gains in performance are offset by growth in volume.

How Long to Restore?

The big Atlanta-based tape house, eMag Solutions, LLC, recently weighed in concerning the difference between the time it *should* take to restore a backup tape considering just its capacity and data transfer rate versus the time it *really* takes considering the following factors that impact restoration:

- Tape format;
- Device interface, i.e., SCSI or fiber channel;
- Compression;
- Device firmware;
- The number of devices sharing the bus;
- The operating system driver for the tape unit;
- Data block size (large blocks fast, small blocks slow);
- File size (with millions of small files, each must be cataloged);
- Processor power and adapter card bus speed;
- Tape condition (retries eat up time);
- Data structure (e.g., big database vs. brick level mailbox accounts);
- Backup methodology (striped data? multi server?).

The following table reflects eMag's reported experience:

Drive Type	Native cartridge capacity	Drive Native Data Transfer Speed[51]	Theoretical Minimum Data Transfer Time	Typical Real World Data Transfer Time
DLT7000	35GB	3MB/sec	3.25 Hrs	6.5 Hrs
DLT8000	40GB	3MB/sec	3.7 Hrs	7.4 Hrs
LTO1	100GB	15MB/sec	1.85 Hrs	4.0 Hrs
LTO2	200GB	35MB/sec	1.6 Hrs	6.0 Hrs
SDLT 220	110GB	11MB/sec	2.8 Hrs	6.0 Hrs
SDLT 320	160GB	16MB/sec	2.8 Hrs	6.0 Hrs

The upshot is that it takes *about twice as long* to restore a tape under real world conditions than the media's stated capacity and transfer rate alone would suggest. Just to generate a catalog for a tape, the tape must be read in its entirety. Consequently, it's not feasible to deliver 3,000 tapes to a vendor on Friday and expect a catalog to be generated by Monday. The *price* to do the work has dropped dramatically, but the *time* to do the work has not.

Common Tape Formats

Here at the close of 2009, the LTO tape format is the clear winner of the tape format wars, having eclipsed all contenders save the disk storage options that now threaten to

[51] " *How Long Does it Take to Restore a Tape*," eMag blog, 7/17/2009 at http://tinyurl.com/tapetime, Some of these transfer rate values are at variance with manufacturer's stated values, but they are reported here as published by eMag.

(finally) extinguish tape as the leading backup medium. The LTO-5 format coming early in 2010 will natively hold 1.5 terabytes of data at a transfer rate of 140 megabytes per second.

But the dusty catacombs beneath Iron Mountain still brim with all manner of legacy tape formats that will be drawn into e-discovery fights for years to come. Here are some of the more common formats seen in the last 25 years and their characteristics:

Name	Format	A/K/A	Length	Width	Capacity (GB)	Transfer Rate (MB/sec)
DLT 2000	DLT3	DLT	1200 ft	1/2”	10	1.25
DLT 2000 XT	DLT3XT	DLT	1828 ft	1/2”	15	1.25
DLT 4000	DLT 4	DLT	1828 ft	1/2”	20	1.5
DLT 7000	DLT 4	DLT	1828 ft	1/2”	35	5
DLT VS-80	DLT 4	TK-88	1828 ft	1/2”	40	3
DLT 8000	DLT 4	DLT	1828 ft	1/2”	40	6
DLT-1	DLT 4	TK-88	1828 ft	1/2”	40	3
DLT VS-160	DLT 4	TK-88	1828 ft	1/2”	80	8
SDLT-220	SDLT 1		1828 ft	1/2”	110	10
DLT V4	DLT 4	TK-88	1828 ft	1/2”	160	10
SDLT-320	SDLT 1		1828 ft	1/2”	160	16
SDLT 600	SDLT 2		2066 ft	1/2”	300	36
DLT-S4	DLT-S4	DLT Sage	2100 ft	1/2”	800	60
DDS-1	DDS-1	DAT	60M	4mm	1.3	.18
DDS-1	DDS-1	DAT	90M	4mm	2.0	.18
DDS-2	DDS-2	DAT	120M	4mm	4	.60
DDS-3	DDS-3	DAT	125M	4mm	12	1.1
DDS-4	DDS-4	DAT	150M	4mm	20	3
DDS-5	DAT72	DAT	170M	4mm	36	3
DDS-6	DAT160	DAT	150M	4mm	80	6.9
M1	AME	Mammoth	22M	8mm	2.5	3
M1	AME	Mammoth	125M	8mm	14	3
M1	AME	Mammoth	170M	8mm	20	3
M2	AME	Mammoth 2	75M	8mm	20	12
M2	AME	Mammoth 2	150M	8mm	40	12
M2	AME	Mammoth 2	225M	8mm	60	12
Redwood	SD3	Redwood	1200 ft	1/2”	10/25/50	11
TR-1		Travan	750 ft	8mm	.40	.25
TR-3		Travan	750 ft	8mm	1.6	.50
TR-4		Travan	740 ft	8mm	4	1.2
TR-5		Travan	740 ft	8mm	10	2.0
TR-7		Travan	750 ft	8mm	20	4.0

Name	Format	A/K/A	Length	Width	Capacity (GB)	Transfer Rate (MB/sec)
AIT 1	AIT		170M	8mm	25	3
AIT 1	AIT		230M	8mm	35	4
AIT 2	AIT		170M	8mm	36	6
AIT 2	AIT		230M	8mm	50	6
AIT 3	AIT		230M	8mm	100	12
AIT 4	AIT		246M	8mm	200	24
AIT 5	AIT		246M	8mm	400	24
Super AIT 1	AIT	SAIT-1	600M	8mm	500	30
Super AIT 2	AIT	SAIT-2	640M	8mm	800	45
3570 B	3570b	IBM Magstar MP		8mm	5	2.2
3570 C	3570c	IBM Magstar MP		8mm	5	7
3570 C	3570c XL	IBM Magstar MP		8mm	7	7
IBM3592	3592	3592	609m	1/2”	300	40
T9840A	Eagle		886 ft	1/2”	20	10
T9840B	Eagle		886 ft	1/2”	20	20
T9840C	Eagle		886 ft	1/2”	40	30
T9940A			2300 ft	1/2”	60	10
T9940B			2300 ft	1/2”	200	30
T10000	T10000	STK Titanium		1/2”	500	120
T10000B	T10000B			1/2”	1000	120
Ultrium	Ultrium	LTO 1	609M	1/2”	100	15
Ultrium	Ultrium	LTO 2	609M	1/2”	200	40
Ultrium	Ultrium	LTO 3	680M	1/2”	400	80
Ultrium	Ultrium	LTO 4	820M	1/2”	800	120

Disk-to-Disk Backup

Tapes are stable, cheap and portable—a natural media for moving data in volumes too great to transmit by wire without consuming excessive bandwidth and disrupting network traffic. But strides in deduplication and compression technologies, joined by drops in hard drive costs and leaps in hard drive capacities, have eroded the advantages of tape-based transfer and storage.

When data sets are deduplicated to unique content and further trimmed by compression, much more data resides in much less drive space. With cheaper, bigger drives flooding the market, hard drive storage capacity has grown to the point that disk backup intervals are on par with the routine rotation intervals of tape systems (e.g., 8-16 weeks), Consequently, disk-to-disk backup options once considered too expensive or disruptive are feasible.

Hard disk arrays can now hold months of disaster recovery data at a cost that competes favorably with tape, Thus, tape is ceasing to be a disaster recovery medium and is instead being used solely for long-term data storage; that is, as a place to migrate disk backups for purposes *other than* disaster recovery, i.e., archival..

Of course, the demise of tape backup has been confidently predicted for years, even while the demand for tape continued to grow. But for the first time, the demand curve for tape has begun to head south.

D2D (for Disk-to-Disk) backup made its appearance wearing the sheep's clothing of tape. In order to offer a simple segue from the 50-year dominance of tape, the first disk arrays were designed to emulate tape drives so that existing software and programmed backup routines needn't change. These are ***virtual tape libraries*** or *VTL*s.

As D2D supplants tape for backup, the need remains for a stable, cheap and portable medium for long-term retention of archival data--the stuff too old to be of value for disaster recovery but comprising the digital annals of the enterprise. This need continues to be met by tape, a practice that has given rise to a new acronym: ***D2D2T,*** for Disk-to-Disk-to-Tape. By design, tape now holds the company's archives, which ensures the continued relevance of tape backup systems to e-discovery.

You can't talk about D2D without mentioning the primary enabling technology that made it possible for hard drive arrays to challenge and best tape on the fields of cost and reliability: RAID.

RAID Technology Enables D2D Backup

The lowest echelon of backup--geared to avoiding failures leading to data loss--is ***fault tolerance,*** typically achieved through redundancy. The most frequently encountered form of redundancy in computer systems, particularly servers, is the use of multiple hard drives configured to work together in a RAID, an acronym for **R**edundant **A**rray of **I**ndependent **D**isks.[52]

Understanding RAID is helpful in selecting cost-effective preservation protocols in e-discovery and when estimating the potential for and cost of computer forensics. For example, knowing that a RAID 1 disk array creates a mirrored duplicate of all data on two separate, identical hard drives might enable you to save a client time, money and business disruption. Instead of hiring an expert to forensically image drives, an in-house IT person might achieve the same end by simply swapping out one of the two drives in the array.

Similarly, it's important to understand the redundancy and performance aspects of RAID in order to judge the potential for forensic examination of the server media.

[52] RAID originally meant **R**edundant **A**rray of ***I****nexpensive* **D**isks, but as RAIDs were often constructed of the most expensive, high-performance SCSI drives on the market, "inexpensive" didn't make much sense.

Although, at first blush, this information seems beyond the pale for legal counsel, it has a decisive impact on costly, consequential decisions made by the legal team.

RAIDs serve two ends: redundancy and performance. The redundancy aspect is obvious—two drives holding identical data safeguard against data loss due to mechanical failure of either drive—but how do multiple drives improve **performance**? The answer lies in splitting the data across more than one drive using a technique called **striping**.

Imagine you stored data on pieces of paper in your pants pocket. Since only one hand can go into the pocket at a time, the rate at which you can retrieve data is limited. But what if you could *divide* the data up between *two* pockets? Since you can now reach into both a left- and right-hand pocket at the same time, the rate at which you can retrieve data doubles. If you were an octopus and had eight hands and pockets…well, you get the idea.

A RAID improves performance by dividing data across more than one physical drive. The data stored on a RAID drive before a same-sized block is stored on the next drive is called the "stripe." By striping data across drives, each drive can deliver data ("reach into a pocket") at the same time, increasing the amount of information handed off to the processor.

But, when you divide information across two or more drives, the failure of any drive creates gaps--so many gaps, in fact, that all of the information may be lost forever. You gain performance, but lose redundancy.

