



Approach to E-Discovery Boolean Search

Methodically improving search for better results

**Megan Bell
Winston Krone, Esq.**

May 2011

KIVU CONSULTING, Inc.
44 Montgomery Street
Suite 700
San Francisco, CA 94104
Tel: (415) 524-7320
Fax: (415) 524-7325
www.kivuconsulting.com
California PI License #26798

No portion of this document may be reproduced, reused or otherwise distributed in any form without prior written consent of Kivu Consulting, Inc.



Introduction

Boolean searching is widely used in e-discovery projects for identifying and retrieving electronically stored information (ESI). Unlike more automated search techniques where rules and algorithms work behind the scenes to deliver search results, Boolean search is the “manual” development and construction of the search query that is used to generate search results. There are several benefits to “manual” Boolean search including defensibility of search results that make Boolean search a mainstay in e-discovery. Boolean search yields a highly targeted set of results when part of a well-defined search project.

This guide is divided into 5 sections to assist users in achieving the best results from the use of Boolean search.

1. Understand e-discovery search.
2. Use a consistent and reliable process.
3. Understand implications of search tool set-up and use.
4. Optimize Boolean search statement construction.
5. Review of Boolean search examples.

1. Understand e-discovery search.

Boolean Search: An Accepted and Established Option

Keyword and Boolean logic search techniques have long been used in the legal community to search for electronically stored information (ESI)¹ and accepted as such in case law. *The Sedona Conference Best Practices Commentary on the Use of Search and Information Retrieval Methods in E-Discovery* states, “The legal community is familiar with keyword and natural language searches on Westlaw® and Lexis® in the context of legal research, and to a lesser extent the use of ‘Boolean’ logic to combine keywords and ‘operators’ (such as “AND,’ ‘OR’ and ‘AND NOT’ or ‘BUT NOT’) that produce broader or narrower searches.” The same techniques have been applied in the field of e-discovery to locate ESI.

¹ ESI is any information stored in digital format. This includes data stored in computers, cell phones, SD cards, network servers, etc.

Boolean search is widely accepted, and research on the use of Boolean search has supported its continued use. In “Assessing Alternative Search Methodologies,” (New York Law Journal, April 22, 2008) H. Christopher Boehning and Daniel J. Toal, concluded that, “It may be that alternative search methods eventually will surpass the performance of keyword and Boolean searches, but that day does not yet seem to have arrived. The independent research conducted to date suggests that, for the time being at least, nothing beats Boolean, particularly when used as part of an iterative process.” Boolean search is an effective technique when coupled with the right process and knowledge to structure Boolean statements.

When not properly executed, Boolean search can be ineffective as cited in the few studies available on performance of search techniques for legal information. In a well-known 1985 study by Blair & Marin, attorneys believed that 75% of relevant documents were identified although the reality was more like 20%. To further stress this point, many e-discovery professionals cite a 2007 TREK study where other search techniques identified 78% of relevant documents whereas Boolean search identified less than 22%. These studies underscore the need for a reliable and thorough process to conduct Boolean search.

Legal Factors Impacting Search

The legal framework for “discoverable” ESI² has evolved from The Sedona Conference and early case law precedent to codification in the Federal Rules of Civil Procedure. This maturation of the e-discovery process has resulted in the following objectives:

1. Locate ALL relevant ESI with respect to proportionality.³
2. Use a defensible process to capture, analyze and produce relevant ESI—i.e., it must hold water in a legal proceeding.
3. Use a thorough methodology to keyword development that will withstand judicial scrutiny.
4. Employ computer-based search techniques like Boolean search where possible.

The challenge of meeting these objectives in the search process is two-fold—having basic knowledge about ESI and understanding the search process.

² Discoverable ESI. Federal Rules of Civil Procedure Rule 34 discusses discovery of documents and specifies “writings, drawings, graphs, charts, photographs, and other data compilations...”

³ Federal Rules of Civil Procedure Rule 26 specifies “All discovery is subject to the limitations imposed by Rule 26(b)(2)(C).”

ESI in a Nutshell

There are three basic principles to understanding ESI and how its relationship to its source impact the search process.

1. ESI Can Come From Anywhere. ESI exists in variety storage media. Media sources may include computers, flash memory sticks, cell phones, cameras, GPS devices, Google's servers and even X-Box machines. It can come from multiple individuals, organizations and geographic regions.

2. The complexity of stored ESI expands at the device-level. The originating device is the starting point of technical complexity. Within a single device, ESI is affected by formatting for data storage and retrieval, the operating system, applications, encryption and user-specific configurations.

