BILETA

17th BILETA Annual Conference

April 5th - 6th, 2002.
Free University, Amsterdam.

Using Computers to Manage Large Litigation Document Populations Effectively: A Primer

Marc S. Mayerson[1]

Technology pervades our daily lives, and in litigation it is no different. Just as other businesses have labored over the past decade to harness the power technology has to offer, so too lawyers now are applying computer-based information-management systems to how we litigate cases, particularly those involving large document populations, more widely than ever before.

Increasingly lawyers and law firms must learn to compete in terms of how we apply technology to the management of litigation - and especially the discovery-document population - because so much of the cost of "big time" litigation concerns the management and control of large litigation-related source-document populations and because so much pre-trial effort is focused on "discovery gotcha" salvos directed to wounding an adverse party's fundamental credibility - if not its entire case - before the court. In this article, I discuss the tools available to large-case litigators today and the related information-management issues that are part of effective system design both to control a discovery population and to extract information from it; I review the nuts and bolts of designing a state-of-the-art litigation support system and provide some glimpses of the future. This is an important time for legal technology as old ways that only recently have been automated are yielding to emergent strategies that, harnessing newer technology, reenvision the discovery-management process.

I. The Development of Modern Discovery-Document Litigation-Support Systems

Because of the recency of the deployment of computers to manage discovery, one still hears that in the "old days" everyone got along just fine without imaging documents (that is, capturing discovery documents as virtual photographs in a computer-retrievable manner) or coding them (creating a database index of the document population). Such comments are non sequiturs because the same techniques of document control and management developed back then still are applied in modern computerized litigation-support systems, but they have been automated to increase their speed and flexibility.

No one litigated cases without knowing the documents, so one always needed a pathway into the document population. This used to be done through manual indexing performed by junior lawyers or paralegals. Most manual systems were effectively library card catalogs: an index card or record was created for each document identifying bibliographic details, such as document-control number, author, recipient, date, subject matter, and the like. From the master index, sub-indices could be created by witness or subject or time period.

These manual index systems did not permit word searches across the card catalog or afford ready sorting of the information captured on the index in chronological order, by source location, or according to other criteria. Consequently, the first applications of computers in major litigation were databases that put these manually generated indices "on line" in a searchable, sortable fashion. The card-catalog metaphor was continued such that each index card was superceded by a database "record" and each entry on the index card was now a "field" in the database.

These first databases were introduced in the 1970s using mainframe ("big iron") computers. The entities that prepared the database sometimes were subsidiaries of corporate defendants created for this purpose (captive vendors) or they were outside litigation-support vendors, either single-purpose business entities or service bureaus within larger business-services providers. Litigation-support vendors established "coding shops" that created the index ("coded the documents"). Typically, vendor coding was based on "objective," decontextualized information apparent from the face of the document: author, date, recipient, and the like. Objective coding permitted vendors to hire lower-skilled and lower-wage workers (including off-shore workers), and utilizing temporary employees vendors could devote unlimited resources to the task of coding a particular population, with a project turnaround that would be far faster than what lawfirms could do and at lower per-unit cost. This Taylorist model reserved unto the lawfirms the task of "subjectively" coding the documents, that is, categorizing the documents into subjects pertinent to the litigation, such as particular legal issues, factual topics, or qualititative categories such as "hot" or privileged.

With the advent of the personal computer in the early 1980s, the cost of computer resources plummeted, and many lawfirms bought such computer-based indexing functions in-house to some extent, using their paralegals and junior lawyers to write indices on paper that were then keyed into computers by data-processing personnel or using paralegals and file clerks to key the bibliographic information directly into computers utilizing forms created using early database and spreadsheet programs such as d-Base, R-Base, Lotus 1-2-3 and many others.

