

SHEBANQ/EMDROS MQL-tutorial v1.26

(Oliver Glanz, Old Testament department, Andrews University)

Contents

1	Basic introduction: Tool preparation	2
1.1	Introduction	2
1.2	Database architecture	2
1.3	Learning the MQL query language	4
1.4	Looking at the text in SHEBANQ: Visualizing syntactic structure	5
1.5	Relation between SHEBANQ and EMDROS Query Tool	6
2	Building Simple Queries	7
2.1	Searching on the WORD-level	7
2.2	Searching on the PHRASE-level	11
2.3	Searching on the Clause/Sentence-level	13
2.4	Miscellaneous searches	14
3	Utilizing Operators for Complex queries	15
3.1	UnorderedGroup	15
3.2	NOTEXIST	15
3.3	AS	16
4	Publishing your query in an academic paper	17
5	Advanced Queries	18
6	ETCBC in Accordance	22
7	ETCBC in Logos	22

1 Basic introduction: Tool preparation

1.1 Introduction

1.2 Database architecture

a) Selection of popular data categories (see Gen 20:1-4 CTT):

Level	object types	Feature	Value
text section			
	verse	book	Genesis, etc.
		chapter	1, 2, etc.
		verse	1, 2, etc.
syntax			
	sentence	sentence	
	clause	typ	WayX, Way0, etc.
		rela	Attr, Objc, etc.
		domain	N, Q, etc.
	phrase	typ	VP, NP, etc.
		function	Pred, Subj, Objc, etc.
lexeme/morphology			
	word	lex	">BRHM/"
		lex_utf8	"אברהם/"
		gn	m, f
		nu	sg, pl, du
		ps	p1, p2, p3
		st	a, c
		vs	qal, nif, etc.
		vt	perf, impf, etc.

Query samples:

Finding Gen 1:1

```
=> [verse book IN (Genesis) AND chapter IN (1) AND verse IN (1)]
```

Finding attributive clauses

```
=> [clause rela IN (Attr)]
```

Finding the word "Abraham"

```
=> [word lex_utf8 = "אברהם/"]
```

Finding "Abraham" as subject of an attributive clause in Genesis chapter 25-35:

```
=>
[verse book IN (Genesis) AND chapter IN (25,26,27,29,30,31,32,33,34,35)
 [clause rela IN (Attr)
  [phrase function = Subj
   [word FOCUS lex = ">BRHM/"]
  ]
 ]
]
```

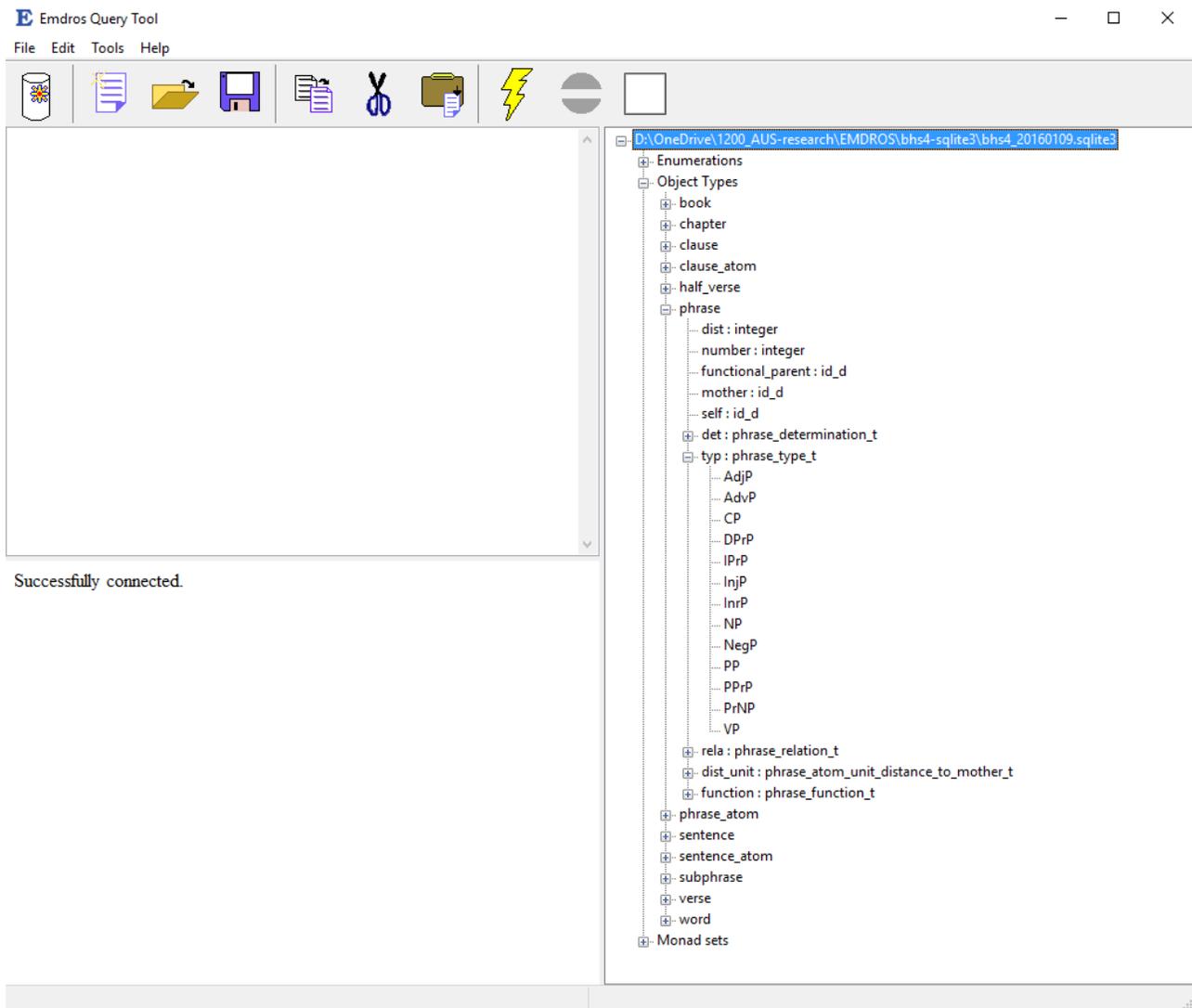
b) find all available object types, features and values here