3. The confluence of these technical factors determines for ESI what will be available for search on a single device. These differences impact the final interpretation of any relevant ESI.⁴ For a given source of ESI, this includes type of data (e.g., SMS text), available metadata (e.g., Date Modified), and other source characteristics (e.g., file access or size limitations). The combination of different devices used for different reasons by different individuals or organizations can result in exponential complexity in large cases.

2. Use a consistent and reliable process.

Search Process: The EDRM Framework

Searching for relevant ESI is not a fly-by-the-seat-of-your-pants process where a single round of identifying keywords from educated guesswork yields an acceptable set of relevant ESI. With the standard for discovery being "all ESI" with respect to proportionality, using an appropriate process and thoughtful planning are essential to maximizing the identification of relevant ESI.

As previously cited from H. Christopher Boehning and Daniel J. Toal, "nothing beats Boolean, particularly when used as part of an iterative process." One such process is defined in an industry-standard publication entitled *EDRM Search Guide*. (See Figure 1.) "The Electronic Discovery Reference Model (EDRM) was created to develop resources and best practices for...e-discovery consumers and providers [to] reduce the cost, time and manual work associated with e-discovery." This process is a foundation and iterative process for completing any ESI-related search.

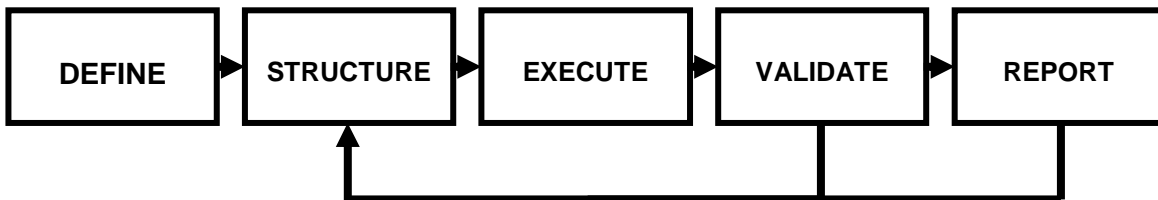


Figure 1. Search Process Phases, *EDRM Search Guide*, version 1.14:

⁴ Relevant ESI refers to ESI that has meaning in a specific legal matter and possible evidentiary value.

Source: Downloadable from http://edrm.net/2008_2009/search.php

1. Define.

The search process begins with establishing a search objective (e.g., identification of privileged ESI) and an expected outcome (e.g., retrieval of responsive emails sent by John Smith). This provides the platform for developing a project plan.

The project plan is an outline of targeted media (including network servers or backup data), search criteria including keywords and custodians, constraints such as who will do the search, definition of the final deliverables, search process methodology, tools used and reporting of results. The project plan may also include specific questions or issues to be addressed during the search process.

ESI and the tools employed to search can have complex functionality with many interdependencies ranging from the source of data capture (e.g., cell phone or laptop) to the systems that host search tools. To account for the complexities, evaluating ESI for a specific legal matter is an important step to complete at the beginning of the search process. (See section, “Achieving Better Results.”)

2. Structure.

The success of any search hinges on appropriately structuring Boolean statements.⁵ A Boolean search statement can be as simple as find “Confidential OR Secret” or a list of 20 search terms and phrases that incorporate more complex Boolean search elements (e.g., *Confidential w/ 5 (proprietary or “trade secret”)). Keys to building successful Boolean statements include:

- Understanding the basic foundations of Boolean search
- Adjusting search terms and phrases to account for the settings of the search tool
- Modifying search terms such as email address or a phrase containing special characters such as the “%” symbol.

Testing Boolean search statements during their development is a critical success step. Testing identifies issues and maximizes the quantity and quality of search results.⁶ Issues may stem from the use of Boolean logic (i.e., ordering of search terms or parentheses placement). Specific search terms may also result in issues such as a large number of false positives—search results that are responsive to search terms but not relevant. Testing assists in determining whether to rearrange or re-write a Boolean search statement. (See section, “Achieving Better Results.”)

Additionally, search tools have software bugs that can directly affect search results. For example, the dtSearch Text Retrieval Engine⁷ is widely used search software that also provides the engine in many e-discovery products. However, the software suffered from a defect that was not widely known but carried a big consequence in conducting searches. Specifically, dtSearch was not locating search terms in Office 2007 documents. This problem was present for more than two years after Office 2007’s release on November 30, 2006.⁸ A best practice to counter missed

⁵ Boolean search syntax in this paper is based on dtSearch syntax.

⁶ Relevant ESI is ESI that matches one or more search criteria in a Boolean statement and is pertinent to a scope of the search project.