At the same time that document indexing was being computerized, the volume of documents managed in large-scale litigation increased or at least seemed to increase - along with the concomitant cost and burden of storing folder after folder of documents. The growing ocean of paper spurred the use of alternate means of storing the physical mass of documents, typically by using microfilm or micro- or ultra-fiche. Micrographic storage of discovery documents certainly was used before the 1980s, but the 1980s and early 1990s saw improved price points and improved micrographic technology for document capture and retrieval.

By the early 1990s, the two technologies - computer databases and "photographic" capture of the source documents stored in a non-paper medium - began to merge with the introduction of imaging technology, which permits computer-based retrieval of images of documents.[2] Unlike micrographics, which require specialized viewing equipment, digitized images appear instantly on a computer user's screen as virtual photographs.[3] In contrast, using an imaging system virtually eliminates these costs. By substituting capital for labor, significant cost savings are achieved because the retrieval costs are built into the computer system and decrease to virtually zero (on a marginal basis). Moreover, imaging the source documents in the first place is no more expensive than is photocopying. Generally, when original source documents are photocopied, a working copy and a archive copy of the source document will be created; this ensures that no document is lost through the manual process of retrieval and refiling. The cost of photocopying plus two "blowbacks" approaches the cost of imaging the same documents in the first place. Imaging holds one further advantage, which is that it reduces the storage costs of documents from cost per square foot of floor space to the cost per megabyte of storing the image. When initially adopted for use by law firms, the electronically "bulky" image files were typically stored on removable magnetic media (*e.g.*, Bernouilli cartridges) or optical-storage media (*e.g.*, 12-inch WORM platters (Write Once Read Many)). Later, the proliferation of CD-ROM technology (a form of WORM) combined with the development of CD jukeboxes holding 100, 150 or more CDs led to more widespread adoption of imaged-enabled litigation support systems. Today, as the cost of magnetic storage (hard drives) decreases, images formerly stored on CDs now are being copied to the hard drives, which increases response time and permits nearly instant, simultaneous access to the document population.

In the current state of the art, lawyers use combined databases and image-retrieval systems. With such systems, a lawyer can, for example, search for all documents written by a witness and on his or her computer monitor view each page of those documents. Typically, a database user (a lawyer or paralegal) will enter a search using the field structure; if the database captures names of authors, for example, the user may search for all documents authored by that person within a given date range (assuming dates also were captured). The user will be presented with a "hit list," which lists in table format the database records containing the search terms. From the hit list, the lawyer can quickly tag which of this population he or she wants to save or print to a witness binder, for example. Through a "drill down" process, the user can review the hit list and select an individual hit to view the corresponding database record, which lists the information abstracted from the document specified in the design of that particular database. If based on the database record the document appears to be something that should be looked at by the lawyer or paralegal, he or she can immediately call up the image of the document itself to review.

In a networked environment, current systems permit each reviewer to comment about each document - or image/page - in a manner that is available to other members of the team. This is a key value-added feature of computer-based litigation support systems and is typically done in one of three ways. A lawyer can place an electronic sticky note recording the comment on the image itself, which is then viewable by others working on the case; alternately, the corresponding database may contain an editable field for lawyer comments; or the database may have predefined categories that the lawyer can flag. The first two methods - sticky notes and comment fields - are dynamic and free form; the latter - an issue or "subjective" field - is static. There are advantages and disadvantages from each in terms of information management and design; consequently, most sophisticated databases today utilize more than one, even all, of these methods of facilitating the transmission and memorialization of the information carried collectively by the litigation team.

A sticky note can show, for example, that a particular document relates to the July 16 meeting, though nothing about its bibliographic data so indicates. It happens that the particular reviewer has other information, such as from a deposition, that she then records using a sticky note. When someone else then retrieves the same document (using whatever search criteria), he sees the sticky note left by the first reviewer and learns that the document relates to the July 16 meeting. In this manner, the knowledge carried in the head of each person can be memorialized and communicated. (Of course, this same information could be saved in a comment field within the database program as opposed to a sticky-note overlay on the image.) One of the information-design questions is where in the system to memorialize such information, that is, is it more likely that the second document reviewer will see the information if it is a sticky note on the image or a field within the database? A further complication is whether these comments should be made searchable and if so whether a common vocabulary should be imposed to facilitate retrieval.

