TLG – Thesaurus Linguae Graecae

TLG is a research center at the University of California, Irvine. Founded in 1972 the TLG has already collected and digitized all ancient text from Homer to AD 600 and a large number of texts from the period between AD 600 and the fall of Byzantium in AD 1453. The goal is to create a digital library that will include the entire corpus of Greek literature from Homer (8th century BC) to the present era.

TLG is available through the Electronic Resources on BearCat.

--to begin searching, choose A (Unicode input and Display) and switch to Greek fonts.

[Greek fonts: Change by using the language bar or menu on the desk top. If bar does not appear, add fonts by clicking : Settings / Control Panel / Regional and Language Options / Language / Details / Add / select Greek from list / click ok. Language menu should appear on desktop or press ALT/SHIFT to switch to Greek.]

Simple Search:
Author: select entire list of works or individual work(s)
--to see list of works, click on name, then “Show works”
(Author list is in Latin; list includes New Testament)

Work: enter title (will need to use English font) OR select work by classification, character or genre
--If you search “Work,” the work(s) will appear. Select all or part, then the text search screen appears for further instructions.

Text screen – offers advanced search options.

Publication: search for a particular edition

“TLG Texts”
--This screen automatically appears after selecting author

Wildcard Searching

The Online TLG® includes a powerful Wildcard facility, allowing for indeterminate searches of various kinds --- on both the word index and in full text searches. These searches are specified using regular expressions, a notation used to describe different string patterns. Each element is introduced below with examples.

<table>
<thead>
<tr>
<th>Search for ANA as part of a word:</th>
<th>ANA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Search for ANA as a prefix (word index)/at the start of a line (full text):</td>
<td>^ANA (Details)</td>
</tr>
<tr>
<td>Search for ANA as a suffix (word index)/at the end of a line (full text):</td>
<td>ANA$ (Details)</td>
</tr>
<tr>
<td>Search for the word ANA (word index only):</td>
<td>^ANA$ (Details)</td>
</tr>
</tbody>
</table>

Baylor University Libraries, 02/28/2007
Complex or Proximity searches:

In addition to the primary search string, the user may specify up to two further search strings modifying the current search. These strings may either trigger alternate searches in the same text, or constrain the current search by context. Any instances of the first additional search string pertinent to the current search are highlighted in the search results in maroon. Any instances of the second additional search string are also highlighted in maroon. For examples, see the following.

Note: The primary search string is searched by either an index search or a full text search, as specified on the search form. The secondary search strings are searched by full text search on the text in the vicinity of the primary search string instances.

If the primary search string is searched through the word index, then the running total given at the top of each results page is the total number of primary string instances looked at. For example, it may report that it has gone through "319 out 5063 instances", which means 319 instances of the primary string have been considered, but only the following 5 out of those have a match for both the primary and the secondary string.

Qualify: and (proximity):

This option restricts search results to those containing both the primary search string and the secondary search string (within a specified interval of words or lines). For example, the following are results from a search for OROS, and EU within 2 lines of OROS, in the Iliad

Qualify: except (proximity):

If this kind of proximity is specified, the search returns results where the secondary search string is not contained within the specified interval of words or lines from the primary search. For example, the following are results from a search for OROS in the Iliad, for which EU does not
appear within 2 lines of OROS:

**Qualify: or (proximity):**

If the primary search has already been qualified by one secondary search string, this option allows an alternative qualification. For example, the following are results of a search in the *Iliad* for OROS, with *either* TEU within 2 lines of OROS, *or* ANHR within 1 line of OROS:

**Proximity within n words/lines:**

The region within which the secondary string is searched is defined in terms of either lines or words. Lines are required to have textual content: lines containing only beta escapes (e.g. @6 'blank line') or citations are not included in the count. The minimum value of the number of lines is 0, meaning the secondary search string must be located in the same line as the primary search string. If a search string is split across two lines, it is deemed to lie within the line where it ends; thus, in the following, METE/XEIN and KATAQE/NTA are *not* considered to lie in the same line, since the words end on two different lines:

I)E/NAI: E) DE/ TIS TO\ PARAUTI/KA ME\ N MH\ E)QE/ LEI CUMPLEI=N, METE/XEIN DE\ BOU/LETAI TH=S A)POIKI/ AS, PENTH/KONTA DRAXMA\S KATAQE/NTA *KORINQI/ AS ME/NEIN. H)=SAN DE\ KAI\ OI( PLE/ONTES POLLOI"

Words are defined by the alternation of word delimiters (space, dash) and textual characters. Beta escapes and citations are ignored, and text interrupting a word is considered part of that word. Thus, in the following, OI)/-{A)NT.KWN is counted as a single word (the marginalium is ignored), and PO/LIN is considered to occur one word before TA\N, ignoring the intervening @1 ('end of page'):

**Proximity near/before/after first word:**

The region within which the secondary string is searched may be either centered at the end of the primary string (near), begin at the end of the primary string (after), or finish at the end of the primary string (before). If the search is before or after, the specified number of words or lines is counted from the end of the primary word. If the search is near, the number of words or lines is counted either side of the end of the primary word.

**Qualify: or (same text):**

As distinct from the foregoing proximity searches, this option allows the user to search for two distinct search strings at once. The following, for example, results from a search for either POIHS or MOUS in Lucian's *Phalaris*