i) SHEBANQ Documentation (Features by category):

(1) Full documentation: <https://etcbc.github.io/bhsa/>

(2) Quick reference: <https://github.com/ETCBC/shebang/wiki/MQL-Quickref>

ii) EMDROS Query Tool: right pane



- c) In case you want to use transcription instead of the Hebrew fonts you can find the transcription table here: <https://github.com/ETCBC/shebang/wiki/BHSA-Transcription>

1.3 Learning the MQL query language

- d) Read the Emdros Query Guide: <https://github.com/ETCBC/shebanq/wiki/Documents/MQL-Query-Guide.pdf>
- e) Now you can build the most advanced queries. See for example: <https://shebanq.ancient-data.org/hebrew/query?version=4b&id=491>

This query, for example, finds all cases in which the same direct speech introduction (same subject [e.g. Abraham] speaks to the same complement [e.g. to Abimelech]) is repeated after the initial direct speech has been heard. The query finds the following pattern:

X speaks to Y: "bla bla bla"

X speaks to Y: "bla bla bla"

In the default Hebrew text-grammar one would expect after the initial

X speaks to Y: "bla bla bla"

that

Y speaks/answers to X: "bla bla bla"

is following.

We are therefore searching for a rather uncommon pattern. The query was inspired by Genesis 20:9-10 where Abimelech initiated a direct speech twice without Abraham responding to Abimelech's first speech.

1.4 Looking at the text in SHEBANQ: Visualizing syntactic structure

← → ↻ Koninklijke Nederlandse Akademie van Wetenschappen [NL] https://shebanq.ancient-data.org/hebrew/text

SHEBANQ The Text Words Queries Notes Tools Help News Sources Log In

show words
show queries
show notes

etcbc 4 4b 4s Genesis ← 20 → hebrew phonetic text Notes (1/3) legend

18 verses

<p>אֲבְרָהָם ʔavrāh,ām</p> <p>אֲבְרָהָם >BRHM/ Abraham</p> <p>pers nmpr</p> <p>m sg NA a NA NA</p> <p>det Subj PrNP NA</p> <p>N WayX NA 1</p>	<p>שָׁם šš'ām</p> <p>שָׁם CM</p> <p>advb</p> <p>NA NA NA NA NA NA</p> <p>NA Cmpl PP NA</p> <p>N WayX NA 1</p>	<p>מִי mi</p> <p>מִן MN</p> <p>prep</p> <p>NA NA NA NA NA NA</p> <p>NA Cmpl PP NA</p> <p>N WayX NA 1</p>	<p>יָסַע yyiss,aʕ</p> <p>נָסַע NS<[</p> <p>pull out</p> <p>verb</p> <p>m sg p3 NA wayq qal</p> <p>NA Pred VP NA</p> <p>N WayX NA 1</p>	<p>וַ wa</p> <p>וַ W</p> <p>and</p> <p>conj</p> <p>NA NA NA NA NA NA</p> <p>NA Conj CP NA</p> <p>N WayX NA 1</p>
<p>יָשַׁב yy,ēšev</p> <p>יָשַׁב JCB[</p> <p>sit</p> <p>verb</p> <p>m sg p3 NA wayq qal</p> <p>NA Pred VP NA</p> <p>N Way0 NA 1</p>	<p>וַ wa</p> <p>וַ W</p> <p>and</p> <p>conj</p> <p>NA NA NA NA NA NA</p> <p>NA Conj CP NA</p> <p>N Way0 NA 1</p>	<p>נֶגְבַּי nn'egev</p> <p>נֶגֶב NGB/</p> <p>south</p> <p>subs</p> <p>m sg NA a NA NA</p> <p>rec 2</p> <p>NA Loca AdvP NA</p> <p>N WayX NA 1</p>	<p>הַ ha</p> <p>הַ H</p> <p>the</p> <p>art</p> <p>NA NA NA NA NA NA</p> <p>rec 2</p> <p>NA Loca AdvP NA</p> <p>N WayX NA 1</p>	<p>אֶרֶץ ʔ'aršā</p> <p>אֶרֶץ >RY/</p> <p>earth</p> <p>subs</p> <p>unknown sg NA c NA NA</p> <p>NA 1</p> <p>NA Loca AdvP NA</p> <p>N WayX NA 1</p>