I. Designing Litigation-Support Systems

Designing a litigation-support systems involves a trade-off between the effort put into design and information capture versus the information-management payoff that such effort - and cost - entails. In most instances, database customization affects design issues at the margins (since the core design of litigation databases is relatively homogenous), but compared with an off-the-shelf model a tailored, well-designed database can prove much more efficient, powerful, and cost-effective. This section reviews key elements in designing a litigation-support system for a document-discovery population.

A. *Document Types and Levels of Treatment*

In a discovery-document collection, typically there are any number of *types* of documents - board minutes, sales invoice, memos, letters, reports and the like. Some types of documents - as a class - are more likely to contain relevant information than are other types. Well-crafted litigation support databases consequently handle each of these document types in a somewhat different fashion; in other words, while all documents of a single type are handled identically, not all documents are treated in the same way. The design construct that guides this differential treatment is known as "levels of treatment."

Design of a coding structure starts with the identification of document types. The typology is the key organizing tool for the manner in which the population is to be coded. A document "type" should be a category of easy and objectively ascertainable categories, and a typical database may contain 15 document types (give or take). Examples of document types include those indicated above (meeting minutes, invoices, memos, letters, reports) plus news releases, financial, correspondence, specialized distinctive forms, calendars - any categorical description of material in the discovery population that may provide a basis for differentiating the likely relative importance of different types of documents. One key in fashioning a document typology is to recognize that a document coder or document "prepper"[4] - usually someone with no knowledge of the particular case - ordinarily is the person who determines to which document type a certain piece of paper belongs. This person needs to categorize the document solely by eyeballing it - that is, without regard to the content of the particular document. The correct classification of particular pieces of paper into "type" is important because all documents of a particular type usually are handled or "treated" in a given, uniform manner.[5]

There may be as many as four (or more) levels of treatment utilized in attacking the different document types within a discovery-document population: (1) bulk, (2) bibliographic, (3) bibliographic plus names and organizations in text, and (4) bibliographic plus key words in text (and sometimes names and organizations too).[6] For any given "doc type," the requirement that each data point be captured based on a visual inspection of the document without regard to its content constrains the granularity of the data-collection effort.

The choice of level of treatment for a given document type should be guided by balancing four factors: (1) the relative likelihood that documents of a certain type contain relevant information; (2) the number of ways the relevant documents need to be coded to facilitate their retrieval by document reviewers; (3) the increased burden on the reviewer and dollar cost to the client from retrieving in a search documents of little or no value; and most significantly (4) the very substantial increased costs associated with various ways of refining the sieve versus the consequent information-management-and-retrieval payoff. Differently treating the population by document type both (i) saves money while increasing the usability of the database and (ii) strengthens the claim that the database as a whole is work product (and thus protected from discovery) because the manner in which each document was handled reflects the lawyer's judgment as to the relative importance of subsets of the discovery-document population.

B. *Coding Structure*

The coding structure specifies the category of information from the document to be extracted and memorialized in a pre-determined database field; it consequently defines the pathways into the document population. Using the database fields, users retrieve, sort and view the documents that have been imaged.