The type of RAID just described is called a **RAID 0** configuration. It's popular among gamers and others trying to wring maximum performance from their systems; but it's so risky, you're unlikely to see it in a business setting.

If RAID 0 is for gamblers, **RAID 1** is ideal for the risk averse. As noted, a RAID 1 completely duplicates everything on one drive to another, so that a failure of one drive won't lead to data loss by mechanical failure. Because a RAID 1 duplicates *everything*, it may duplicate a virus or data corruption as well. Thus, it only protects against drive failure, not bad behavior or user error. Two other downsides of RAID 1 are, it doesn't improve performance and it's expensive to dedicate two hard drives to storing the same information.

So, how do we secure the *performance* of RAID 0 and the *protection* of RAID 1?

You could create what's called a "RAID 0+1" and mirror the two striped drives to two more drives, but then you'd need four hard drives and end up with access to only half of their total storage capacity, Safe and fast, but not cost-efficient. The solution lies in a concept called ***parity,*** key to a range of other sequentially numbered RAID configurations. Of those other configurations, the one you most need to understand is called **RAID 5.**

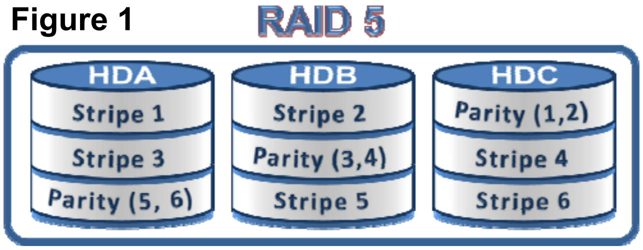
Parity

Consider the simple equation 5 + 2 = 7. If you didn't know one of the three values in this equation, you could easily solve for the missing value, i.e., presented with "5 + __ = 7," you can reliably calculate the missing value is 2. In this example, "7" is the ***parity value*** or checksum for "5" and "2."

The same process is used in many RAID configurations to gain increased performance by striping data across multiple drives while, at the same time, using parity values to permit the calculation of any missing values lost to drive failure. Any one of the three drives can fail, and we can use the remaining two to recreate the third.

Looking at Figure 1, data is striped across three hard drives, A, B and C. Hard Drive C holds the parity values for data stripe 1 on hard drive A and stripe 2 on hard drive B. It's shown as "Parity (1, 2)" in Figure 1. The parity values for the other stripes are distributed on the other drives. Again, any one of the three drives can fail and 100% of the data can be recovered. This configuration is called RAID 5 and, though it requires a minimum of three drives, it can be expanded to dozens of disks.

Figure 1



Essential Technologies: Compression and Deduplication

Along with big, cheap hard drives and RAID redundancy, compression and deduplication have made cost-effective disk-to-disk backup possible. But compression and deduplication are important for tape, too, and bear further mention.

Compression

The design of backup systems is driven by considerations of speed and cost. Perhaps surprisingly, the speed and expense with which an essential system can be brought back online after failure is less critical than the speed and cost of each backup. The reason for this is that (hopefully) failure is a rare occurrence whereas backup is (or should be) frequent and routine. Certainly, no one would seriously contend that restoring a failed system from a morass of magnetic tape is the fastest, cheapest way to rebuild a failed system. No, the advantage of tape is its relatively low cost per gigabyte to store data, not to restore it.

Electrons move much faster than machines. The slowest parts of any backup systems are the mechanical components: the spinning reels, moving heads and the human beings loading and unloading tape transports. One way to maximize the cost advantage and efficiency of tape is to increase the density of data that can be stored per inch of tape. The more you can store per inch, the fewer tapes to be purchased and loaded and the fewer miles of tape to pass by the read-write heads.

Because electrons move speed-of-light faster than mechanical parts of backup systems, a lot of computing power can be devoted to restructuring data in ways that it fits more efficiently on tape or disk. For example, if a horizontal line on a page were composed of one hundred dashes, it takes up less space to describe the line as "100 dashes" or 100(-) than to actually type out 100 dashes. Of course, it would take some time to count the dashes, determine there were precisely 100 of them and ensure the shorthand reference "100 dashes" doesn't conflict with some other part of the text; but, these tasks can be accomplished by digital processors in infinitely less time than that required to spin a reel of tape to store the difference between the data and its shorthand reference.

This is the logic behind data compression; that is, the use of computing power to re-express information in more compact ways to achieve higher transfer rates and consume less storage space. Compression is an essential, ubiquitous technology. Without it, there would be no iPods, Tivos, YouTube, music CDs, DVD movies, digital cameras, Internet radio or pretty web pages.

And without compression, you'd need a whole lot more time, tape and money to backup a computer system.

While compression schemes for files tend to comprise a fairly small number of published protocols (e.g., Zip, LZH), compression algorithms for backup have tended to be proprietary to the backup software or hardware implementing them and to change from version-to-version. Because of this, undertaking the restoration of legacy backup tapes entails more than simply finding a compatible tape drive and determining the order and contents of the tapes. You may also need particular software to decompress the data.

Deduplication

Companies that archive backup tapes may retain years of tapes, numbering in the hundreds or thousands. Because each full backup is a snapshot of a computer system at the time it's created, there is a substantial overlap between backups. An e-mail in a user's Sent Items mailbox may be there for months or years, so every backup replicates that e-mail, and restoration of every backup adds an identical copy to the material to be reviewed. Restoration of a year of monthly backups would generate 12 copies of the same message, thereby wasting reviewers' time, increasing cost and posing a risk of inconsistent treatment of identical evidence (as occurs when one reviewer flags a message as privileged but another decides it's not). The level of duplication between ne backup to the next is often as high as 90%.

Consider, too, how many messages and attachments are dispatched to all employees or members of a product team. Across an enterprise, there's a staggering level of repetition.

Accordingly, an essential element of backup tape restoration is deduplication; that is, using computers to identify and cull identical electronically stored information before review. Deduplicating within a single custodian's mailboxes and documents is called ***vertical deduplication***, and it's a straightforward process. However, corporate backup tapes aren't geared to single users. Instead, business backup tapes hold messages and documents for multiple custodians storing identical messages and documents. Restoration of backup tapes generates duplicates within individual accounts (vertically) and across multiple users (horizontally). Deduplication of messages and documents across multiple custodians is called (not surprisingly) ***horizontal deduplication*****.**

Horizontal deduplication significantly reduces the volume of information to be reviewed and minimizes the potential for inconsistent characterization of identical items; however, it can make it impossible to get an accurate picture of an individual custodian's data collection because many constituent items may be absent, eliminated after being identified as identical to another user's items.

Consequently, deduplication plays two crucial roles when backup sets are used as a data source in e-discovery. First, deduplication must be deployed to eliminate the substantial identicality from one backup iteration to the next; that is, to eliminate that 90% overlap mentioned above. Second, deduplication is useful in reducing the cost and burden of review by eliminating vertical and horizontal repetition within and across custodians.

Modern backup systems are designed to deduplicate ESI *before* it's stored; that is, to eliminate all but a single instance of recurring content, hence the name, *single-instance storage*. Using a method called *in-line deduplication,* a unique digital fingerprint or *hash value* is calculated for each file or data block as it's stored and that hash value is added to a list of stored files. Before being stored, each subsequent file or data block has its hash value checked against the list of stored files. If an identical file has already been stored, the duplicate is not added to the backup media but, instead, a pointer or stub to the duplicate is created. An alternate approach, called *post-process deduplication*, works in a similarly, except that all files are first stored on the backup medium, then analyzed and selectively culled to eliminate duplicates.

Data Restoration

Clearly, data in a backup set is a bit like the furniture at Ikea: It's been taken apart and packed tight for transport and storage. But, when that data is needed for e-discovery--it must be reconstituted and reassembled. It starts to take up a lot of space again. That restored data has to go *somewhere,* usually t*o* a native computing environment just like the one from which it came.

But the system where it came from may be at capacity with new data or not in service anymore. Historically, small and mid-size companies lacked the idle computing capacity to effect restoration without a significant investment in equipment and storage. Larger enterprises devote more stand-by resources to recovery for disaster recovery and may have had alternate environments ready to receive restored data, but those resources had to be at the ready in the event of emergency. It was often unacceptably risky to dedicate them, even briefly, to electronic discovery.

The burden and cost of recreating a restoration platform for backup data was a major reason why backup media came to be emblematic of ESI deemed "not reasonably accessible." But while the inaccessibility presumption endures, newer technology has largely eliminated the need to recreate a native computing environment in order to restore backup tapes. Today, when a lawyer or judge opines that "backups are not reasonably accessible, *per se*," you can be sure they haven't looked at the options in several years.

Non-Native Restoration

A key enabler of low cost access to tapes and other backup media has been the development of software tools and computing environments that support ***non-native restoration****.* Non-native restoration dispenses with the need to locate copies of particular backup software or to recreate the native computing environment from which the backup was obtained. It eliminates the time, cost and aggravation associated with trying to reconstruct a sometimes decades-old system. All major vendors of tape restoration services offer non-native restoration options, and it's even possible to purchase software facilitating in-house restoration of tape backups to non-native environments.

Perhaps the most important progress has been made in the ability of vendors both to generate comprehensive indices of tape contents and extract specific files or file types from backup sets. Consequently, it's often feasible for a vendor to, e.g., acquire just certain types of documents for particular custodians without the need to restore all data in a backup. In some situations, backups are simply not that much harder or costlier to deal with in e-discovery than active data, and they're occasionally the smarter *first* resort in e-discovery.

Going to the Tape *First*?

Perhaps due to the *Zubulake*[53] opinion or the commentary to the 2006 amendments to the Federal Rules of Civil Procedure,[54] e-discovery dogma is that backup tapes are the costly, burdensome recourse of last resort for ESI.

Pity. Sometimes backup tapes are the *easiest, most cost-effective* source of ESI.

For example, if the issue in the case turns on e-mail communications between Don and Elizabeth during the last week of June of 2007, but Don's no longer employed and Elizabeth doesn't keep all her messages, what are you going to do? If these were messages that should have been preserved, you could pursue a forensic examination of Elizabeth's computer (cost: $5,000-$10,000) or collect and search the server accounts and local mail stores of 50 other employees who might have been copied on the missing messages (cost: $25,000-$50,000).

Or, you could go to the backup set for the company's e-mail server from July 1 and recover just Don's or Elizabeth's mail stores (cost: $1,000-$2,500).

The conventional wisdom would be to fight any effort to go to the tapes, but the numbers show that, on the right facts, it's both faster and cheaper to do so.

[53] Zubulake v. UBS Warburg, 217 F.R.D. 309 (S.D.N.Y. 2003

[54] Fed R. Civ. P. 26(b)(2)(B).

