

## Let's get EGGcited about ENF

Wade Peterson, [www.enfdiscovery.com](http://www.enfdiscovery.com)

September 30, 2018

In one sense, the field of ESI and E-Discovery has seen a tremendous change the past few years. The use of TAR, CAR, Analytics, and even A.I. has taken a solid foothold in people who live with processing documents on a daily basis. Our collections are getting larger. The data types we handle are getting more complex. The use of the Internet, cell phones, portable devices, audio, video, and social media is exploding; and dealing with that in today's environment of E-Discovery workflows is causing an enormous strain on systems and technologies that were, frankly, designed decades ago. We ingest terabytes instead of megabytes. We deal with what I term "3-dimensional documents" instead of 2-dimensional documents. We face (and fight with) new file formats daily. And, yet, turnaround times from collection, to processing, to review, to production seems to be shortening. What to do?

Let's take a look at that 3-dimensional document concept mentioned above. When's the last time you didn't have a URL in a Word document (e.g., even this one does)? An image inside your document? How about a portion of an Excel table embedded inside a Word document? A PowerPoint with a graph, created in Excel, and embedded into a slide? A video inside a PPT? An email with several attachments, which could even be nested emails of other emails with other attachments? Not to mention color, and what file format isn't in color these days? Documents today are rich and 3-dimensional.

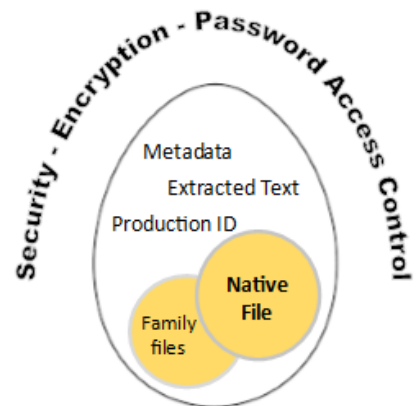
In olden times, we had documents; we printed them out; put them in banker boxes; then scanned them back into the document review system. Two-dimensional documents are flat images. Called TIFF images. They have no dimension to them beyond black and white, grey scale, 300dpi, Group IV. It's a format designed by Aldus (now part of Adobe) in 1986. **It's 2018 and we're living with a technology that is 3 decades old!** Why?

TIFF productions have been the de facto standard production format for as long as I can remember (and yes, I go back in my career to earlier than 1986). PDF productions were introduced, I think, primarily because some recipients of TIFF productions didn't know what to do with them. If they didn't have a document review tool that could accept "load files" + single page TIFF images + metadata OPT files + OCR or native file extracted text files; they were out of luck to view the production. They said give me PDFs. Or, more recently, native files.

But, we have concerns with native file productions. They're not secure. Someone might alter the original file. They may not have the program to view the native file. They can't be endorsed (darn that old mechanical bates stamper). They just haven't been universally adopted.

What if there was a way to solve these problems? There is. Although it's a bit like the old chicken and egg problem. What came first, the chicken or the egg. Let's look at my egg for a minute, which I call the "Encapsulated Native File". Or, ENF for short.

An ENF file contains all the elements for each file in a normal document production. But, it does it in a single file. A single egg, if you will. We take the native file, the metadata, the extracted text, associated family attachments, some additional identifier information (production name, date, originator, etc.), add a dash of security and password functional access control, and we scramble it all together in a single container, much like an eggshell contains the outer membrane, the egg white, and the yoke.



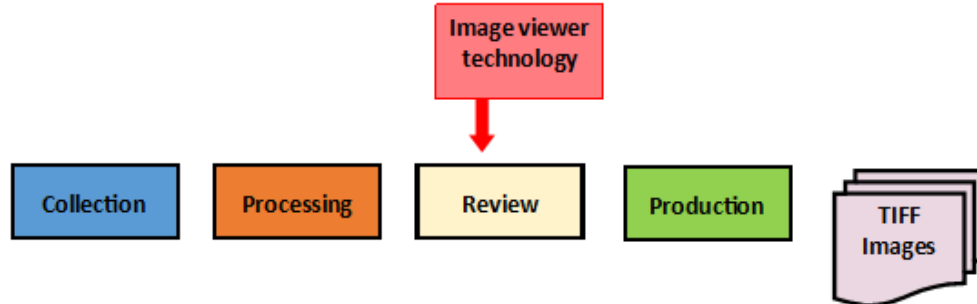
The ENF file is encrypted, is self-contained, has built-in access controls, and contains the original native file(s). Can it be easily altered? No, it's got MD5 hash values to ensure integrity and double layers of encryption.

This is not a theoretical concept. The architecture of ENF has been designed and implemented. A tool to create ENFs was developed to illustrate its feasibility. The tool is called *makeENF*. The tool scans a collection of documents, identifies available metadata, allows the user to configure the metadata fields to produce, adds one or more permission-based passwords, encrypts the file and internal structure, assigns electronic bates numbers, and produces an ENF production. Could we optionally implement blockchain technology into the ENF architecture to digitally validate productions – yes, it's possible.