For all levels of treatment, because porting physical documents to a computer through imaging (or to film through micrographics) omits staples, paper clips, and the like, *the physical relationship* of the source population - attachment ranges, file-box locations, document breaks, and the like - should be preserved in the litigation-support system. This can be done at scan time by indicating the beginning and end of each document, plus capturing which documents are physically attached to others, such as a cover memo for a report or news clipping ("physical unitization"); unitization can occur during the coding process by determining what logically appears to be an attachment or "child" document and what is the "parent" document ("logical unitization"). Such physical relationships should be recorded in the database whether captured at scan time by relying on the physical document breaks and attachments or at coding time though a logical analysis (or preferably both).[7] Similarly, through the use of specialized coding sheets used at scan time, information about the actual date of the document and document types can be captured. Again, such limited data collection is no substitute for full-blown coding, but it may be cost-effective to capture some information at scan time, and the information captured provides an immediate pathway into the population. Similarly, the database records should include other information about the document captured in the document-acquisition process such as source locations for documents (an individual's file cabinets, document warehouses, plant sites, and the like) and any other pertinent trail information that correlates the image to its source original.

Bulk Coding. Bulk coding as its name suggests usually does not involve coding each individual document within the document type. Instead, a drawer-full of travel receipts, for example, may be coded in bulk, such that all the 1993 receipts are grouped together in the database as a single record. In other words, although the receipts may run to 500 pages, only a single database record is created for all the 500 pages (which may comprise 500 separate documents). The reason for this economy is that it is unlikely that one will need to access and retrieve each of the receipts; in the event that it is necessary to do so, one is able to retrieve and review, in a brute-force manner, each of the receipts for 1993 (or once can send out that group for further coding).[8] Bulk coding keeps the database free of clutter, so users do not receive an unreasonable number of "hits" when performing a search. More important, bulk coding saves money since vendors typically charge a fixed amount for the creation of each database record. (Lassoing 500 travel receipts together under one database entry costs 1/500th of coding each document - and indeed the cost savings is likely greater because fewer fields likely will be used for documents coded in bulk.)

Bibliographic Coding. In contrast to bulk coding, bibliographic coding is done on a document-by-document basis. Basic bibliographic ("bib") coding captures the following objective information about a document: date, author, recipient, copyees, and the "re" or subject line.[9]

In designing the field structure, one needs to focus on both coding and retrieval. For example, should M. Mayerson, Marc Mayerson, MSM, Marc Steven Mayerson, MM, and Marc S. Mayerson be coded identically? Should the end-user be forced to perform a search for "Mayerson or MSM or MM" in order to retrieve all my documents? Should the coder be forced to code documents where MM appears as always and inevitably "Mayerson" documents, without regard to their content or the possibility that some other MM existed within the organization? Names standardization can be imposed on the front end of the coding process through the use of hard-wired substitution rules (in the event that the relevant witness's names and their aliases are known by the time of coding) or in a back-end process by which all entries in any field where a variant of name appears are reviewed and then standardized.

Similarly, when a document does not have a date or a re line, for example, should the coder try to supply one? For dates, there may be information within the document indicating it is a quarterly report from the third quarter of 1996; using an attribution rule (created during the database-design phase, such as using the last date of the quarter as the coded date of an undated quarterly report) the database can record that imprecise-though-relatively-accurate date (and thus permit the population to be put in chronological order).[10] Likewise, documents that do not have a subject line or a meaningful re line can have one created; typically, the coder will review the contents of the first paragraph or two and seek to create or enhance the re line. By creating or enhancing a re line, one obtains a more intelligible hit list following a search, which therefore avoids the necessity of a lawyer or paralegal having to view a document that would be, upon such inspection, irrelevant to the search being conducted.[11] This increases the efficiency of the document review and, for personnel billing time on an hourly bases, saves the client money.

Finally, other objectively discernable, non-bibliographic information relating to a document's "characteristics" may be captured, such as whether the document is handwritten, whether there are marginalia (handwritten comments in the margins), whether the document is in a foreign language, or whether the document is a carbon copy. Because discovery-document populations inevitably contain many duplicates, capturing marginalia may permit the user, on seeing the hit list so indicating, to review the one of the twelve identical copies containing marginalia.

Enhanced Bibliographic Coding: Names and Orgs/Keywords. In addition to capturing information at the ordinary bibliographic and document-characteristic level, coding can capture other "objective" information within a document.