Sampling

Sampling backup tapes entails selecting parts of the tape collection deemed most likely to yield responsive information and restoring and searching only those selections before deciding whether to restore more tapes. Sampling backup tapes is like drilling for oil: You identify the best prospects and drill exploratory wells. If you hit dry holes, you pack up and move on. But if a well starts producing, you keep on developing the field.

The size and distribution of the sample hinges on many variables, among them the breadth and organization of the tape collection, relevant dates, fact issues, business units and custodians, resources of the parties and the amount in controversy. Ideally, the parties can agree on a sample size or they can be encouraged to arrive at an agreement through a mediated process.

Because a single backup may span multiple tapes, and because recreation of a full backup may require the contents of one or more incremental or differential backup tapes, sampling of backup tapes should be thought of as the selection of data snapshots at intervals rather than the selection of tapes. Sensible sampling necessitates access to and an understanding of the tape catalog. Understanding the catalog likely requires explanation of both the business system hardware (e.g., What is the SQL Server's purpose?) and the logical arrangement of data on the source machines (e.g., What's stored in the Exchange Data folder?). Parties should take pains to insure that each sample is complete for a selected date or interval; that is, the number of tapes shouldn't be arbitrary but should fairly account for the totality of information captured in a single relevant backup event.

Welcome to the Future

Harvard Law professor Lawrence Lessig recently observed, "We are not going back to the twentieth century. In a decade, a majority of Americans will not even remember what that century was like."[55] Yet, much of what even tech-savvy lawyers understand about enterprise backup systems harkens back to a century ten years gone.

Backup is unlikely to play a large role in e-discovery in the twenty-first century, if only because the offline backup we knew--dedicated to disaster recovery and accreted grandfather-father-son[56]--is fast giving way to data repositories nearly as accessible as our own laptops. The distinction between inaccessible backups and accessible active data stores will soon be just a historical curiosity, like pet rocks or Sarah Palin. Instead, we will turn our attentions to a panoply of electronic archives encompassing tape, disk and "cloud" components. The information we now pull from storage and extract tape-by-tape will simply be available to us--all the time--until someone jumps through hoops to make it go away.

Our challenge won't be in restoring information, but in making sense of it.

[55] Lawrence Lessig, *Against Transparency*, The New Republic, October 9, 2009.

[56] Grandfather-father-son describes the most common rotation scheme for backup media. The last daily "son" backup graduates to "father" status at the end of each week. Weekly "father" backups graduate to "grandfather" status at the end of each month. Grandfather backups are often stored offsite long past their utility for disaster recovery.

Appendix 1: Exemplar Backup Tape Log

Tape No.	Sess. ID	Host Name	Backup Date/Time	Size in Bytes	Session Type
ABC 001	37	EX1	8/1/2007 6:15	50,675,122,176	Exchange 200x
ABC 001	38	EX1	8/1/2007 8:28	337,707,008	System state
ABC 001	39	MGT1	8/1/2007 8:29	6,214,713,344	files incremental or differential
ABC 001	40	MGT1	8/1/2007 8:45	5,576,392,704	SQL Database Backup
ABC 001	41	SQL1	8/1/2007 8:58	10,004,201,472	files incremental or differential
ABC 001	42	SQL1	8/1/2007 9:30	8,268,939,264	SQL Database Backup
ABC 001	43	SQL1	8/1/2007 9:52	272,826,368	System state
ABC 005	2	EX1	8/14/2007 18:30	51,735,363,584	Exchange 200x
ABC 005	3	EX1	8/14/2007 20:35	338,427,904	System state
ABC 005	4	MGT1	8/14/2007 20:38	6,215,368,704	files incremental or differential
ABC 005	5	MGT1	8/14/2007 20:53	5,677,776,896	SQL Database Backup
ABC 005	6	SQL1	8/14/2007 21:06	10,499,260,416	files incremental or differential
ABC 005	7	SQL1	8/14/2007 21:38	8,322,023,424	SQL Database Backup
ABC 005	8	SQL1	8/14/2007 21:57	273,022,976	System state
ABC 002	207	NT1	8/15/2007 20:19	31,051,481,088	loose files
ABC 002	18	NT1	8/16/2007 8:06	47,087,616,000	loose files
ABC 014	9	EX1	8/17/2007 6:45	52,449,443,840	Exchange 200x
ABC 014	10	EX1	8/17/2007 8:53	337,969,152	System state
ABC 014	11	MGT1	8/17/2007 8:54	6,215,368,704	files incremental or differential
ABC 014	12	MGT1	8/17/2007 9:09	5,698,748,416	SQL Database Backup
ABC 014	13	SQL1	8/17/2007 9:22	10,537,009,152	files incremental or differential
ABC 014	14	SQL1	8/17/2007 9:47	8,300,986,368	SQL Database Backup

ABC 014	15	SQL1	8/17/2007 10:08	272,629,760	System state
ABC 003	16	NT1	8/18/2007 6:15	46,850,179,072	loose files
ABC 003	17	NT1	8/18/2007 9:26	44,976,308,224	loose files
ABC 004	19	NT1	8/21/2007 6:16	46,901,690,368	loose files
ABC 004	20	NT1	8/21/2007 9:30	44,742,868,992	loose files
ABC 009	30	EX1	8/22/2007 8:52	53,680,603,136	Exchange 200x
ABC 009	31	EX1	8/22/2007 11:01	348,782,592	System state
ABC 009	32	MGT1	8/22/2007 11:03	6,215,434,240	files incremental or differential
ABC 009	33	MGT1	8/22/2007 11:18	5,715,722,240	SQL Database Backup
ABC 009	34	SQL1	8/22/2007 11:31	10,732,371,968	files incremental or differential
ABC 009	35	SQL1	8/23/2007 4:08	8,362,000,384	SQL Database Backup
ABC 009	36	SQL1	8/23/2007 4:33	272,629,760	System state
ABC 011	44	NT1	8/23/2007 6:16	46,938,193,920	loose files
ABC 011	45	NT1	8/23/2007 9:32	44,611,403,776	loose files

Musings on Electronic Discovery

“Ball in Your Court” May 2006 – January 2010

Selected Columns on Low Cost EDD

The *Law Technology News* column “Ball in Your Court” is both the 2007 and 2008 Gold Medal honoree as “Best Regular Column” as awarded by Trade Association Business Publications International. It’s also the 2009 Gold and the 2007 Silver Medalist honoree of the American Society of Business Publication Editors as “Best Contributed Column” and their 2006 Silver Medalist honoree as “Best Feature Series” and “Best Contributed Column.”

Do-It-Yourself Digital Discovery

by Craig Ball

[Originally published in Law Technology News, May 2006]

Recently, a West Texas firm received a dozen Microsoft Outlook PST files from a client. Like the dog that caught the car, they weren't sure what to do next. Even out on the prairie, they'd heard of online hosting and e-mail analytics, but worried about the cost. They wondered: Did they really *need* an e-discovery vendor? Couldn't they just do it themselves?

As a computer forensic examiner, I blanch at the thought of lawyers harvesting data and processing e-mail in native formats. "Guard the chain of custody," I want to warn. "Don't mess up the metadata! Leave this stuff to the experts!" But the trial lawyer in me wonders how a solo/small firm practitioner in a run-of-the-mill case is supposed to tell a client, "Sorry, the courts are closed to you because you can't afford e-discovery experts."

Most evidence today is electronic, so curtailing discovery of electronic evidence isn't an option, and trying to stick with paper is a dead end. We've got to deal with electronic evidence in small cases, too. Sometimes, that means doing it yourself.

The West Texas lawyers sought a way to access and search the Outlook e-mail and attachments in the PSTs. It had to be quick and easy. It had to protect the integrity of the evidence. And it had to be cheap. They wanted what many lawyers will come to see they need: the tools and techniques to stay in touch with the evidence in smaller cases without working through vendors and experts.

What's a PST?

Microsoft Outlook is the most popular business e-mail and calendaring client, but don't confuse Outlook with Outlook Express, a simpler application bundled with Windows. Outlook Express stores messages in plain text, by folder name, in files with the extension .DBX. Outlook stores local message data, attachments, folder structure and other information in an encrypted, often-massive database file with the extension .PST. Because the PST file structure is complex, proprietary and poorly documented, some programs have trouble interpreting PSTs.

What about Outlook?

Couldn't they just load the files in Outlook and search? Many do just that, but there are compelling reasons why Outlook is the wrong choice for an electronic discovery search and review tool, foremost among them being that it doesn't protect the integrity of the evidence. Outlook changes PST files. Further, Outlook searches are slow, don't include attachments and can't be run across multiple mail accounts. I considered Google Desktop--the free, fast and powerful keyword search tool that makes short work of searching files, e-mail and attachments--but it has limited Boolean search capabilities and doesn't limit searches to specific PSTs.

Non-Starters

I also considered several extraction and search tools, trying to keep the cost under $200.00. One, a gem called Paraben E-Mail Examiner ($199.00), sometimes gets indigestion from PST files and won't search attachments. Another favorite, Aid4Mail Professional from Fookes Software ($49.95), quickly extracts e-mail and attachments and outputs them to several production formats, but Aid4Mail has no search capability. I looked at askSam software

($149.95), but after studying its FAQ and noodling with a demo, askSam proved unable to access any PST except the default profile on the machine—potentially commingling evidence e-mail and the lawyer's own e-mail.

dtSearch

The answer lay with dtSearch Desktop, a $199.00 indexed search application offering a command line tool that extracts the contents of PST files as generic message files (.MSG) indexed by dtSearch. In testing, once I got past the clunky command line syntax, I saved each custodian's mail to separate folders and then had dtSearch index the folders. The interface was wonderfully simple and powerful. Once you select the indices, you can use nearly any combination of Boolean, proximity, fuzzy or synonym searches. Search results are instantaneous and essential metadata for messages and attachments are preserved and presented. It even lets you preview attachments.

dtSearch lacks key features seen in products designed as e-discovery review tools, like the ability to tag hot documents, de-duplicate and redact privileged content. But you can copy selected messages and attachments to folders for production or redaction, preserving folder structures as desired. You can also generate printable search reports showing search results in context. In short, dtSearch works, but as a do-it-yourself e-mail tool, it's best suited to low volume/low budget review efforts.

Wave of the Future?

Any firm handles a fifty-page photocopy job in-house, but a fifty *thousand*-page job is going out to a copy shop. Likewise, e-discovery service providers are essential in bigger cases, but in matters with tight budgets or where the evidence is just e-mail from a handful of custodians, lawyers may need to roll up their sleeves and do it themselves.

Tips for Doing It Yourself

If you'd like to try your hand, dtSearch offers a free 30-day demonstration copy at www.dtsearch.com. Practice on your own e-mail or an old machine before tackling real evidence, and if you anticipate the need for computer forensics, leave the evidence machines alone and bring in an expert.

