



ARCASEARCH
archive it. access it.

Using Boolean Search:

Under Search Options/Type you will find an option for "Boolean". This search option is the most powerful and also the most complicated. For that reason we have dedicated the following section to its use.

A Boolean search request consists of a group of words, phrases, or macros linked by connectors such as **AND** and **OR** that indicate the relationship between them.

Examples:

Search Request	Explanation
apple and pear	<i>both words must be present.</i>
apple or pear	<i>either word can be present.</i>
apple w/5 pear	<i>apple must occur within 5 words of pear.</i>
apple pre/5 pear	<i>apple must occur 5 or fewer words before pear.</i>
apple not w/5 pear	<i>apple must not occur within 5 words of pear.</i>
apple and not pear	<i>only apple must be present.</i>
name contains smith	<i>the field name must contain smith</i>
apple w/5 xfirstword	<i>apple must occur in the first five words.</i>
apple w/5 xlastword	<i>apple must occur in the last five words.</i>

If you use more than one connector, you should use parentheses to indicate precisely for what you want to search. For example, **apple and pear or orange juice** could mean (apple and pear) or orange, or it could mean apple and (pear or orange).

IMPORTANT: Noise words, such as “if,” “and,” “the” are ignored in searches because they are so prevalent in the language. Including these as actual search words would overwhelm the search function..

Words and Phrases

In a Boolean search, you do not need to use any special punctuation or commands to search for a phrase. Simply enter the phrase the way it ordinarily appears. You can use a phrase anywhere in a search request. Examples follow:

apple w/5 fruit salad

If a phrase contains a noise word, dtSearch will skip over the noise word when searching for it. For example, a search for **statue of liberty** would retrieve any document containing the word statue, any intervening word, and the word liberty.

Punctuation inside of a search word is treated as a space. Thus, **can't** would be treated as a phrase consisting of two words: **can** and **t**. The string **1843(c)(8)(ii)** would become 1843 c 8 ii (four words).

“AND” connector

Use the **AND** connector in a search request to connect two expressions, both of which must be found in any document retrieved. For example: **apple pie and poached pear** would retrieve any document that contained both phrases.

(apple or banana) and (pear w/5 grape) would retrieve any document that:

- (1) contained either apple OR banana, AND
- (2) contained pear within 5 words of grape.

“AndAny” connector

AndAny lets you combine a search for required search terms with other terms that are optional. The words before **AndAny** are required, and the words after **AndAny** are optional. Example:

(apple and pear) AndAny (grape or banana) would find any word that contains apple and pear, and grape and banana will also be counted as hits.

“OR” connector

Use the **OR** connector in a search request to connect two expressions, at least one of which must be found in any document retrieved. For example, **apple pie or poached pear** would retrieve any document that contained apple pie, poached pear, or both.

“W/N” connector

Use the **W/N** connector in a search request to specify that one word or phrase must occur within N words of the other. For example, **apple w/5 pear** would retrieve any document that contained apple within 5 words of pear. The following are examples of search requests using W/N:

**(apple or pear) w/5 banana (apple w/5 banana)
w/10 pear (apple and banana) w/10 pear**

Some types of complex expressions using the **W/N** connector will produce ambiguous results and should not be used. The following are examples of ambiguous search requests:

**(apple and banana) w/10 (pear and grape)
(apple w/10 banana) w/10 (pear and grape)**

In general, at least one of the two expressions connected by **W/N** must be a single word or phrase or a group of words and phrases connected by **OR**. Example: **(apple and banana) w/10 (pear or grape) (apple and banana) w/10 orange tree**

ArcaSearch uses two built-in search words to mark the beginning and end of a file: **xfirstword** and **xlastword**.

The terms are useful if you want to limit a search to the beginning or end of a file. For example, **apple w/10 xlastword** would search for apple within 10 words of the end of a document.

The **pre/N** connector is like **W/N**, but it also requires that the first expression must occur before the second. Example:

(apple or pear) pre/5 banana

“NOT” and “NOT W/N”

Use **NOT** in front of any search expression to reverse its meaning. This allows you to exclude documents from a search. Example:

apple sauce and not pear

“NOT” standing alone can be the start of a search request. For example, **not pear** would retrieve all documents that did not contain pear.

If **NOT** is not the first connector in a request, you need to use either **AND** or **OR** with **NOT**:

apple or not pear not (apple w/5 pear)

In this instance, the **NOT W/** ("not within") operator allows you to search for a word or phrase not in association with another word or phrase. Example:

apple not w/20 pear

Note that unlike the **W/** operator, **NOT W/** is not symmetrical. That is, **apple not w/20 pear** is not the same as **pear not w/20 apple**.

In the **apple not w/20 pear** request, ArcaSearch searches for apple and excludes cases where apple is too close to pear. In the **pear not w/20 apple** request, ArcaSearch searches for pear and excludes cases where pear is too close to apple.