The bibliographic-coding process naturally captures authors and recipients, but it typically would not record that a document exchanged between two people contains important information about a third person mentioned in the text. Ordinary bib coding will not highlight this document for retrieval, because the third-person's name appears in text. A memorandum to one person that describes what another person said at a meeting may well be useful in the deposition of the speaker. Consequently, it is not uncommon for a few document types to capture the names of individuals and organizations/companies appearing within the text.

In much the same vein, key words that appear in text can be captured in a key-word field. Unlike "names and orgs," which the coders should recognize they should capture upon inspection, key words require the coders to remember the words they are supposed to identify. As a practical matter, coders can effectively keep only a limited number of keywords in their head, so keyword lists need to be boiled down to a manageable number; a rule of thumb suggests something in the range of thirty. Moreover, this keyword list needs to include synonyms for the core words of interest, which serves to further pare down the number of key words that practically can be captured.

I. Optical Character Recognition and ReKeying

The methodology for constructing a database is premised on accessing documents through the use of a coded field structure. However, words in the source document cannot be searched (other than to the extent keywords in text have been captured). Accordingly, an important possible addition to the litigation-support system is to have the ability to conduct word searches within the documents across all or part of the population.

There are two ways of enabling such text searches: a manual process and a computerized one. The manual process involves someone retyping the document in a text-file format (like any word-processing program) and adding that data to the litigation-support system. The re-keying of text typically is done overseas with two people each typing the identical document, and the two versions electronically compared to identify errors. The computerized version of this process, known as optical character recognition (OCR), uses the computer to recognize what letters correspond to the "letter shapes" on the image of the original page.

OCR technology is increasing in speed and accuracy, but OCR'd discovery documents typically are replete with errors in recognition. As a practical matter, for documents more than 20 years old or that are in poor physical condition, OCR accuracy rates are fairly low. In a 250 word page, even a 95 percent accuracy rate yields 12 to 13 misspelled words a page. Such errors may be "cleaned up" in a process similar to the one used for rekeying - running the same document through multiple OCR engines and comparing the differences or through a manual process of word-by-word verification. The clean-up process adds considerable expense, but "dirty" OCR, without clean up, is not likely to be sufficiently accurate to rely on as the sole means of retrieving the discovery documents.

Where documents have been text converted either manually or through OCR, the basis for retrieval is a boolean, word-proximity search. In addition to the problem of inaccuracy in recognition or rekeying, boolean searches do not account for synonyms or other word choice; generally, unless the word(s) searched for are in the document in that form, the desired document will not be retrieved. Though some retrieval engines employ fuzzy logic and other programmatic techniques to increase the likelihood that the desired document will be swept into the hit list, those same techniques naturally increase the number of irrelevant documents also retrieved.[12]

One advantage of OCR'ing a population is that one can respond if the shape of the case changes (dramatically) after coding has been completed. Particularly where a population has been issue coded and the case has changed, such as antitrust case turning into a patent case, having some access to the population through text searching can be invaluable.

I. Next Generation Tools

As litigation-support approaches and technology become more widespread, we can expect continued development in the methodologies, processes, and tools applied to the specialized context of litigation.

The current model of coding documents represents evolutionary development from the big-case methods developed in the horse-and-buggy days. For a new paradigm, some thought has been given to using OCR and full-text searching to facilitate coding rather than as a substitute or complement to it. This is an emerging approach dependent on the increasing power and sophistication of OCR programs and the computers that run them as well as the improving physical quality of discovery documents increasingly at issue in litigation at present which were generated after the widespread adoption of laser printers. In this model, after the document population has been scanned and OCR'd as the first part of document processing, the OCR'd population is combed using (fragments of) a lengthy and comprehensive list of keywords to select documents for coding (applying the appropriate level of treatment for each document type). The advantage of this approach is that the entire population is OCR'd (which costs only in computer time and initial software acquisition costs, *i.e.*, OCR is capital expense, not labor), and only a fraction of the documents are coded, which is labor intensive and thus more costly. The OCR sweep sifts the more relevant documents by use of the keywords, which then are sent for bibliographic or other coding. In other words, rather than coding all letters at a certain level of treatment, the only letters that are coded are those containing keyword fragments. Coding documents in some degree based on a rough relevance test should result in a database with a higher percentage of useful or relevant documents than the current methods permit and at a lower overall cost.