Whether e-mail is stored locally as a PST, in a similar format called an OST or remotely on an Exchange server depends on the sophistication and configuration of the e-mail system. To find a local PST file on a machine running Windows XP, NT or 2000, look for C:\Documents and Settings*Windows user name*\Local Settings\Application Data\Microsoft\Outlook\Outlook.pst. Archived e-mail resides in another file typically found in the same directory, called Archive.pst. Occasionally, users change default filenames or locations, so you may want to use Windows Search to find all files with a PST extension.

When you locate the PST files, record their metadata; that is, write down the filenames, where you found them, file sizes, and dates they were created, modified and last accessed (right click on the file and select Properties if you don't see this information in the directory). Be sure Outlook's not running and copy the PST files to read-only media like CD-R or DVD-R. Remember that PSTs for *different* custodians tend to have the *same* names (i.e., Outlook.pst and Archive.pst), so use a naming protocol or folder structure to keep track of who's who. When dealing with Outlook Express, search for messages stored in archives with a DBX extension.

Though dtSearch will index DBX files, PSTs must first be converted to individual messages using the included command line tool, mapitool.exe. For DOS veterans, it's old hat, but those new to command line syntax may find it confusing. To use mapitool, you'll need to know the paths to mapitool.exe and to the PSTs you're converting. Then, open a command line window (Start>Run>Command), and follow the instructions included with mapitool.

When mapitool completes the conversion, point the dtSearch Index Manager to the folder holding the extracted messages and index its contents. Name the index to correspond with the custodian and repeat the process for each custodian's PST files.

Rules of Thumb for Forms of ESI Production

by Craig Ball

[Originally published in Law Technology News, July 2006]

Come December 2006, amended Rule 34(b) of the Federal Rules of Civil Procedure has a gift for requesting parties both naughty and nice. It accords them the right to specify the form or forms of production for electronically stored information (ESI) sought in discovery. Though December may seem remote in these dog days of July, litigators better start making their lists and checking them twice to insure that, come December, they'll know what forms are best suited to the most common types of ESI.

Last month, I covered the five principal forms ESI can take:

1. Hard copies;
2. Paper-like images of data in, e.g., TIFF or PDF;
3. Data exported to "reasonably usable" electronic formats like Access databases or load files;
4. Native data; and
5. Hosted data.

This month, we'll look at considerations in selecting a form of production for the kinds of data most often seen in e-discovery.

Word Processed Documents

In small productions (e.g., less than 5,000 pages), paper and paper-like forms (.PDF and .TIFF) remain viable. However, because amended Rule 34(b) contemplates that producing parties not remove or significantly degrade the searchability of ESI, both parties must agree to use printouts and "naked" image files in lieu of electronically searchable forms. When the volume dictates the need for electronic searchability, image formats are inadequate unless they include a searchable data layer or load file; otherwise, hosted or native production (e.g., .DOC, .WPD, .RTF) are the best approaches. Pitfalls in native production include embedded macros and auto date features that alter the document when opened in its native application. Moreover, word processor files can change their appearance and pagination depending upon the fonts installed on, or the printer attached to, the computer used to view the file. Be careful referring to particular pages or paragraphs because the version you see may format differently from the original.

Consider whether system and file metadata are important to the issues in your case. If so, require that original metadata be preserved and a spreadsheet or other log of the original system metadata be produced along with the files.

E-Mail

Again, very small productions may be managed using paper or images if the parties agree on those forms, but as volume grows, only electronically searchable formats suffice. These can take the form of individual e-mails exported to a generic e-mail format (.EML or .MSG files), image files (i.e., .PDF or TIFF) coupled with a data layer or load file, hosted production or native production in one of the major e-mail storage formats (.PST for Outlook, .NSF for Lotus Notes, .DBX for Outlook Express). While native formats provide greatest flexibility and the potential to see far more information than hard copies or images, don't seek native production if you lack the tools and skill to access the native format without corrupting its contents or commingling evidence with other files.

All e-mail includes extensive metadata rarely seen by sender or recipient. This header data contains information about the routing and timing of the e-mail's transmission. Require preservation and production of e-mail metadata when it may impact issues in the case, particularly where there are questions concerning origin, fabrication or alteration of e-mail.

Spreadsheets
Even when spreadsheets fit on standard paper, printed spreadsheets aren't electronically searchable and lack the very thing that separates a spreadsheet from a table: the formulae beneath the cells. If the spreadsheet is just a convenient way to present tabular data, a print out or image may suffice, but if you need to examine the methodology behind calculations or test different theories by changing variables and assumptions, you'll need native file production. Hosted production that allows virtual operation may also suffice. When working with native spreadsheets, be mindful that embedded variables, such as the current date, may update automatically upon opening the file, changing the data you see from that previously seen by others. Also, metadata about use of the spreadsheet may change each time it is loaded into its native application. Once again, decide if metadata is important and require its preservation when appropriate.

PowerPoint Presentations:
You can produce a simple PowerPoint presentation as an electronically searchable image file in PDF or TIFF, but if the presentation is animated, it's a poor candidate for production as an image because animated objects may be invisible or displayed as incomprehensible layers. Instead, native or hosted production is appropriate. Like spreadsheets, native production necessitates preservation of original metadata, which may change by viewing the presentation.

Voice Mail
Often overlooked in e-discovery, voice mail messages and taped conversations (such as recorded broker-client transactions) may be vitally important evidence. As voice mail converges with e-mail in so-called integrated messaging systems, it's increasingly common to see voice mail messages in e-mail boxes. Seek production of voice mail in common sound formats such as .WAV or .MP3, and be certain to obtain voice mail metadata correlated with the audio because information about, e.g., the intended recipient of the voice message or time of its receipt, is typically not a part of the voice message.

Instant Messaging
Instant messaging or IM is similar to e-mail except that exchanges are in real-time and messages generally aren't stored unless the user activates logging or the network captures traffic. IM use in business is growing explosively despite corporate policies discouraging it. In certain regulated environments, notably securities brokerage, the law requires preservation of IM traffic. Still, requests for discovery of IM exchanges are commonly met with the response, "We don't have any;" but because individual users control whether or note to log IM exchanges, a responding party can make no global assertions about the existence of IM threads without examining each user's local machine. Although IM applications use proprietary formats and protocols, most IM traffic easily converts to plain text and can be produced as an ASCII- or word processor-compatible files.

Databases
Enterprises increasingly rely on databases to manage business processes. Responsive evidence may exist only as answers obtained by querying a database. Databases present enormous e-discovery challenges. Specify production of the underlying dataset and application and you'll likely face objections that the request for production is overbroad or intrudes into trade secrets or the privacy rights of third parties. Producing parties may refuse to furnish copies of database applications arguing that doing so violates user licenses. But getting your own license for applications like Oracle or SAP and assembling the hardware needed to run them can be prohibitive.

If you seek the dataset, specify in your request for production the appropriate back up procedure for the database application geared to capture all of the data libraries, templates and configuration files required to load and run the database. If you simply request the data without securing a back up of the entire database environment, you may find yourself missing an essential component. By demanding that data be backed up according to the publisher's recommended methodology, you'll have an easier time restoring that data, but be sure the back up medium you specify is available to the producing party (i.e., don't ask for back up to tape if they don't maintain a tape back up system).

An approach that sometimes works for simpler databases is to request export of records and fields for import to off-the-shelf applications like Microsoft Access or Excel. One common export format is the Comma Separated Variable or CSV file, also called a Comma Delimited File. In a CSV file, each record is a single line and a comma separates each field. Not all databases lend themselves to the use of exported records for analysis, and even those that do may oblige you to jump through hoops or engage an expert.

If you aren't confident the producing party's interrogation of the database, will disgorge responsive data, consider formulating your own queries using the application's query language and structure. For that, you'll need to understand the application or get expert help, e.g., from a former employee of the responding party or by deposing a knowledgeable employee of your opponent to learn the ins-and-outs of structuring a query.

Summer Reading
ESI. CSV. WAV. It's a new language for lawyers, but one in which we must be fluent if we're to comply with amended Rule 26(f)(3) and its requirement that parties discuss forms of production in the pre-discovery meet-and-confer. So, this summer, lay down that Grisham novel in favor of a work that has us all in suspense: *The Rules.*

Do-It-Yourself Forensics

by Craig Ball

[Originally published in Law Technology News, June 2007]

All over America, vendors stand ready to solve the e-discovery problems of big, rich companies. But here's the rub: Most American businesses are small companies that use computers—and along with individual litigants, they're bound by the same preservation obligations as the Fortune 500, including occasionally needing to preserve forensically significant information on computer hard drives. But what if there's simply no money to hire an expert, or your client insists that its own IT people must do the job?

THE D-I-Y CHALLENGE

I challenged myself to come up with forensically sound imaging methods for conventional IDE and SATA hard drives—methods that would be inexpensive, use off-the-shelf and over-the-net tools, yet simple enough for nearly anyone who can safely open the case and remove the drive. In that vein, the safest way to forensically preserve evidence is to employ a qualified computer forensics expert to professionally "image" the drive and authenticate the duplicate. No one is better equipped to prevent problems or resolve them should they arise.

Further, when you open up a computer and start mucking about, plenty can go awry, so practice on a machine that isn't evidence until you feel comfortable with the process.

FORENSICALLY SOUND

When you empty deleted files from your computer's recycle bin, they aren't gone. The operating system simply ceases to track them, freeing the clusters the deleted data occupies for reallocation to new files. Eventually, these unallocated clusters may be reused and their contents overwritten, but until that happens, Microsoft Corp.'s Windows turns a blind eye to them and only recognizes active data. Because Windows only sees active data, it only copies active data. Forensically sound preservation safeguards the entire drive, including the unallocated clusters and the deleted data they hold.

Even lawyers steeped in electronic data discovery confuse active file imaging and forensically sound imaging. You shouldn't. If someone suggests an active data duplicate is forensically sound, set them straight and reserve "forensically sound" to describe only processes preserving all the information on the media.

PRIMUM NON NOCERE

Like medicine, forensic preservation is governed by the credo: "First, do no harm." Methods employed shouldn't alter the evidence by, e.g., changing the contents of files or metadata. But that's not always feasible, and the first method described departs from the forensic ideal.

METHOD 1: THE DRIVE SWAP COMPROMISE

Pulling the plug and locking a computer away is a forensically sound preservation method, but rarely practical. By the same token, imaging programs such as Symantec Corp.'s Ghost (www.ghost.com) or Acronis Inc.'s True Image (www.acronis.com) leave unallocated clusters behind and may alter the source. Our first do-it-yourself approach strikes a balance between practical and perfect by recognizing that users obliged to preserve the contents of unallocated clusters have no use for those contents. They use only active data. So, the first method employs off-the-shelf cloning software to copy just active files from the original evidence drive to a duplicate of equal or greater capacity. The forensic twist is that you preserve the original drive and put the duplicate back into service.