⁷ dtSearch is a platform of text search and retrieval tools such as dtSearch Text Retrieval Engine. This suite of tools is developed by dtSearch Corp. More information can be found at www.dtsearch.com.

⁸ Source: <http://www.microsoft.com/presspass/press/2006/nov06/11-062007officertmpr.msp>. Kivu Consulting identified and verified this issue in early 2010 and reported it to dtSearch Corp. Kivu has been involved in subsequent proofing of the beta version designed to fix the problem. For more information, review “Software Bugs in Common E-Discovery Search Tools” published by Kivu Consulting, In. at http://www.kivuconsulting.com/Forms___Publications.html.

search results due to search tool software bugs is to maintain a list of known issues for the search tool being used. This list can be used to evaluate source ESI for potential issues.

3. Execute.

When executed, a well-defined Boolean search statement yields an optimized result set. Earlier preparation and testing should eliminate issues and questions regarding the result set.

4. Validate.

The *EDRM Search Guide*⁹ indicates that “the validation steps seeks to determine if the search ‘worked’ — that is, did the search include all of the records that were to be searched and did it achieve the goals established during the definition phase?” Much of the validation work should take place during the structuring of the Boolean search statement. Validation is a final confirmation against what was observed when developing and testing a Boolean search statement. (See Step 2 above.)

Validation may include the use of metrics such as file counts or gigabytes of data returned from a search. Metrics vary according to the needs of a legal matter, but identifying relevant metrics for a specific matter improve the quality of results. (See section, “The Importance of Search Quality.”)

PST Processing Summary		
Custodian	# PSTs Processed	# De-duped MSGs Responsive to Search Terms
Custodian 1	25	4,531
Custodian 2	19	3,211
Custodian 3	23	996

Figure 2. Sample email processing report illustrating the output of searched and de-duped email MSGs that were originally stored as weekly backups (PST files) on a server.

NOTE: Validation of a final result set should also apply to post-search processes such copying or transferring results to a review platform. This includes checking logs or other production records against results from the original search output. Transferring data across different systems can alter results (i.e., delete files from a result set). For example, the report illustrated in Figure 2 can be used to validate the results of copying files to a secondary storage location.

5. Report.

Reporting is the compilation of project documentation from different project phases. Sufficient detail should be present that the search can be redone if set up with the same conditions and source data. Attachments that describe source data, output logs, and custodians should also be included.

Achieving Better Results

Having a process such as the EDRM search process provides the foundation for consistency. However, researching, testing and analyzing search results determine whether a search is successful. This includes understanding source data and the criteria necessary to building the most appropriate Boolean search statements. Finally, analysis of search results is completed using metrics that are most appropriate for a given matter.

1. Research source ESI.

⁹ EDRM Search Guide, version 1.14. Downloadable from http://edrm.net/2008_2009/search.php.

The objective of research, in particular during project definition, is to establish knowledge about source data and the requirements for Boolean search statements. This knowledge also may be used to configure search tools. Research must be comprehensive and not be limited to the search terms. Characteristics such as file type and other metadata may be equally as important in the final result set. Listed below are several properties to consider when reviewing source ESI.

Property	Description
File Type	<p>This is the kind of file—picture, database, Word file, etc. This can often be determined by looking at a file’s extension—three digit code following a file name.</p> <p>Files such as system files, media files (e.g., jpg), other binary file types, and certain database files are not good candidates for text search unless specific metadata (e.g., a file’s name) is being searched.</p> <p>Some file types may not be readily identifiable. The file extension may be incorrect or not present. For these cases, certain forensic tools can “match” the unique file “header” to the file extension and identify a false match—i.e., a file extension does not match the actual file type. However, not all file types are easily identifiable, and research is critical to determine whether text is available for search. Additionally, specialized tools like a hex editor may be needed to examine a file and determine file type.</p>
Data Type	<p>ESI can exist as “structured” or “unstructured” data. “Structured” data refers to databases—organized and related collections of information. Search engines like dtSearch may treat each row in a database table as a separate document—making review a potential challenge. “Unstructured” ESI refers to email, Microsoft Word, and other text documents.</p> <p>Media files such as pictures are stored as binary files. Search of these files is limited to available metadata.</p>
File System Metadata	<p>This refers to properties that describe ESI including creation and access dates, source application, author and other file properties. Metadata varies greatly across different file types, file systems, operating systems, and storage location.</p>
Original Storage Location	<p>This is the original source location of the ESI. Where and how data was originally stored defines the ESI available for capture, metadata, applications in use and history of stored ESI.</p>
Compressed / Archived	<p>Files stored in a compound file (or compressed file) format can present many complexities. For example, text files stored in heavily nested .zip files may not be searchable without “mounting” or extracting the contents of the .zip files. A file’s native metadata may also be compromised when stored in compressed format.</p>
File Size	<p>Large file sizes may present search issues for older search tool versions or outdated computer hardware that hosts search tools. Heavily compressed files can also present issues since the actual file size may be substantially greater.</p>

Property	Description
Read/Write	Files that are read-only and can't be modified.
Sharing	Files that are shared by more than one person. For example, a Word document with change tracking.
Other	There are thousands of file types in existence with many different file attributes. Examples include versioning in Microsoft Office files or compound file structure of Microsoft Office documents that can impact search.