We now also are seeing the development of virtual coding operations, with coders distributed nationwide, internationally, or within a law firm - all able to access the source images to be coded via the internet. Such distributed coding promises access to a sufficient number of coders with specialized knowledge that could increase the accuracy and reliability of the database (such as coders with knowledge of molecular chemistry to code documents in a genetic- engineering patent dispute), faster turnaround because of the unlimited number of coders available, and potentially lower unit costs for coding.

These process changes, however, do not affect the end user or how she accesses information that has been captured in the litigation-support system. External data-manipulation programs are being introduced, which permit higher-order and thus higher-value management of the data in the discovery database by treating the data points each as manipulable, sortable objects; each field is treated as a handle to the database record that can be grabbed and grouped by the user through the external program in any number of ways. Current databases require the formulation of a query in order to retrieve data; these external programs permit one to navigate the database fields as objects, hopping from one category to another without querying. The process of following a chain of inquiry - what documents did someone write, who received those documents, and what other documents did the recipient also receive - is accomplished by clicking and dragging and dropping, not by querying the main database with field delimiters.[13] These external programs permit the easy export and memorialization of information in the main database, such as the accumulation of database records in virtual notebooks by witness, issue or other common elements. Finally, these products also may have links to document-management software so that key memoranda can be registered in the program and thus made available as their own manipulable objects.

Finally, one can expect in the near term the (further) development of tools that actually will analyze the data collected in the litigation-support system. Databases remain mute sources until accessed and reviewed by lawyers and paralegals. Although the current state of the art permits extremely accurate and extremely fast location and retrieval of documents or data out of the database, these tools do not tell the user what's in the database (or the discovery collection). The next important tool will be data-analysis software - that is, software that reads and analyzes the information in the litigation-support system and organizes the information for the user in some directed manner. In other contexts, 3-D data mapping software is being introduced, which organizes the data into logical clusters displayed for the user via a topographical-map metaphor. Such data-mapping software tells the user the contents of the materials and their significant relationships before the user makes any query. These types of tools could prove extremely valuable in reviewing a population for substance - that is, analyzing the population to discover facts, relationships and themes.[14] As computer processing power continues to increase, the speed and intelligence of these programs should increase to the point that they should quickly become useful supplements to the current state-of-the-art systems utilizing imaged-based litigation-support databases, combined with external data-manipulation-and-organizational programs.

CONCLUSION

As with other areas where computer-based technology has been applied, litigation-related applications of technology have moved from their origins where computers were used to automate existing processes followed by the current Rococo period where we embroider improvements on the computerized systems to finally - and what is emerging now - where we rethink how we go about the entire processes to better take advantage of what power computers have to offer. Though it may be argued that computer technology penetrated the legal profession slower than in other industries, it is also the case that the technology providers did not (and may still not) understand the unique information-management and analytical needs of lawyers, and litigators in particular. That is changing as more and more lawyers adopt the current tools and as lawyers increasingly take a leadership position in spurring the development of computerized tools for our use in serving our clients' ends.

---

[1] Copyright 2002 Marc S. Mayerson. Mr. Mayerson is a partner at Washington, D.C.'s Spriggs & Hollingsworth, where in addition to his litigation practice representing corporate policyholders in major insurance-coverage disputes, he is the partner responsible for the firm's use and application of computer-based technology. He regularly speaks at legal-technology symposia on the application of computerization to legal practice and in particular litigation support.