Be sure that the drive you swap has the same size enclosure as the original (typically 2.5 inches for laptops and 3.5 inches for desktops) and that it connects to the computer in the same way, e.g., parallel ATA (a.k.a. "IDE") or Serial ATA. Pull the plug (for laptops, remove the battery too), then open the case to determine the type of drive interface before heading to the store. Buy the proper replacement internal drive in a gigabyte capacity at least as large as the original. Greater capacity is fine.

Accessing a laptop drive can be tricky, so check the manufacturer's website if you're uncertain how to remove and safely handle the drive. Another hurdle: laptops lack cabling to add a second internal drive, so you'll need an adapter to connect the target drive via USB port. A Vantec Thermal Technologies' (www.vantecusa.com) CB-ISATAU2 adapter cable runs about $25 at www.newegg.com, or find other adapters and suppliers by web searching "sata/ide usb adapter."

Follow the software's instructions, but never install the duplication software to the drive you're preserving because that overwrites unallocated clusters. Instead, run the application from a CD, floppy or thumb drive. It's critically important that you don't inadvertently copy the contents of the blank drive onto the original, so check settings, and then check them again before proceeding.

When the imaging completes, label the original drive with the date imaged, name of the user, machine make, model and serial number, and note any inaccuracy in the BIOS clock or calendar. Secure the original drive in an anti-static bag and install the duplicate drive in the machine. Confirm that it boots. The user should see no difference except that the drive offers more storage capacity.

Done right, this method hews close to a forensically sound image, the qualifier being that the cloning software and the operating system may make some (typically inconsequential) alterations to the source drive. The method combines the advantages of Ghosting (speed and ease-of-use) with the desirable end of preserving the original digital evidence with [most] metadata and unallocated clusters intact. Best of all, it employs tools and procedures likely to be familiar to the service techs at your local electronics superstore. Be sure they adhere to the cautions above.

Next month, I'll describe a do-it-yourself approach to *true* forensically sound imaging.

Do-It-Yourself Forensic Preservation (Part II)

by Craig Ball

[Originally published in Law Technology News, July 2007]

How does a non-expert make a forensically sound copy of a hard drive using inexpensive, readily available tools? That's the D-I-Y challenge. Last month, we discussed a nearly perfect way to forensically preserve hard drives that entails swapping the original drive for a Ghosted copy containing just active files.

But when it comes to crucial evidence, nearly perfect doesn't cut it. Last month's method made minor changes to the source evidence, didn't grab unallocated clusters (necessitating we sequester the original drive) and offered no means to validate the outcome.

Because a forensically sound preservation protects all data and metadata along with deleted information in unallocated clusters, think of the Three Commandments of forensically sound preservation as:

1. Don't alter the evidence;
2. Accurately and thoroughly replicate the contents; and
3. Prove the preceding objectives were met.

This month's method employs write blocking to intercept changes, software that preserves every byte and cryptographic hash authentication to validate accuracy.

Write Blocking

Computer forensics experts use devices called "write blockers" to thwart inadvertent alteration of digital evidence, but write blockers aren't sold in stores (only online) and cost from $150-$1,300. Hardware write blocking is best if timetable and budget allow. Manufacturers include Tableau, LLC (www.tableau.com), WiebeTech, LLC (www.wiebetech.com), Intelligent Computer Solutions, Inc. (www.ics-iq.com) and MyKey Technology, Inc. (www.mykeytech.com).

If you're running Windows XP or Vista, you may not need a device to write protect a drive. To hinder data theft, Windows XP Service Pack 2 added support for software write blocking of USB storage devices. A minor tweak to the system registry disables the computer's ability to write to certain devices via USB ports. To make (and reverse) the registry entry, you can download switch files and view instructions explaining how to manually edit the registry at http://www.lawtechnews.com/r5/showkiosk.asp?listing_id=1560974 (the contents of this web link follow on page 50).

You'll also need:

• **Imaging Machine**--a computer running Windows XP with Service Pack 2 and equipped with both USB 2.0 and IEEE 1394 (aka Firewire or i.Link) ports.

• **Forensic Imaging Application**--though forensic software companies charge a pretty penny for their analysis tools, several make full-featured imaging tools freely available. Two fine Windows-compatible tools are Technology Pathway's Pro-Discover Basic Edition (in the Resource Center at http://www.techpathways.com) and AccessData's FTK Imager (http://www.accessdata.com/support/downloads/). I prefer FTK Imager for its simplicity and ability to create images in multiple formats, including the standard Encase E01 format.

• **Target Drive**--a new, shrink-wrapped external hard drive to hold the image. It should be larger in capacity than the drive being imaged and, if using software write blocking, choose a drive that connects by IEEE 1394 Firewire(as USB ports will be write blocked).

• [Software write blocking only] A **USB bridge adapter cable or external USB 2.0 drive enclosure** matching the evidence drive's interface (i.e., Serial ATA or Parallel ATA). Though you'll find drive enclosures at your local computer store, I favor cabling like the Vantec Thermal Technologies' (www.vantecusa.com) CB-ISATAU2 adapter cable because they connect to 2.5", 3.5" and 5.25" IDE and SATA drives and facilitate imaging without removing the drive.

Imaging the Drive

Here is a step-by-step guide:

1. It's important to carefully document the acquisition process. Inspect the evidence machine and note its location, user(s), condition, manufacturer, model and serial number or service tag. Photograph the chassis, ports and peripherals.

2. Disconnect all power to the evidence machine, open its case and locate the hard drive(s). If more than one drive is present, you'll need to image them all. Accessing a laptop drive can be tricky, so check the manufacturer's website if you're uncertain how to safely remove and handle the drive. Take a picture of the drive(s) and cabling. If you can't read the labeling on the face of the drive or comfortably access its cabling, uninstall the drive by disconnecting its data and power cables and removing mounting screws on both sides of the drive or (particularly in Dell machines) by depressing a lever to release the drive carriage.

Handle the drive carefully. Don't squeeze or drop it, and avoid touching the circuit board or connector pins. If using a hardware write blocker, connect it to the evidence drive immediately and leave it in place until imaging is complete and authenticated.

3. Download and install FTK Imager on the imaging machine. If using software write blocking, initiate the registry tweak, reboot and, using a thumb drive or other USB storage device, test to be sure it's working properly.

4. Connect the evidence drive to the imaging machine through the hardware write block device or, if using software write protection, through either the USB drive enclosure or via bridge cable connected to a software write blocked USB port. ***Above all, be sure the evidence drive connects only through a write blocked device or port***.

5. If USB ports are software write blocked, connect the target drive via the IEEE 1394 port. Optionally, connect via USB port if using hardware write blocking.

6. Run FTK Imager, and in accordance with the instructions in the program's help file for creating forensic images, select the write protected evidence drive as the source physical drive, then specify the destination (target) drive, folder and filename for the image. I suggest incorporating the machine identifier or drive serial number in the filename, choosing "E01" as the image type,

accepting the default 650MB image fragment size and opting to compress the image and verify results.

Hash Authentication

Creating a forensically sound compressed image of a sizable hard drive can take hours. FTK Imager will display its progress and estimate time to completion. When complete, the program will display and store a report including two calculated "digital fingerprints" (called MD5 and SHA1 hash values) which uniquely identify the acquired data. These hash values enable you to prove that the evidence and duplicate data are identical. Hash values also establish whether the data was altered after acquisition.

7. When the imaging process is done, label the target drive with the date, the names of the system user(s) and machine identifier. Include the model and serial number of the imaged drive.

8. With the evidence drive disconnected, reconnect power to the evidence machine and boot into the machine's setup screen to note any discrepancy in the BIOS clock or calendar settings. Disconnect power again and re-install the evidence drive, being careful to properly reconnect the drive's power and data cables.

Whether you return the evidence machine to service or lock it up depends on the facts of the case and duties under the law. But once you've secured a forensically sound, authenticated image (along with your notes and photos), you've got a "perfect" duplicate of everything that existed on the machine at the time it was imaged and, going forward, the means to prove that the data preserved is complete and unaltered.

The safest way to forensically preserve digital evidence is to engage a qualified computer forensics expert because no one is better equipped to prevent problems or resolve them should they arise. But when there's no budget for an expert, there's still an affordable way to meet a duty to forensically preserve electronic evidence: ***do-it-yourself***.

Enabling and Disabling USB Write Protection in Microsoft Windows XP P2 and Vista

(This is the target page for the link in the preceding BIYC July 2007 column)

Windows XP machines updated with Service Pack 2 (SP2) acquired the option to enable write protection for removable storage devices connected to the machine via USB. You can still read from the devices, but you can't write to them. In my testing, it works as promised, preventing changes to the data and metadata of external USB hard drives and thumb drives. Though the Windows cache may make it seem that data has been written to the protected device, subsequent examination demonstrated that no changes were actually made. And you can't beat the price: it's free.

Still, software write protection has its ardent detractors (See, e.g., *The Fallacy of Software Write Protection in Computer Forensics*, Menz & Bress 2004), and because there's no outward manifestation that software write blocking is turned on and working, there's none of the reassurance derived from seeing a hardware write blocker play burly bodyguard to an evidence drive. Other downsides are that software write protection requires a geeky registry hack and lacks the selectivity of hardware write blocking. That is, when you implement software write blocking, it locks down all USB ports, including the one you'd hoped to use to connect an external USB hard target drive. Write blocked for one is write blocked for all.

Caveat: Software write protection of the USB ports only works in Windows XP with Service Pack 2 and Windows Vista. It can be implemented only by users with Administrator level privileges on the machine. Failing to disable write blocking may cause the loss of data you seek to store on external USB storage devices.

The Easy Way

To simplify software write protection, you can download a file from http://www.craigball.com/USB-WProtect.zip containing two .REG files that, when run (i.e., double clicked), serve as switches to enable and disable software write protection of the USB ports.

The Geeky Way

If you'd rather make the registry changes manually, here's how:

Caveat: It's prudent to create a system restore point before editing the registry. To do so, click Start > All Programs > Accessories > System Tools > System Restore. Select "Create a restore point," then click "Next." Type a brief description for your restore point (e.g., "Before adding write protection"), then click "Create."

Enabling Write Protection

To block the computer's ability to write to a removable storage device connected to a USB port, begin by calling up a Windows command dialogue box:

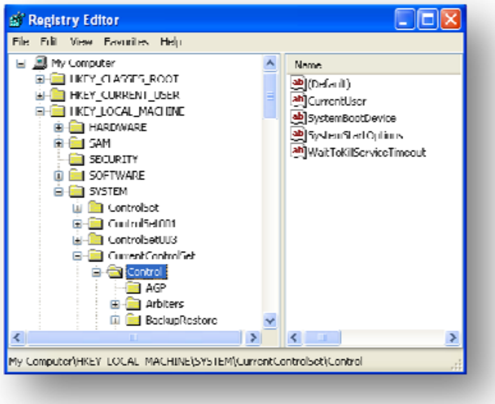
Press the Windows key + R to bring up the Run dialogue box (or click Start > Run).