2. Test Boolean statements.

Boolean search statements often presented in literature (e.g., “cat OR dog”) can disguise the complexity of identifying relevant ESI using Boolean search statements. Boolean search statements employed to retrieve relevant ESI often use techniques like grouping, stemming and proximity in addition to traditional Boolean operators like AND, OR and NOT. Testing the behavior of more complex Boolean techniques consists of the following factors:

- Search Environment. The result set from a Boolean statement depends on the set-up of the search tool employed for searching. For example, periods are treated as space symbols in many search engines. Entering “U.S.A.” in Google to search returns results with “USA” like “USA.gov.” Boolean searches are constrained by the rules of a given search tool.
- Terms and Operators. Each term in the Boolean statement must have the proper formatting and logic structure. Mistakes in either element may directly affect the search result set. For example, the statement, “vet AND dog OR cat” is not the same as “(vet AND dog) OR cat” as each returns a different set of results.
- Order of Search Terms. Correct ordering of terms, operators and characters is as important as using the right terms and logical operators. For example, looking for a vet that handles dogs or cats indicates that vet is the primary term and either dog or cat is acceptable when searching for vet. The correct Boolean statement would be “vet AND (dog OR cat).”

3. Evaluate search quality.

The success of any Boolean search statement resides in the quantity and quality of relevant results that are returned after searching. Information engineers and researchers use measures such as precision and recall to compare the success and limitations of different search techniques.

- Recall. The quantity of retrieved relevant ESI to all relevant ESI.
- Precision. The quality of retrieved relevant ESI to all relevant ESI.

In practice, these measures may not be exactly quantified as defined since “all relevant ESI” may not be practically quantifiable. However, it is still possible to evaluate search results using other quantitative and qualitative metrics. During the definition phase of the search process, a variety of metrics can be identified for use in validating Boolean search results. Example metrics include:

- Number of hits per keyword
- Number of total returned emails
- Number of total returned user files
- Number of returned emails where a specific custodian is the sender
- Files having the correct full file path
- Custodians with no search results
- Quantity of emails and user files per custodian (e.g., gigabytes)
- Review of metadata

- Sampling of documents across email, other user docs, custodians and keywords for qualitative review

Use of metrics to evaluate the outcome of Boolean search statements maximizes the recall and precision of search results. Metrics should be used throughout research and development of Boolean search statements not as a single step after the final execution of a Boolean search statement. Ongoing evaluation of results while developing a Boolean statement leads to optimizing the structure of the Boolean query. It also provides the baseline for validating final results of relevant ESI that will be handed over to a client.

3. Understand implications of search tool set-up and use.

Index, Then Search

Most search engines use indexing to improve search efficiency. A search engine index is similar to the index in the back of a book in that there is a master list of words, and each word has known locations. When search terms are entered into the search engine, matched items are identified by their indexed location.

Part 1: Preparing the Index

Indexing is completed using a defined set of rules to identify and locate words. By default, search tools contain a pre-established set of rules to address the most likely search scenarios. Familiarity with the type of rules and their set-up is necessary to understanding the resulting output from Boolean search (e.g., words such as “between” may not be searchable).

1. Tokenization. Search tools must identify individual words—or “tokens”—in order to index. In English and many other languages, the space character is used for word-recognition. This technique does not apply to all languages (e.g., Chinese), and search tools employ other rules and methods to identify words.

```

- - - -
t t T t
u u U u
v v V v
w w W w
x x X x
y y Y y
z z Z z

[Hyphens]
-

[Spaces]
!"#$%&'()*+,-./:;<=>?@[\\5c]^`{|}~

```

Figure 3. Default alphabet file settings.¹⁰
 Source: default.abc file from Wave Software’s Trident Pro version 6.6

¹⁰ This snapshot displays a default list of characters that are treated as spaces. Trident Pro version 6.6 uses the dtSearch® Text Retrieval / Full Text Search Engine, and this alphabet settings file is part of dtSearch.