What's interesting about *makeENF* is the speed. Since no manual intervention is required to transform those 3-dimensional documents into 2-dimensional renditions (e.g., "print ready format"); native files are produced in a hyperloop fashion into ENF files. Can anyone spell "Cost Savings" here? In a typical TIFF production, we take a perfectly good, rich, Word document that is say 100 pages long. We TIFF it and produce it. The result? A set of 100 individual single page TIFF images + a typical OPT image document break file + a typical DAT metadata file + 1 (or 100 if single-page rather than document level) extracted/OCR text files. One file = 103 files. Is it a wonder our data volume is growing? In the ENF world, 1 native file = 1 ENF file. And, that TIFF production is black and white, grainy, and at times almost impossible to render into a flat rendition.

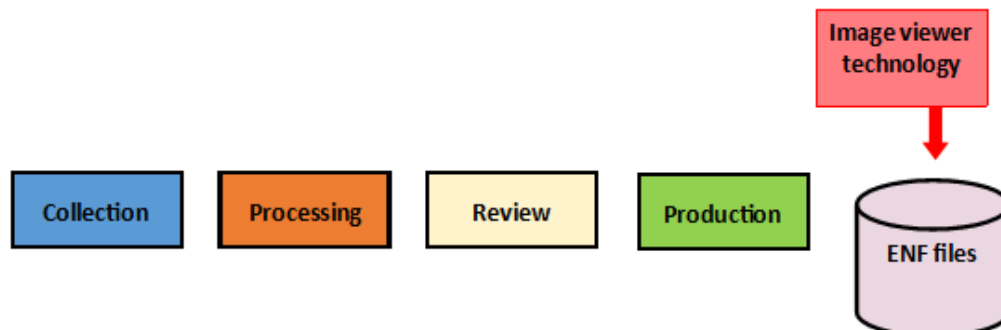
So, we get this basket of ENF files, and how do we view them. How do we work with them? How do we crack them open and look at the stuff inside? Let's discuss that for a bit.

Virtually all document review systems contain some type of viewer technology to view native files. Be it, QuickView Plus, or Oracle's Outside In, or even Microsoft's Office Web viewer. They all contain a method to render the native file. Here's a typical workflow:



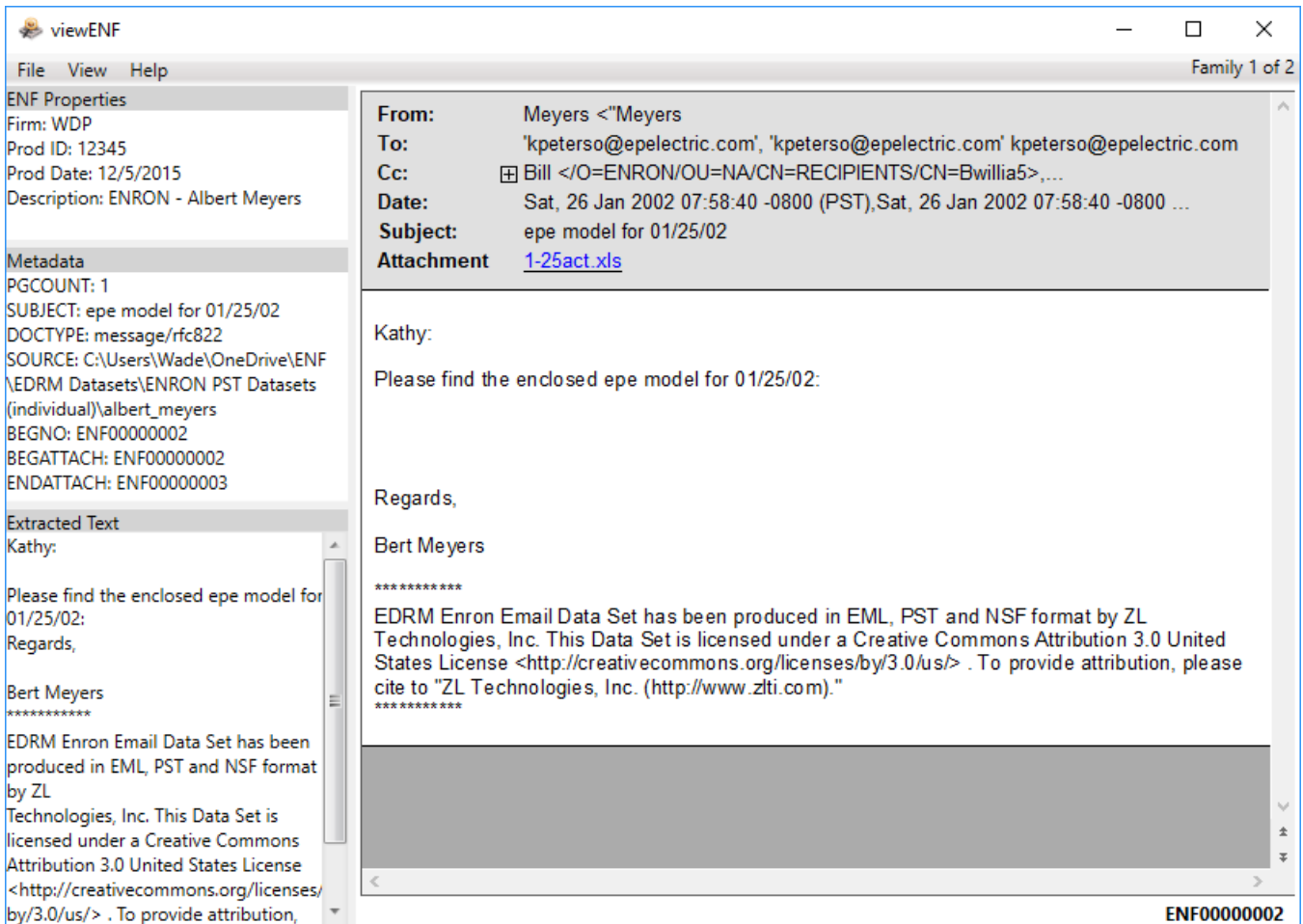
But, what if we shifted the viewer technology into the production output phase? In essence, that is what we do with PDF productions since the resulting production must be viewed with Adobe Reader or other PDF viewer application.