Type regedit and click “OK” to activate the Windows Registry Editor.

Click the plus sign alongside HKEY_LOCAL_MACHINE, then drill down to SYSTEM\CurrentControlSet\Control. [Fig 1.]

Examine the tree under Control to determine if there is a folder called “StorageDevicePolicies.” If not, you need to create it by right clicking on Control and selecting New > Key. [Fig. 2]

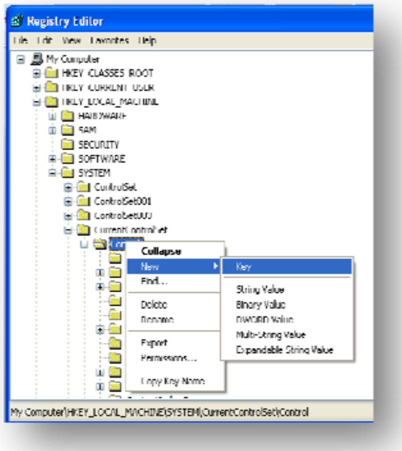
Figure 2



My Computer\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control

Name the key “StorageDevicePolicies,” (All one word. Match capitalization. Omit quotation marks) then right click on the key you’ve just created and select New > DWORD value [Fig. 3]

Figure 3



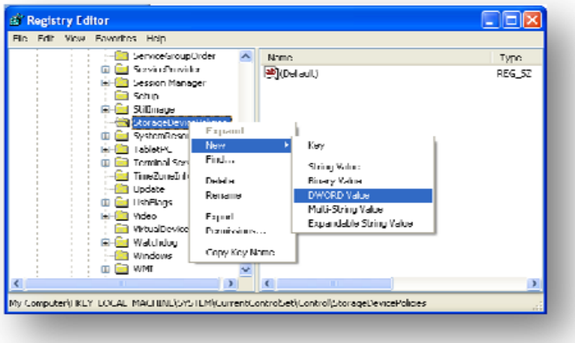
My Computer\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control

Name the new DWORD “WriteProtect” and hit Enter.

Right click on the new DWORD value and select “Modify.” Set the WriteProtect DWORD value to 1. [Fig. 4]

Exit the Registry Editor and reboot the machine. The USB ports should now be write protected.

Figure 3



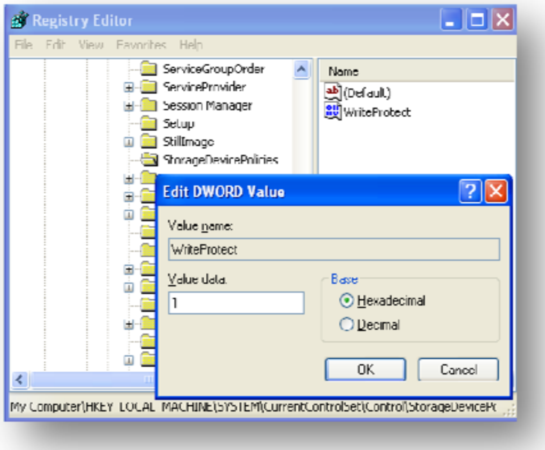
My Computer\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\StorageDevicePolicies

Disabling Write Protection

To restore the system’s ability to write to USB media, navigate to the WriteProtect key as above and either delete it or change its value to 0.

Reminder: WriteProtect = 1 [ON]
WriteProtect = 0 [OFF]

Figure 4



Value name: WriteProtect

Value data: 1

Base
● Hexadecimal
○ Decimal

My Computer\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\StorageDevicePolic

Ask the Right Questions

by Craig Ball

[Originally published in Law Technology News, December 2007]

Sometimes it's more important to ask the right questions than to know the right answers, especially when it comes to nailing down sources of electronically stored information, preservation efforts and plans for production in the FRCP Rule 26(f) conference, the so-called "meet and confer."

The federal bench is deadly serious about meet and confers, and heavy boots have begun to meet recalcitrant behinds when Rule 26(f) encounters are perfunctory, drive-by events. Enlightened judges see that meet and confers must evolve into candid, constructive mind melds if we are to take some of the sting and "gotcha" out of e-discovery. Meet and confer requires intense preparation built on a broad and deep gathering of detailed information about systems, applications, users, issues and actions. An hour or two of hard work should lay behind every minute of a Rule 26(f) conference. Forget "winging it" on charm or bluster, and forget, "We'll get back to you on that."

Here are 50 questions of the sort I think should be hashed out in a Rule 26(f) conference. If you think asking them is challenging, think about what's required to deliver answers you can certify in court. It's going to take considerable arm-twisting by the courts to get lawyers and clients to do this much homework and master a new vocabulary, but, there is no other way.

These 50 aren't all the right questions for you to pose to your opponent, but there's a good chance many of them are . . . and a likelihood you'll be in the hot seat facing them, too.

1. What are the issues in the case?

2. Who are the key players in the case?

3. Who are the persons most knowledgeable about ESI systems?

4. What events and intervals are relevant?

5. When did preservation duties and privileges attach?

6. What data are at greatest risk of alteration or destruction?

7. Are systems slated for replacement or disposal?

8. What steps have been or will be taken to preserve ESI?

9. What third parties hold information that must be preserved, and who will notify them?

10. What data require forensically sound preservation?

11. Are there unique chain-of-custody needs to be met?

12. What metadata are relevant, and how will it be preserved, extracted and produced?

13. What are the data retention policies and practices?

14. What are the backup practices, and what tape archives exist?

15. Are there legacy systems to be addressed?

16. How will the parties handle voice mail, instant messaging and other challenging ESI?

17. Is there a preservation duty going forward, and how will it be met?

18. Is a preservation or protective order needed?

19. What e-mail applications are used currently and in the relevant past?
20. Are personal e-mail accounts and computer systems involved?
21. What principal applications are used in the business, now and in the past?
22. What electronic formats are common, and in what anticipated volumes?
23. Is there a document or messaging archival system?
24. What relevant databases exist?
25. Will paper documents be scanned, at what resolution and with what OCR and metadata?
26. What search techniques will be used to identify responsive or privileged ESI?
27. If keyword searching is contemplated, can the parties agree on keywords?
28. Can supplementary keyword searches be pursued?
29. How will the contents of databases be discovered? Queries? Export? Copies? Access?
30. How will de-duplication be handled, and will data be re-populated for production?
31. What forms of production are offered or sought?
32. Will single- or multi-page .tiffs, PDFs or other image formats be produced?
33. Will load files accompany document images, and how will they be populated?
34. How will the parties approach file naming, unique identification and Bates numbering?
35. Will there be a need for native file production? Quasi-native production?
36. On what media will ESI be delivered? Optical disks? External drives? FTP?
37. How will we handle inadvertent production of privileged ESI?
38. How will we protect trade secrets and other confidential information in the ESI?
39. Do regulatory prohibitions on disclosure, foreign privacy laws or export restrictions apply?
40. How do we resolve questions about printouts before their use in deposition or at trial?
41. How will we handle authentication of native ESI used in deposition or trial?
42. What ESI will be claimed as not reasonably accessible, and on what bases?
43. Who will serve as liaisons or coordinators for each side on ESI issues?
44. Will technical assistants be permitted to communicate directly?
45. Is there a need for an e-discovery special master?
46. Can any costs be shared or shifted by agreement?
47. Can cost savings be realized using shared vendors, repositories or neutral experts?
48. How much time is required to identify, collect, process, review, redact and produce ESI?
49. How can production be structured to accommodate depositions and deadlines?
50. When is the next Rule 26(f) conference (because we need to do this more than once)?

For alternate views on the EDD topics to be addressed at a Rule 26(f) conference, Magistrate Judge Paul Grimm's committee's "Suggested Protocol for Discovery of ESI," (www.mdd.uscourts.gov/news/news/ESIProtocol.pdf), and the U.S.D.C. for the District of

Kansas'"Guidelines for Discovery of Electronically Stored Information" (www.ksd.uscourts.gov/guidelines/electronicdiscoveryguidelines.pdf).

Dealing with Third-Parties

by Craig Ball

[Originally published in Law Technology News, May 2008]

Recently, a team of e-discovery consultants called, seeking feedback on a plan to collect responsive data from non-parties. To their credit, they recognized that not all relevant electronically stored information resides on their client's systems. Contractors, agents, vendors, clients, lawyers, accountants, consultants, experts, outside directors and former employees also hold responsive ESI.

Consequently, parties must factor non-parties (over whom they have influence) into litigation hold and production strategies. The consultants had done so, but now wondered how to retrieve relevant data without compromising its integrity and usability.

They planned to send external hard drives loaded with Microsoft Corp.'s Robocopy backup utility to each non-party custodian, asking them to herd responsive ESI into a single folder, then run Robocopy to replicate and return their collection on the external hard drive. They were proud of their plan, noting that use of Robocopy would preserve system metadata values for the files.

Or would it? Recall that system metadata is data a computer's operating system compiles about a file's name, size and location, as well as its Modified, Accessed and Created (MAC) dates and timestamps.

Don't confuse hardworking *system* metadata with its troublemaker cousin, *application* metadata. The latter is that occasionally embarrassing marginalia embedded in documents, holding user comments and tracked changes.

By contrast, system metadata values are important, helpful dog tag data. They facilitate searching and sorting data chronologically, and shed light on whether evidence can be trusted. System metadata values present little potential for unwitting disclosure of privileged or confidential information and should be routinely preserved and produced.

But Microsoft makes it tough to preserve system metadata. Open a file to gauge its relevance, and you've changed its access date. Copy a file to an external hard drive, and the creation date of the copy becomes the date copied. *Grrrrr!* Robocopy, a free download from Microsoft's website, does a fine job preserving system metadata, but it can't restore data already corrupted.

When I pointed out that copying the files to assemble them would change their MAC dates before Robocopy could preserve them, one of the consultants countered that he'd thought of that already. Each third-party would be instructed to use the Windows "Move" command to aggregate the data.

They'd thought of everything . . . *or had they?*

An advantage of the Move command is that it preserves a file's MAC dates. But, faithful to its name, Move also relocates the file from the place where the third-party keeps it to a new location. So here, it's like requiring those assembling files for production to dump their carefully ordered records into a sack. Demanding non-parties sabotage their filing systems is a non-starter.

To make matters worse, Robocopy is a *command line* application—more like DOS than Windows—employing six dozen switch options, so it's hardly a tool for the faint of heart. Mistype one of these cryptic command line instructions, and the source data's gone forever. Moreover, Robocopy only runs under Windows. What if the data resides on a Mac or Linux machine?

