<u>Guide to Advanced Search in NEEScentral</u>

1. Overview

The NEEScentral Advanced Search offers users numerous options for making database searches more precise and getting more useful results. Based on Lucene technology, an open source Java product adopted by the Apache Software Foundation, the NEEScentral Advanced Search engine allows users to search the NEES database as well as supporting documents.

In order to effectively utilize the NEEScentral Advanced Search option, users should become familiar with the idioms and grammar used in a Lucene-based search engine. This document highlights those functionalities and offers users tips to help refine their search results.

2. Terms

A search query can be separated into terms and operators. There are two types of terms: Single Terms and Phrases.

- "Single Term" is one word, e.g. "test" or "research".
- "Phrase" is a group of words surrounded by double quotes such as "shake table".

Multiple terms can be combined together with Boolean operators to form a more complex query (see Boolean Operators).

3. Fields

Lucene-based search engines also support fielded data. When conducting a search, users may either specify a field, or use the default field. The field names and default field is implementation specific.
Users can search any field by typing the field name followed by a colon ":" and then the term. For example, document files typically have a filename as the field name and text (content of the document), which is the default field. In order to find the document, "Data Curation Web Report", containing the text "EntityType", users should enter:

- filename:"Data Curation Web Report" AND text:entitytype
  or simply,
- title:"Data Curation Web Report" AND entitytype

It is significant to note, since text is the default field, a field indicator is not required. The field is only valid for the term that it directly proceeds, for example:

- filename:Data Curation Web Report

The query only finds "data" in the title field. It will find "Curation" and "web" and "report" in the default field (in this case the text field).

4. Wildcard Searches

Single and multiple character wildcard searches within single terms (not within phrase queries) are also supported in the NEEScentral Advanced Search. To perform a single character wildcard search, use the "?" symbol. To perform a multiple character wildcard search, use the "*"symbol.

- Single character wildcard searches look for similar terms that match the original term with a single character replaced.
- For example, to search for "text" or "test" use the search: "*te?t*"

- Multiple character wildcard searches look for 0 or more characters.
  For example, to search for "test", "tests" or "tester", use the search: "*test*"*

Users may also use the wildcard searches in the middle of a term, for example: "*te*t*". However, users cannot use the "*"or "?" symbols as the first character of a search.

5. Fuzzy Searches

Fuzzy searches are based on the Levenshtein Distance, or Edit Distance algorithm. To conduct a fuzzy search, use the tilde, "~", symbol at the end of a single word term. For example, to search for a term similar in spelling to "roam" use: "*roam~*". This search will find terms like "foam" and "roams".

6. Proximity Searches

The ability to locate words that are separated within a specific distance is also supported by the Lucene-based search engine. To conduct a proximity search, use the tilde, "~", symbol at the end of a phrase. For example, to search for an "EntityType" and "Name" within 10 words of each other in a document, use the search: "*"entitytype name"~10*".

7. Boolean Operators

Boolean operators allow terms to be combined through logic operators. Lucene supports AND, "+", OR, NOT and "-"as Boolean operators. Note: Boolean operators must be ALL CAPS.

OR (Operator)

The OR operator is the *default* conjunction operator. This means that *if there is no Boolean operator between two terms, the OR operator is used*. The OR operator links two terms and finds a matching document if either of the terms exist in a document. This is equivalent to a union using sets. The symbol || can be used in place of the word OR.

To search for documents that contain either "EntityType Large-Scale" or just "Large-Scale", use the query:

- "EntityType Large-Scale" Large-Scale
  or
- "EntityType Large-Scale" OR Large-Scale

AND (Operator)

The AND operator matches documents where both terms exist anywhere in the text of a single document. This is equivalent to an intersection using sets. The symbol "&" can be used in place of the word AND.
To search for documents that contain "two-span reinforced" and "concrete bridge", use the query:
- "two-span reinforced" AND "concrete bridge"

+ (Operator)

The "+"or required operator requires that the term after the "+"symbol exist somewhere in a the field of a single document.

To search for documents that must contain "concrete" and may contain "bridge", use the query:

- +concrete bridge

NOT (Operator)

The NOT operator *excludes* documents that contain the term after *NOT*. This is equivalent to a difference using sets. The symbol "!" can be used in place of the word *NOT*.

To search for documents that contains "bridge" but not "concrete", use the query:

- bridge NOT concrete

Note: The NOT operator cannot be used with just one term. For example, the following search will return no results:

- NOT "concrete bridge"

Grouping

Parentheses may also be used to group clauses to form subqueries. This can be very useful if users desire to control the Boolean logic for a query. To search for either "bridge" or "reinforced" or "concrete", use the query:

- (bridge OR reinforced) AND concrete