In the ENF world, ENF files can be viewed with a special *viewENF* application. Essentially, *viewENF* is the Adobe Reader for the E-Discovery field. ENFs could also easily be ingested into other document review tools.

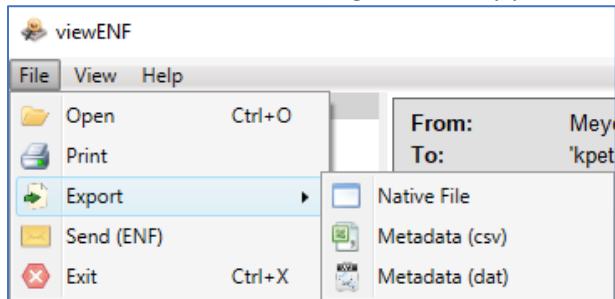


The *viewENF* application is real and not a theoretical illusion. Because *viewENF* uses the same viewer technology as the document review tool, the production looks exactly like it did to the reviewer. Here's the *viewENF* interface:

- Panes on the left show
  - Properties of the ENF file (e.g., firm, prod ID, prod date, etc.)
  - Metadata fields produced
  - Extracted or OCR text
- The viewer window on the right, uses the same viewer technology as the review platform, to render the native file.
- Since there is one attachment to this email (i.e., the Excel file) the toolbar shows this is a family of documents, showing the first of two.
- An electronic bates number shows on the bottom footer of the viewer pane. Endorsements would show on the bottom left.



When opening the ENF file, I'm asked for a permissions password. Based on the permissions, I'm given the ability to view the metadata, view the extracted text, and have the ability to export the information as a native file, a DAT file, or a CSV of the metadata fields. Other permission level password capabilities can easily be added to the architecture of ENF (e.g., the ability print the document with a bates number and/or endorsements).



We've seen TIFF productions have historically been the choice for E-Discovery productions. That's a technology dating back to the era of fax machines, the late 80's, and slow analog phone lines which is what faxes were designed for.

The E-Discovery field needs a new standard. Their own standard. For production formats. ENF can be that standard. Many fields have their own unique file formats. DICOM for radiology. PDF designed for the print industry. RAW for photography. Microsoft DOC, XLS, and PPT. Why shouldn't E-Discovery have its own file standard?

ENF abandon's 2-dimensional print page renderings. ENF abandons the slow process of transforming a perfectly good file into a poor rendition of the original. ENF exponentially speeds up, and therefore substantially reduces the cost of, productions. ENF is a flexible architecture that can expand to meet the ever-demanding needs of the legal industry, new data types, and digital authority validation.

To go beyond the prototyping stage, which is where I am with ENF, we need to consider it for adoption as the E-Discovery production standard. We need to adopt it as the E-Discovery production file format. Only when it is an adopted standard will there be movement to get it integrated into the tools the professionals use daily – the collection tools, the processing tools, the review tools, the production tools. ENF is an architecture. It is a design. While prototyping tools such as *makeENF* and *viewENF* have been created, they are not commercially available. Rather, the goal of ENF is "Advancing the Art of E-Discovery".

*Wade Peterson, CEDS is a technology professional with a career spanning over 45 years in both IT and legal; having done pioneering work with introducing personal computers into vendor and law firms, having one of the first Apple IIe computers, designing one of the earliest computer-based citation checking tools (WestCheck), and owning and donating his LISA computer to a museum in Boulder. He continues to create solutions to address issues in the E-Discovery field.*