Finally, the approach wasn't geared to collecting e-mail evidence. Sure, they could copy Outlook .pst files holding complete e-mail collections, but non-parties won't agree to share unrelated personal and confidential data. Instead, they'll need to select responsive messages and save them out to a new container file or as individual messages.

Further, if their Exchange e-mail system doesn't support local .pst container files, or if the system uses a different e-mail application like IBM's Lotus Notes or Novell's GroupWise, an entirely different approach is needed.

The well-intentioned consultants were so enamored of their favored "solution," they lost sight of its utter impracticality. Still, they were on the right track seeking low-cost, out-of-the-box approaches to collection—approaches that preserve metadata and don't require technical expertise.

The consultants went back to the drawing board. Their better mousetrap will incorporate input from the other side, an easier-to-implement collection scheme and the use of experts for the most important data.

Sometimes there's no getting around the need to use properly trained personnel and specialized tools; but, if you decide to go a different way, be sure you:

1. Know the systems and applications housing and creating the electronic evidence;

2. Assess the technical capabilities of those tasked to preserve and collect evidence;

3. Understand and thoroughly test collection tools and techniques; and

4. Discuss collection plans with the other side. They may not care about metadata and will accept less exacting approaches.

Tumble to Acrobat as an E-Discovery Tool

by Craig Ball

[Originally published in Law Technology News, June 2008]

When the time comes to turn over e-data uncovered by forensic examination, it's hardly surprising that e-mail makes up a big chunk of the evidence. Notwithstanding its prevalence, e-mail is among the more challenging evidence types to share with clients in ways they can readily review messages and attachments without corrupting the metadata.

I've tried nearly everything, including converting messages to web formats and furnishing a browser-based viewer. That proved easy to run and navigate, but offered no search tools. Imaged formats (e.g., .tiff and .jpg files) also weren't searchable without load files and demanded that my clients have an EDD review platform on hand.

Some lawyers don't have the budget for .tiff conversion and load file generation, let alone a recent copy of Concordance or CT Summation. I've furnished native formats (e.g., .pst or .nsf), quasi-native formats (.eml, .msg) and even Access or NTSearch databases, but there are many pitfalls to trying to review e-mail using desktop applications. And if you need to engage in even the tiniest bit of techno-tinkering it turns lions of the courtroom to jelly. Nothing was quite easy enough.

So, the challenge was to convert e-mail into something I could give to a client with confidence that they could:

1. *Easily open the e-mail evidence on any machine without buying software.*
2. *Search messages quickly and powerfully, with full-text indexing and Boolean support.*
3. *View the messages in a way that faithfully preserves their appearance.*
4. *Print e-mail in a consistent way no matter what printer they used.*
5. *Enjoy document security, authentication and reliable redaction, too.*

While I'm at the wishing well, it would be nice if I could accomplish all this with software I already owned and something that could effortlessly handle the volume of e-mail I come across in computer forensic examinations.

Wouldn't you like to know what wondrous tool fills the bill? *So would I*, because I've yet to find it!

But, the happy news is I got *darn close* to the ideal using the latest version of Adobe System, Inc.'s Acrobat.

Yes, Adobe Acrobat 8.0, that utilitarian tool used to prepare documents for e-filing and keep secrets from sneaking off as Word metadata. Who knew that when this dowdy librarian of a program lets her hair down the results are easy, agile and gorgeous?

Despite a few drawbacks, Acrobat 8.0 turned out to be a nifty way to deliver moderate volumes of e-mail to technophobes and provide a way to search message text with instantaneous results.

Cons:

1. It's slow, taking hours to convert and index about 15,000 messages, even on a fast machine.
2. It refuses to even attempt conversion if you point it at more than 10,000 e-mails. So, for big collections, you must convert the data in chunks and stitch up the results as best you can.
3. Though it retains attachments in native formats with transport messages, the sole attachment type it can search is PDF. Microsoft Corp.'s Outlook, too, has long suffered from an inability to search within attachments. That's a serious shortcoming in both applications, but it's a shortcoming slated to improve in Acrobat's next release.
4. You can redact PDF documents beautifully within Acrobat 8, but not other formats; so, be wary of the potential for privileged data slipping out via an attachment.

Pros:

1. Anyone can review the resulting collection or "PDF Package" on any operating system using the ubiquitous, free Adobe Reader.
2. The search is fast and allows for fine tuning by, inter alia, Boolean operators, stemming, whole word search and case sensitivity.
3. Browsing messages is speedy, and image quality is excellent (screen or printed).
4. It supports annotation and book marking, so it's not a bad review platform for the price.
5. The Acrobat interface is instantly familiar and unintimidating.

To give credit where it's due, I was pointed in the right direction by Rick Borstein, an Adobe business development manager. He's the perfect public face for Adobe because he loves the product and enjoys teasing out its hidden joys without overselling its virtues. He has a fine blog called Acrobat for Legal Professionals (http://blogs.adobe.com/acrolaw/).

Unfortunately, you can't simply point Acrobat to an e-mail container and convert it. Acrobat must run as a PDF Creator toolbar within Outlook or Lotus Notes. The e-mail container must be in either .pst or .nsf format and must be accessible via Outlook or Notes. You can set up a dummy user account for conversion to prevent mixing your mail with evidence mail—a big no-no. The hurdle the first time I used Acrobat for e-mail production was that the evidence e-mail was in Eudora, so I had to apply another tool, **Aid4Mail** from Fookes Software, to convert the Eudora mail to an Outlook-compatible .pst format. This was easy, and the nifty Aid4Mail program costs less than $25, so it paid for itself on first usage.

Adobe Acrobat holds enormous promise as an EDD processing and review platform in smaller cases. It's not all it can or will be, but each new version brings us closer to the goal of effective, affordable electronic discovery for everyone.

SNAFU

by Craig Ball

[Originally published in Law Technology News, September 2008]

On September 2, 1945, my father was ordered to fashion nine impregnable containers to carry the just signed Japanese surrender documents to the President of the United States, the King of England and other heads of state. Dad earned his law degree from Harvard in 1932; so naturally, the Navy made him a gunnery officer.

Good thing, because I can't imagine there's much Lt. Commander Herbert Ball took from Langdell Hall that equipped him to convert five-inch powder charge casings into watertight containers. His ingenuity helped the important V-mail (Victory mail) make it to Mr. Truman, safe and sound.

I proudly share this family lore because a very different war requires me to deconstruct electronic containers carrying missives from the front. Safe in my lab, thousands of miles from IEDs and insurrection, I'm grappling with wacky date values on thousands of e-mail messages from Iraq. It brings to mind that wonderful WWII acronym: SNAFU, for "Situation Normal: All Fouled Up," though no sailor ever said "fouled."

When e-mails originate around the globe on servers from Basra to the Bronx, they seem to travel back in time. Replies precede by hours the messages they answer. Such is the discontinuity between the languorous rotation of the earth and the near light speed of e-mail transmission. A message sent from Baghdad at dinner arrives in Austin before lunch. E-mail client applications dutifully—some might say stupidly—report the time of origin. The confusion grows when receiving machines apply different time zone and daylight savings time biases. It gets even more fouled up when a user in Iraq sends mail via a stateside server. In the end, it's tough to figure out who said what when.

What's needed is time and date normalization; that is, all dates and times expressed in a single consistent way called UTC for Temps Universel Coordonné or Coordinated Universal Time. It's a fraction of a second off the better known Greenwich Mean Time (GMT) and identical to Zulu time in military and aviation circles. Why UTC instead of TUC or CUT? It's a diplomatic compromise, for neither French nor English speakers were willing to concede the acronym. Peace in our time.

My mission was to convert all messages to UTC, changing Situation Normal: All Fouled Up into Situation Normalized: All Fixed Up.

This requires going deeper than the date and time information displayed by Microsoft Corp. Outlook, down to the header data in the message source. There you find a time-stamped listing of servers that handed off the message and the message's time of receipt, expressed in hours plus or minus UTC.

Of course, you've got to have header data to use header data. But when e-mail is produced as .tiff or PDF images, header data is stripped away. The time seen could indicate the time at the place of origin or at the place of receipt. It could reflect daylight savings time … or not.

Absent header data or the configuration of the receiving machine, you just don't know. So reasonably usable production necessitates a supplemental source for the UTC values and offsets (such as a spreadsheet, slip sheet or load file); otherwise, messages should be reproduced in a native or quasi-native format (e.g., .pst, .msg or .eml).

If you're the party gathering and producing e-mail from different time zones, make it a standard part of your electronically stored information collection protocol to establish and preserve the relevant UTC and daylight savings time offsets for the custodial machines. On Microsoft Windows devices, this data can be readily ascertained by clicking on the clock in the System Tray. It can also be gleaned by examination of the System Registry hives if the boot drive was preserved in a forensically sound fashion.

E-mail threads pose additional challenges because erroneous time values may be embedded in the thread. It's important that production include not only the threaded messages, but also each of the constituent messages in the thread.

Don't underestimate the importance of date and time normalization when the timing of events and notices may prove key issues. In a flat world, or one at war, keeping communications on a common clock is a necessity.

Tell Ol' Yahoo, Let my e-Mail Go

by Craig Ball

[Originally published in Law Technology News, September 2009]

A voice came from on high and said unto me, "Go forth and harvest the clouds." Well, not a *voce in excelsis* exactly, but a court order directing I gather up parties' webmail. The task seemed simple enough: The litigants would surrender their login credentials, and I'd collect and process their messages for relevance while segregating for privilege review.

Their data lived "in the cloud," and considering its celestial situation, I might have taken a cue from Ecclesiastes 11:4: "Whoever looks at the clouds shall not reap." So it was, I nearly got smote--not by Yahweh but by Yahoo!

Cloud computing refers to web-based tools and resources that supplant local applications and storage. It's called "the cloud" because of the cloud-shaped icon used to signify the Internet in network schematics.

Cloud computing lets companies avoid capital expenditure for hardware and software. Instead, they scale up or down by renting "virtual machines" as needed, connecting to them via the Internet. Cloud computing also encompasses Software as a Service (SaaS), where users “lease” programs via the Internet--think Google Apps or SalesForce.com--along with the much-touted Web 2.0--a catchall for Internet-enabled phenomena like social networking, blogs, wikis, Twitter, YouTube and arguably any web-centric venture that survived the dot-com apocalypse.

Such cloud-based services aren't new--my e-mail's been in the cloud for five years and twice that for my calendar. But cloud computing is big news in today's economy as companies great and small seek savings by migrating data services to the ether. For the rest of us, accessing and searching our e-mail from anywhere, coupled with near-limitless free storage, makes webmail irresistible. No surprise, then, that Yahoo! Mail's estimated 260 million users make it the largest e-mail service in the world. Add Hotmail and Gmail, and we're talking half a billion webmail users!