1.5 Relation between SHEBANQ and EMDROS Query Tool

- f) “userfriendly” Offline (Emdros) vs all-powerful Online (SHEBANQ)
- g) Installing Emdros

2 Building Simple Queries

2.1 Searching on the WORD-level

- a) Simple search for the word “Abraham” in the book of Genesis in the chapter 17-22.

```
select all objects where
[chapter book = Genesis AND chapter IN (17,18,19,20,21,22)
[word FOCUS lex = ">BRHM/"] //or write: [word FOCUS lex_utf8 = "אברהם/"]
]
```

Link to this query: <https://shebanq.ancient-data.org/hebrew/query?version=4b&id=1365>

=> When you search for nouns you have to add “/” behind the word: “DBR/” (word)

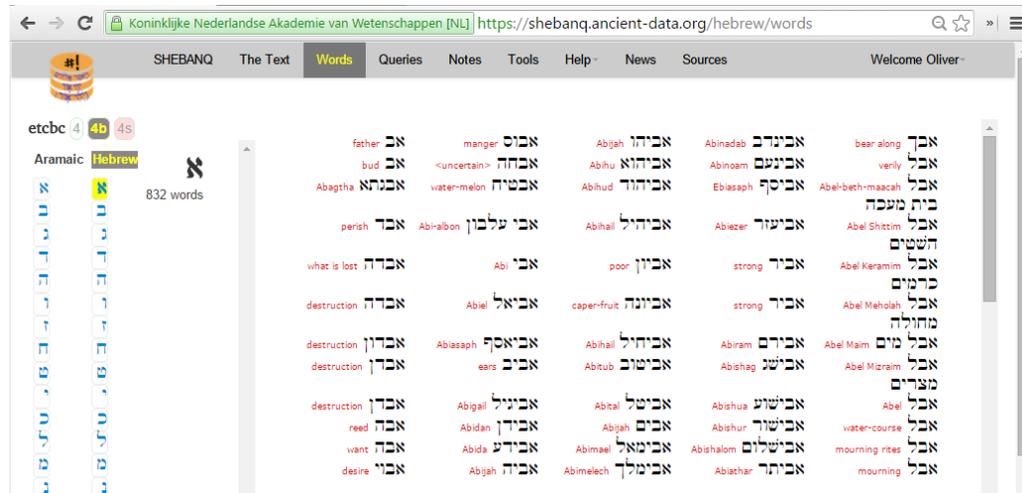
=> When you search for verbs you have to add “[” behind the word. “DBR[” (to speak)

If you want to save time with typing Hebrew words or transliterated words, you can just copy/paste them from a concrete text. For example:

Genesis 17: 5				
אֶתְּ	עוֹד	יִקְרָא	לֹא־	וְ
את	/עוד/	קרא	לא	ו
>T	<WD/	QR>[W	
<object marker>	duration	call	not	and
NA NA NA NA NA NA	m sg NA a NA NA	m sg p3 NA impf nif	NA NA NA NA NA NA	NA NA NA NA NA NA
n/a n/a	n/a n/a	J	n/a n/a	n/a n/a
Subj NA	Modif NA	Pred NA	Nega NA	Conj NA
NQ WxYX NA	NQ WxYX NA	NQ WxYX NA	NQ WxYX NA	NQ WxYX NA

For the lex_utf8 entry →
For the lex entry →

You can also go to the dictionary on SHEBANQ and copy past from there:



- b) Word advanced: Searching “Abraham” in clauses that contain a predicate in Way tense in the book of Genesis.

```
select all objects where
[book book = Genesis
[clause
[word FOCUS vt = wayq]
..
[word FOCUS lex_utf8 = "אברהם/"] //or write [word FOCUS lex = ">BRHM/"]
]
]
```

Link to this query: <https://shebanq.ancient-data.org/hebrew/query?version=4b&id=1366>

The full list of verbal tenses can be found in the ETCBC feature doc (go to https://etcbc.github.io/bhsa/features/hebrew/2017/0_home.html):

vt data

[Edit on GitHub](#)

verbal tense

Form of the verb that indicates tense or mood.