2] Computer video-display technology did not easily display those images until the creation of the VGA standard display resolution in the early 1990s.

[3] The reduction in response time in viewing the images accounts for the most significant source of cost savings associated with imaging the documents and putting them on-line. In the absence of imaging, even if a database is used, the process of retrieving documents is labor intensive and therefore expensive. A lawyer, for example, will prepare a list of documents to be retrieved and provide it to a paralegal or file clerk, who in turns will remove the documents from the master set, copy the documents for the lawyer, replace the "original" documents in the master set, and provide

the copy to the lawyer. This process can be repeated as lawyers ask for copies of documents by witness or by issue or to be put in chronological order or the like. Without imaging, documents are photocopied again and again.

[4] The person that takes the original document, removes the staples and bindings, and otherwise prepares the physical document for scanning or photocopying.

[5] Document-control systems utilized in litigation typically handle all documents of a single type uniformly because the purpose of the litigation support system is to enable the lawyers and paralegals to *find* the material that is important; one does not first find all the important stuff and then put it into the litigation support system. This means that all letters, for example, are coded identically, resulting in both relevant and irrelevant material receiving the same level of treatment. Ideally, one would prefer not to treat the irrelevant material at all (and thus spend no money or time on those documents), but it is not cost effective - or even wise - to engage in a relevance-triage at the outset of the case to separate which documents of a given type are sufficiently important to warrant coding. There are several reasons for this. First, the only people that can perform such a relevance cut are lawyers and experienced paralegals; the dollar cost of deploying these resources at the outset usually cannot be justified. Second, the shape of a case can change over time, such that documents of marginal relevance to the case as filed (when the relevance cut would have been made) become central to the case that is about to be tried. Third, a relevance determination should really only be made once in a case (as an efficiency value and as a cost-savings measure); if that is done after the database is created, the information can be memorialized and communicated.

[6] These levels of treatment are "objective." Content-based coding, known as "subjective coding," by which certain documents are identified as "hot" or as related to a predefined list of issues involved in the case, usually is reserved for reviewers at the lawfirm (lawyers and paralegals) rather than an outside vendor.

[7] At least two companies have developed proprietary software that permits the capturing of certain objective information at scan time through the use of preassigned labels to particular keys on a keyboard (such as the function or "F" keys) or key-entry device. This software permits the scanning team to assign keys to a particular document sources, sites, document types, date ranges, etc. This technology generally is no substitute for full-blown coding; however, the technology captures information in a fielded database that later can be exported to a full-fledged database (created by the coders); alternately, and as an intermediate step, this technology yields a crude database that permits some pathway into the documents while the main database is gesstating, a process that can take several months.

[8] Of course, there are cases where travel receipts are key documents (such as conspiracy cases), in which event that document type would not be a candidate for bulk treatment.

[9] The type of document the particular piece of paper is has already been determined and captured, because it is the document typology that sorts the population for bibliographic coding.

[10] In the absence of creating a date, when the database records are placed in chronological order, documents without dates will appear first or last in the hit list.

[11] When a date is created or re line created or enhanced, one can memorialize that by placing the created information in brackets, thus indicating to the document reviewer that the date or subject was concocted by the coder.

[12] One trick that ameleoriates some of these problems is to search by word fragment - letter strings - rather than an entire word. Using a letter string that is part of the word first decreases the number of letters of that word that the OCR program must have correctly recognized, and because key words

are often used more than once in a text using the fragment increases the likelihood that the desired, accurately OCR'd fragment will be found and the correct document retrieved.

[13] For example, in one program, the user can begin a database navigation by locating all documents authored by a person. Immediately, that data (and the records they represent) can be sorted in a new hierarchy based on the recipients. The results of this re-ordering would then indicate what documents were received by the recipient and the like.

[14] These programs should prove helpful in handling depositions, as to which the current state of the art is a combined issued-coded database and full-text search capacity (with additional tools like sticky-note comments, highlighting, and the like).