2. Alphabet Settings. Search engines require a method of character recognition in order to recognize words or other meaningful character combinations like a phone number. This also includes special characters like apostrophe and case definition (e.g., uppercase A). Characters are either recognized and used or ignored depending on their definition in Alphabet settings. For example, the apostrophe symbol is often treated as a space and ignored in indexing. (See Figure 3.)

3. Noise (Stop) Word Settings. To increase search efficiency (i.e., reduce the number of irrelevant hits), search engines ignore common words like “the” or “all.” These words are classified as noise (or stop) words and are stored in a separate file that is referenced during indexing. Boolean logical operators like AND, OR and NOT are also in this file and considered noise words (i.e., unable to search for the word “and”). Ignoring adjustment of default noise words on a case-by-case basis could result in missing relevant search results.

4. File Type. Search tools can recognize and process hundreds of file types. This includes common user file types such as Microsoft Office documents and specialized file types such as database files. Depending on the source ESI and project scope, it may be necessary to add or delete file types to improve the quality of indexed ESI. One example is the treatment of media files. Media files contain limited text (e.g., file name) and may be removed from indexing when they are not relevant to an investigation.

5. Hyphens. By default, hyphens are treated as spaces during indexing. When this occurs, words like “e-mail” are transformed to “e mail.” Hyphen behavior can be adjusted in some search tools and may be worth modification depending on how hyphens are used in source ESI. The recommended treatment is to leave hyphens as spaces, but hyphens can be recognized in the indexing process by adjusting alphabet settings.

6. Foreign Language. With the appropriate libraries installed, most search engines can handle a broad range of written languages. This includes Cyrillic languages and character-based language like Chinese. As a standard practice, a comprehensive set of language libraries should be installed in any search tool used to evaluate ESI. Additionally, researching source ESI for foreign language content early in a project will assist in determining whether translation of search terms is needed.

Part 2: Conducting Boolean Search

In Boolean search, two or more words or phrases are combined with Boolean logical connectors like “AND” and “OR” to form a Boolean search statement. Search engines use this statement to locate results that meet the criteria identified in the statement. Proper construction of a Boolean statement has several benefits such as:

- Increasing efficiency in searching large sources of ESI.
- Identifying ESI based on keyword relationships.
- Locating a collection of related ESI for a review.

Iteratively constructing and testing Boolean search statements is the most effective process to acquiring the optimal set of search results. One caution with Boolean search is that different search tools may not produce the same results. Therefore, reviewing a specific tool’s help guide and evaluating a search tool prior to any search project are important to obtaining reliable results.

Presented below is advice on Boolean search and several examples of Boolean search. The examples are structured from less to more complex.

4. Optimize Boolean search statement construction.

Incorrectly or inefficiently drafted Boolean search statements increase the risk of missing important search results. This risk can be minimized with the adoption of several principles that improve the quality of Boolean search statement construction.

1. Invest time in identifying relevant search terms and phrases. Successful searches begin with understanding the search terms that are present in the source ESI. This includes researching indexes created from source ESI and selecting a sample of files for detailed review.

2. Determine which search terms to search in combination. The grouping of search terms in a single search statement is as important as search term selection. Search terms that are improperly combined in a single Boolean search statement return ambiguous search results. For example, a search statement such as “(pediatrician OR doctor) AND (baby OR child) AND (ears OR nose)” can return results that range from otolaryngologists and to piercing a baby’s ears. A more concise search statement such as-“(pediatrician AND (ears OR nose))” would deliver more targeted results.

3. Use the most appropriate Boolean logic. Boolean search logic enables the identification of files with specific combination of related words or phrases. The best results are achieved when the relationship between a set of search terms or phrases is understood. For example, searching for all email sent by John Smith containing the terms “Project X” and “opportunity” is a clear search objective. However, the selection of Boolean operator choices of OR or W/ to use on the construction of the Boolean search statement will yield different result sets. Understanding whether terms “Project X” and “opportunity” must be within a few words of each other or whether either word must be present is critical to acquiring the relevant set of search results.

4. Adjust Boolean search statements to account for variations in search term wording, spellings and abbreviations. Search terms may have multiple wording variations, spellings or abbreviations. This includes names such as “Jennifer” and commonly misspelled words such as “accommodate.” When acceptable, variations should be incorporated into Boolean search statements. However, if the variations are too general or identify incorrect search results, variations may be excluded. The following steps are important to identifying the spectrum of variations for a search term or phrase and whether to incorporate it into a Boolean search statement:

- Research source ESI for specific variations before constructing Boolean statements.
- Evaluate the impact of adding variations to Boolean search statements.
- Adjust Boolean search logic to account for variations. This may include the use of stemming to find multiple variations of the same root (i.e., the root “appl” is part of “apply” and “application.”). Another possibility is the creation of several individual Boolean search statements.
- Always return to the original source when proof-checking search terms. For example, when a term is an email address, review actual emails to confirm the correct spelling—rather than relying on legal filings or litigation metadata.