The silver lining for e-discovery is that all those candid, probative revelations once the exclusive province of e-mail now flood social media like FaceBook and Twitter. But cloud computing poses e-discovery challenges of near-Biblical proportions because it’s harder to access, isolate and search ESI without physical dominion over the data. Moreover, repatriation of cloud content depends on the compatibility of cloud formats with local storage formats and tools, including the ability to preserve and produce relevant metadata.

Consider the unique way Gmail threads messages into conversations. How do you replicate that structure in the processing and presentation of ESI? You can say, "We don't care about structure;" but increasingly, the *arrangement* of information is vital to full comprehension of the information. Such meta-information is key to a witness' ability to identify and authenticate evidence, especially when it's culled from collaborative environments like virtual deal rooms and Microsoft Corporation's popular SharePoint products.

Crafting protocols to reliably collect ESI from the cloud isn't tomorrow's problem. Today, it’s the rare e-discovery scenario that doesn’t involve webmail, and the court appointing me demanded action now.

I wasn't about to employ Yahoo! Mail's rudimentary search tools to tackle tens of thousands of messages and attachments. I needed a local collection amenable to indexing, search and de-duplication.

Yahoo! Mail lets users download messages and attachments using the common Post Office Protocol (POP), but only from the Inbox folder! *Thou shalt not download from Sent items, custom folders or Drafts.*

I'd either have to forgo multitudes of messages or find a workaround that would make Yahoo! let my e-mail go. I investigated third-party applications like Zimbra and YPOPS that claim to download from beyond the Inbox and tried them without success.

The workaround I devised required multiple steps and careful accounting. The initial set-up involved three steps:

1. I created a pristine user account in a local e-mail client to receive the messages. This can be done using Microsoft Outlook, but I turned to something every Windows user already owns: "Windows Live Mail."
2. I next downloaded the entire contents of the user's Yahoo! Mail Inbox to the Windows Live Mail Inbox, checking to be certain that message counts matched.
3. Then, I created a Live Mail folder called "Hold Inbox" and moved the downloaded messages to it. I did the same thing on the Yahoo! Mail side; that is, created a folder to temporarily hold the contents of the Inbox, then relocated those contents.

Now, the Inboxes were empty and available to serve as conduits to transfer the contents of other folders. In turn, I moved each folder's contents to the empty Yahoo! Mail Inbox, downloaded those items to the local Live Mail Inbox and shifted them to a like-named counterpart folder. After I'd captured all the folders of interest, I replaced the temporarily relocated Inbox contents on both sides and deleted the "Hold Inbox" folders.

Finally, I had a local counterpart of the Yahoo! Mail collection complete with matching folder structure. Using Live Mail, I could even export it as an Outlook PST for processing. Handled with care, the user should see no change to their Yahoo! Mail. But if you try this, be sure that the collecting POP client is set to leave messages on the server and that any Yahoo! Mail that arrives during the collection process makes its way to the local and Yahoo! Mail Inboxes.

This process worked, but it felt like that riddle where the man with the rowboat has to get a duck, a fox and a bag of corn across a river, transporting only one at a time. It's a reminder to consider more than cost savings alone when making the jump to cloud computing. It pays to know how much control you're ceding and how quickly and easily you can harvest your data, for "He that reapeth receiveth wages." [John 4:36]. Amen to that!

E-Discovery Bill of Rights
by Craig Ball

[Originally published in Law Technology News, January 2010]

There's a move afoot to revamp the e-discovery rules. When it comes to electronic evidence, some want to strip the comma from the mandate that litigation be "just, speedy and inexpensive."

Dig beneath the efforts to "reform" e-discovery, and you'll find familiar corporate interests dedicated to closing the courthouse doors. Their rallying cry: "Let's do things as we've always done them." Even trial lawyers, erstwhile champions of discovery rights, are so cowed and confused by e-discovery, they're ready to trade the cow for magic beans enabling them to dodge the hard and humbling task of acquiring new skills.

True, there's waste and inefficiency in e-discovery, largely driven by fear and ignorance. Requesting parties are struggling to adapt, and their demands for the moon and stars would be silly if they weren't so serious.

But requesting parties have rights. If there were a Bill of Rights protecting parties seeking electronic discovery, it might read like this:

I am a requesting party in discovery. I have rights. I am entitled to:

1. Production of responsive ESI in the format in which it's kept in the usual course of business. A producing party's fear of alteration, desire to affix Bates numbers or preference for TIFF images doesn't trump my right to receive the evidence in its native or near-native form.
2. Clear and specific identification of any intentional alteration of ESI made in the discovery process. If, e.g., a producing party omits attachments or redacts content or metadata, the producing party must promptly disclose the alteration with sufficient detail to permit me to assess whether such action was warranted.
3. Production of relevant metadata when I can promptly and specifically identify the metadata fields sought and articulate a reasonable basis for the production.
4. Discover the methodology employed to either select ESI for production or cull ESI from production whenever the method employed was automated, i.e., something other than manual review for responsiveness. This includes disclosure of the relevant capabilities and limitations of electronic search and indexing tools employed to produce or exclude ESI.
5. A detailed explanation of costs when a producing party asserts cost as a basis to resist e-discovery.
6. Put my technical advisor in direct communication with a knowledgeable counterpart for the producing party when technical issues arise, with reasonable and appropriate limits to protect legitimate privilege or confidentiality concerns.
7. Assume a producing party is preserving ESI that I specifically requested be preserved absent timely notice to the contrary.

8. Rely on the use of an iterative approach to electronic search, whereby the production of ESI from an initial search and review informs at least one further electronic search effort.
9. Adequate preservation and complete production, both in proportion to the amount in controversy and importance of the matters at issue.
10. Competence, candor and cooperation from producing party's counsel and support personnel commensurate with the competence, candor and cooperation extended by my counsel and support personnel.

These rights come coupled with duties. Requesting parties have a parity obligation to learn this new craft, work cooperatively and let relevance and reasonableness bound their actions.

I am a requesting party in discovery. I have duties. I am obliged to:

1. Anticipate the nature, form and volume of the ESI under scrutiny and tailor my requests to minimize the burden and cost of securing the information I seek.
2. Clearly and promptly communicate my expectations as to the forms of ESI and fields of metadata sought and be prepared to articulate why I need a specified form of production or field of metadata.
3. Work cooperatively with the producing party to identify reasonable and effective means to reduce the cost and burden of discovery, including, as appropriate, the use of tiering, sampling, testing and iterative techniques, along with alternatives to manual review and keyword search.
4. Know the tools I expect to use for review and processing of ESI produced to me and whether those tools are suited to the forms of ESI sought.
5. Work cooperatively with the producing party to minimize the burden of preservation and to agree promptly to release from a preservation obligation any sources that do not appear likely to hold responsive ESI.
6. Accommodate requests to produce ESI in alternative forms when such requests won't materially impair my ability to access relevant information or use the material produced.
7. Accede to reasonable requests for clawback and confidentiality agreements or orders when to do so won't materially impair my rights or those of others similarly situated.
8. Direct requests for production first to the most accessible sources, and to consider responsive information produced and available to me in framing subsequent requests for production.
9. Make available a competent technical advisor to communicate directly with a knowledgeable counterpart for the producing party concerning technical issues and accommodate reasonable and appropriate limits to protect legitimate privilege or confidentiality concerns.
10. Employ counsel and support personnel who possess a level of e-discovery competence, candor and cooperation commensurate with the competence, candor and cooperation I expect from producing party's counsel and support personnel.

James Madison, author of the U.S. Bill of Rights, wrote, “Knowledge will forever govern ignorance; and a people who mean to be their own governors must arm themselves with the power which knowledge gives.” It takes years to learn the law and become an able litigator. It will take time for lawyers to arm themselves with the novel skills e-discovery requires. There are no shortcuts, and none to be found by “reforming” that which is not yet fully formed in support of ignorance.

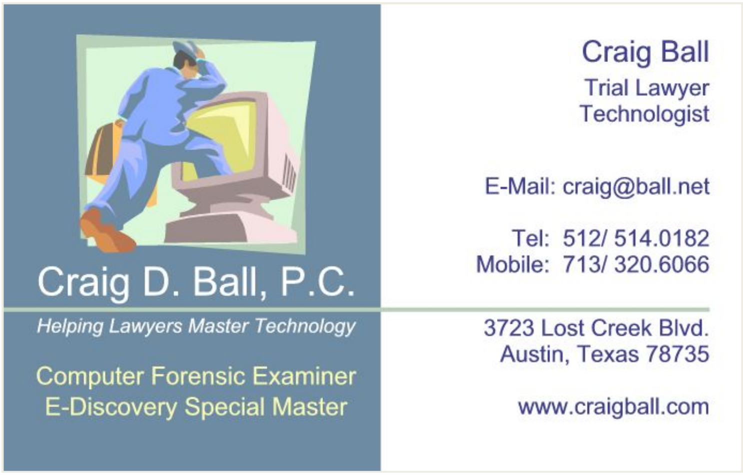
About the Author

Craig Ball, of Austin is a Board Certified Texas trial lawyer and an accredited computer forensics expert, who's dedicated his career to teaching the bench and bar about forensic technology and trial tactics. Craig hung up his trial lawyer spurs to till the soils of justice as a court-appointed special master and consultant in electronic evidence, as well as publishing and lecturing on computer forensics, emerging technologies, digital persuasion and electronic discovery. Fortunate to supervise, consult on or serve as Special Master in connection with some of the world's largest electronic discovery projects and most prominent cases, Craig Ball was named "Consultant of the Year" on December 11, 2009 in the LTN Technology Awards, Craig also greatly values his role as an instructor in computer forensics and electronic evidence to the Department of Justice and other law enforcement and security agencies.

Craig Ball is a prolific contributor to continuing legal and professional education programs throughout the United States, having delivered over 600 presentations and papers. Craig's articles on forensic technology and electronic discovery frequently appear in the national media, including in American Bar Association, ATLA and American Lawyer Media print and online publications. He also writes a multi-award winning monthly column on computer forensics and e-discovery for Law Technology News and Law.com called "Ball in your Court." Rated AV by Martindale Hubbell and named as one of the Best Lawyers in America and a Texas Superlawyer, Craig is a recipient of the Presidents' Award, the State Bar of Texas' most esteemed recognition of service to the profession.

Craig's been married to a trial lawyer for 22 years. He and Diana have two delightful teenagers and share a passion for world travel, cruising and computing.

Undergraduate Education: Rice University, triple major, 1979
Law School: University of Texas, 1982 with honors



Craig D. Ball, P.C.
Helping Lawyers Master Technology
Computer Forensic Examiner
E-Discovery Special Master

Craig Ball
Trial Lawyer
Technologist

E-Mail: craig@ball.net

Tel: 512/ 514.0182
Mobile: 713/ 320.6066

3723 Lost Creek Blvd.
Austin, Texas 78735

www.craigball.com