5. Modify Boolean search statement when special characters are present. Search terms or phrases may contain characters such as the “%” symbol. Since these special characters are often ignored by search engines, a Boolean search statement may require modification to location results. For example, the phrase “% of Rejects” may be searched using variations such as “ of Rejects” or “of w/ 2 rejects.” It may be necessary to modify the search tool settings if a suitable wording variation can not be identified.

6. Avoid complex nesting of Boolean search statements. Complex Boolean search statements with layers of AND, OR and special operators such as W/ can produce an optimal set of results or an incomplete or incorrect set of results. When search terms are ambiguous, the risk of a poor result set increases substantially. For instance, the search statement “(product X OR product Y) w/ 5 (red OR green OR purple)” is better structured with product X and product Y being separate Boolean search statements. “(product X OR product Y)” in the same statement means that results could contain product X alone, product Y alone or both product X and product Y. The statement “product X w/ 5 (red OR green OR purple)” will generate results only about product X and the identified colors. Using fewer search terms or phrases in combination with the most appropriate Boolean logic yields more relevant search results.

5. Review of Boolean search examples.

Boolean Search Foundations

AND:

Using AND between search terms returns results where all search terms must be present. Returned ESI must contain the keywords defined with the AND relationship. Using + also identifies a word as required and can have the meaning as an AND relationship.

<u>Example</u>	<u>Boolean Statement</u>
ACME sales in California	ACME AND sales AND california

OR:

Using OR between search terms returns results where any search term is present.

<u>Example</u>	<u>Boolean Statement</u>
California or Nevada	California OR Nevada

Parentheses:

Parentheses enforce the order of relationships in Boolean search. Consistent use of parentheses minimizes the risk that a search tool will incorrectly interpret a Boolean search statement.

<u>Example</u>	<u>Boolean Statement</u>
ACME in California or Nevada <u>NOTE:</u> Without the use of parentheses, a search tool will determine whether the AND or OR is more important in the search criteria and how to interpret the AND combination. The result is a larger quantity of search results, in particular search results without relevance.	ACME AND (California OR Nevada)
ACME sales in California and Nevada <u>NOTE:</u> Parentheses do not impact this search but the relationship is more clearly defined with parentheses. Consistent use of parentheses can lead to better consistency in overall quality of search results.	ACME AND sales AND (California AND Nevada)

Quotes:

The default behavior of many search engines is to search for individual words and not phrases—or groups of words in a specific order. Searching for a phrase requires the use of quotes. In cases where there are multiple variations of a quoted phrase, the specific variations of the quote should be individually set up for search to ensure results for all variations are found. (By design, many search engines assume an AND relationship for phrases when no quotes are present. For example, “legal hold” without quotes returns files that contain both words “legal” and “hold” in the same document but not necessarily together. Quotes limit a search specifically to the phrase “legal hold.”)

<u>Example</u>	<u>Boolean Statement</u>
2009 sales competition	“2009 sales competition”
2009 sales competition in Florida	“2009 sales competition” AND Florida
A-1 homes and the abbreviation “est. 1992” <u>NOTE:</u> “A-1” and “est.” may not return results due to the hyphen and period characters. When search terms contain special characters, evaluate results for the search term and determine whether to modify indexing rules.	“A-1 homes” AND “est. 1992”

NOT:

Incorporating NOT into a Boolean statement excludes a keyword or phrase from search. NOT can be used at the beginning of a statement, but only applies to the first keyword. Using “-” serves the same purpose.

<u>Example</u>	<u>Boolean Statement</u>
ACME sales in California but not in Sacramento	ACME AND sales AND California NOT Sacramento
Return all results for “apple” except granny smith apples.	apple AND NOT “granny smith”
Search for California and exclude Sacramento.	California and NOT Sacramento
If the search logic for “California and NOT Sacramento” is reversed, NOT only applies to Sacramento, but search results may vary. This depends on the design of the search tool in use. <u>NOTE:</u> Not all search tools allow NOT to be used first unless there are other Boolean operators in use.	NOT Sacramento and California

Wildcards (= and ? and *):

Wildcards expand the scope of a search for a single number (=), single character (?), or group of characters (*). When used, the result is an increase in the quantity of returned search results. Wildcards are a powerful tool and can be used to resolve search issues when certain search terms can't be found or contain typos. Using wildcards increases the amount of time required to search (with the greatest impact when used at the beginning of a keyword).

<u>Example</u>	<u>Boolean Statement</u>
Search for all years in from 1970 to 1979.	197=
Some names have more than one spelling. Search for the name Karen where there are the same number of characters but different spellings (i.e., 'k' or 'c' at the beginning and the fourth character could be 'h', 'e' or 'y').	?ar?n
In large email systems, there may be several variations for the same user name. Search for user names that begin with jsmith.	jsmith*
In large email systems, there may be several variations for the same user name. Search for user names that end with smith.	*smith
In large email systems, there may be several variations for the same user name. Search for user names that contain smith.	*smith*

Proximity (W/):

Proximity refers to searching for combinations of search terms that appear together within a specific number of words. W/ can be used at the beginning or end of a Boolean statement and can also be combined with NOT.

W/-based searches can quickly become complex or ambiguous if not properly defined. This is addressed by using parentheses to enforce the correct relationship. Caution should be taken to test placement and phrasing of W/ logic in a Boolean search statement to ensure that search results yield relevant search results.

<u>Example</u>	<u>Boolean Statement</u>
"Household" and "cleaners" must be within 10 words of one another.	household W/ 10 cleaners
"Cleaner" must be within 10 words of specific household locations such as "bathroom" or "kitchen."	household W/ 10 (bathroom OR kitchen)
"Cleaner" must be within 10 words of specific household locations such as "bathroom" or "kitchen" but exclude "auto."	household W/ 10 ((bathroom OR kitchen) AND NOT auto)

Proximity (NOT W/):

NOT restricts search results when combined with W/. The same issues for W/ apply to NOT W/.

<u>Example</u>	<u>Boolean Statement</u>
The term "household cleaners" must not be within 10 words of "packaging."	"household cleaners" NOT W/ 5 packaging

Stemming (~):

Stemming is the identification of word variations that use the same root. In the right circumstance, stemming is more efficient than listing all variations of a word. However, unexpected results may also appear as indicated in the example below. Finally, stemming rules vary by language so the search tool in use should be appropriately configured to account for language differences.

<u>Example</u>	<u>Boolean Statement</u>
All variation of the root word "list."	list~ <i>(Returned results will include list, listing, and listed. However, listless could also be returned.)</i>

Fuzzy (%):

In cases where a character in a search term is missing or misspelled, fuzzy searching provides a method of searching for these variations. The position of % character determines where fuzzy searching takes place within a search term. Fuzzy searching allows the user to control the granularity of the search. The number of % characters used determines how many letters can be different within a keyword. Fuzzy searching should be used with caution as it reduces the accuracy of search by locating more search results.

<u>Example</u>	<u>Boolean Statement</u>
Variations of "accommodate"	ac%om%odate
Variations of "their" and "there"	the%r%
Variations of "they're" and "there"	the%%re

Phonic (#):

Many search tools support phonic searching. Phonic searching expands the scope of search, but should be used with caution as it relies on a search tools' rules and not solely on the search terms and logic in a Boolean search statement.

<u>Example</u>	<u>Boolean Statement</u>
Variations of the last name "Smith" where a "y" or "i" may be present as well as a silent "e" at the end.	#smith

Synonym (&):

Synonym searching, like phonic searching, expands the scope of search. It can be useful in a preliminary review of ESI to identify a range of keywords used in a data set. Caution should be used as the search is limited to the rules and dictionary built into the search tool used for search.

Example	Boolean Statement
Synonyms for “attorney” which may include “counsel” or “advocate.”	Attorney&

Numeric Ranges (~~):

Numeric range searching is a limited, but useful technique (e.g., researching a numeric range of product ID’s). It only works with integers greater than zero and does not support searching numeric values with commas and periods (e.g., currency). Commas and periods are treated as spaces by most search tools.

Example	Boolean Statement
Find all product ID’s within 5 of 160 to 170.	“product id” W/5 160~~170.

Fieldname Searching:

Boolean search can be applied to specific search fields like the Sender field in an email search. The fields available to search depend on the source ESI. Boolean search is highly effective when searching specific data fields, but the data field’s definition (e.g., text or date) defines the scope of Boolean search.

Special Cases

File Path Search:

File names or file paths may contain special characters that have specific meanings as Boolean search symbols. For example, tilde (~) is used to name temporary documents in the Microsoft OS environment (e.g., ~WRL0005.tmp) and is used in search tools for stemming. To avoid confusion, use wildcards, fuzzy searching and quotes when possible.

Also consider using a field search instead of developing a Boolean statement with a file name or path. If file name or file path are searchable fields, then using the “contains” operator with a Boolean statement will increase the likelihood of acquiring results.

Example	Boolean Statement
Search for file “~WRL0005.tmp.” Notice that wildcards are substituted for the tilde and period to avoid search confusion.	*WRL0005*tmp
Search for path “C:\Windows\Temp\~DF0AF9DF5F05A4F380.TMP.”	C*Windows*Temp *DF0AF9DF5F05A4F380*TMP

Email Address:

Boolean search statements for email addresses must account for special characters like “@” and the variations of email address content (i.e., email address only versus an address and an associated contact name). Researching source ESI is important to determining variations in email addresses. In some cases, users may replace characters such as “@” with words like “at” when the emails exists on websites or in other public domains. Email addresses can be searched within specific fields (e.g., Sender) or though entire text documents.

NOTE: Many search engines do not index special characters like “@” and apply a space where these symbols exist in their original text. Even though these characters are not recognized, using quotes around an original email address returns the same result as long as the search engine uses the same rules of character recognition as the index. Keep in mind that it may be possible to adjust indexing and searching for recognizing special symbols. (See “3. Understand implications of search tool set-up and use.” in this paper for additional discussion.)

Email Address (continued):

<u>Example</u>	<u>Boolean Statement</u>
Search for email address john_smith@acme.com . NOTE: Email addresses should always be in quotes.	"john_smith@acme.com"
Search for email address john_smith@acme.com where the name John Smith is also identified as the contact name for this email address. NOTE: Contact names for an email address can be defined in different formats (i.e., first name and then last or vice versa). Determine the variations and then list in the Boolean search statement.	"john_smith@acme.com" AND ("John Smith" OR "Smith, John")
Search for John Smith when he is identified by his name (John Smith), his nickname "JJ," and his computer alias "jsmith13." NOTE: Consider alternative spelling conventions (e.g., periods used in nicknames) when building the Boolean statement.	((John OR "JJ" OR "J-J") W/2 Smith) AND jsmith13

Apostrophes, Percents, & Other Characters:

```
[Hyphens]
-

[Spaces]
!"#$%&'()*+,-./:;<=>?@[\\5c]^`{|}~
```

Figure 4. Default alphabet file settings.
Source: default.abc file from Wave Software's Trident Pro version 6.6

Search tool indexing rules affect the results for search terms that use special characters like apostrophes. The solution is to use quotes to correct for the impact of search tool indexing rules. (See Figure 4.) For example, words like "can't" may be separated into two words by a search tool—"can t"—if special characters like the apostrophe are treated as spaces due to their definition in the Alphabet file. Noise words can also present issues. Noise words like "the" may be ignored even though there is importance when defined in a phrase. Consider the case when searching for a musical band called "The The." If "the" is a designated noise word, there would be no results returned from this search.

Using Wildcards to Correct for Search Tool Bugs:

Search tools may contain software bugs that affect search results. The best practice for handling software bugs is to understand which bugs are present in the search tool being used. With this knowledge, evaluate search results while constructing Boolean search statements. In some cases, Boolean wildcard characters can be used to adjust for software bug issues.

<u>Example</u>	<u>Boolean Statement</u>
Search for the term “confidential” when the search tool does not correctly identify text in the footer of a Microsoft Office document. <u>NOTE:</u> Using a wildcard does not guarantee a search term will be identified. It increases the chances of locating a search term that is not immediately recognized by a search term as text.	*confidential*

About Kivu Consulting and Wave University

Kivu Consulting combines technical, legal and business experience to offer investigative, e-discovery, incident response and forensic analysis services to clients worldwide.

Megan Bell mbell@kivuconsulting.com

Winston Krone, Esq. wkrone@kivuconsulting.com

Offices:

Headquarters

44 Montgomery Street, Suite 700
San Francisco, CA 94104

Tel: (415) 524-7320

Fax: (415) 524-7325

New York

845 Third Avenue, 6th Floor
New York City, NY 10022

Tel: (646) 430-8383

Fax: (646) 430-8384

Los Angeles

515 South Flower Street, 36th Floor
Los Angeles, CA 90071

Tel: (213) 596-7700

Fax: (213) 596-7705



Wave University was designed to provide free educational opportunities through webinars, 'lunch and learn' sessions and Trident software training webinars. Webinars offered will focus on discovery issues and be hosted by industry thought-leaders and experts. Registration is free so please visit www.discoverthewave.com for more information.

Alexander J. Lewis
alex.lewis@discoverthewave.com
300 S Orange Ave. Suite 900
Orlando, FL 32801
Tel: (407)-432-9422
Fax:(407)-386-7240