This feature is present on objects of type *word*.

The Hebrew verb has multiple stems, and each of these stems can be marked with tense/mood. For the possible stems, see *vs*. Not all words have verbal tense (e.g. nouns) and these are marked as *NA*.

Here is the list of possible values:

```
perf perfect
impf imperfect
wayq wayyiqtol
impv imperative
infa infinitive (absolute)
infc infinitive (construct)
ptca participle
ptcp participle (passive)
```

c) The query template for verbal morphology:

```
select all objects where
//section01: part of speech
[word FOCUS sp = verb
 //section02: preformative
  AND g_pfm ~ '\![J][.][A]!\!'
 //section03: lexeme
  AND lex ~ '^ [>BGDHWZXVJKLMNS<PYQRFCT] [>BGDHWZXVJKLMNS<PYQRFCT] [>BGDHWZXVJKLMNS<PYQRFCT] \[$ '
 //section04: verbal ending
  AND g_vbe = "[W."
 //section05a: verbal stem
  AND vs IN (qal,nif,piel,pual,hit,hif,hof)
 //section05b
  //AND vbs = "HT" //alternatively one could use vbs ("absent"=qal,piel,pual; "HT"=hit;
"N"=nif, "H"=hif,hof)
 //section05c
  //AND g_vbs ~ '\\[N][IE]\]' (this would for example search for all cases in which we
would find the stem signs for the niphāl ך and ך).
 //section06: pronominal suffixes
  AND g_prs ~ '[+].' //this finds any suffix form
]
```

Template explanation:

This query is intended to showcase how to build complex morphological queries. Here are some explanations that should help anybody to modify the query for him/herself:

//section01:

here we define the part of speech (sp) as verbs (verb)

//section02:

here the graphical preformative (that is phenomenological not paradigmatic description of the preformative)(g_pfm) is defined with the help of regular expression (~). Each pfm starts with "!" and ends with "!". Since I am using regular expressions I need to put a "\" before each "!" to make sure that the exclamation marks are not read as regular expressions but as ETCBC database values. The "[J]" stands for a Yod, the "[.]" stands for a Dagesh, and the "[A]" stands for a

Patach. A full transcription list for vowels and other Masoretic markers can be found here: <https://shebanq.ancient-data.org/shebanq/static/docs/ETCBC4-transcription.pdf>

//section03:

here we define the lexeme (lex) with the help of regular expressions. In the sample below we find 3 [>BGDHWZ XVJKLMNS <PYQR FCT] blocks. The letters within the square brackets of block 1 and 3 are all 23 Hebrew consonants in transcribed form. Block 2 misses the letters "W" and "J" in order to exclude Il"yod/waw verbs. Each block stands for one radical of the verb. The content within the square brackets allows for "OR" functionality. Thus, [BK] would stand for either "B" or "K". [>BGDHWZ XVJKLMNS <PYQR FCT] then stands for one consonant of the alphabet. Before the first block I have placed the regular expression "^" in order to indicate that this is the very beginning of the lexeme. At the end of the third block you find "[\" marking the lexeme as a verb. The normal sign of verbs is "[\". However, since we are using regular expression (regular expressions are enabled by the "~" signed after "lex" we have to make sure that "[\" is not read as a regular expression but as a normal sign. This we do by putting "\" before "[\". After the "[\" I have placed a \"\$" indicating that this is the very end of the lexeme. Using "^" and \"\$" guarantees that the verbal lexeme does not have more than 3 radical.

//section04:

here the graphical verbal ending (that is phenomenological not paradigmatic description of the preformative)(g_vbe) is defined. Each g_vbe has to start with "[\" and can then be followed by and consonant and vowel. In my case I have defined the "W" and "." (Dagesh) in order to define the ׀ ending.

//section05a:

here the verbal stem (vs) is defined.

//section05b:

alternatively to "vs" one could also chose for "vbs" (still paradigmatic representation) en "g_vbs" if one wants to look only for the graphical/phenomenological representation of the verbal stem and not for the paradigmatic information. E.g.: vbs = "HT" (thee following values are available: "absent"=qal,piel,pual; "HT"=hit; "N"=nif, "H"=hif,hof)

//section05c:

alternatively to "vs" or "vbs" (both paradigmatic representations) one could search for "g_vbs" if one wants to look only for the graphical/phenomenological representation of the verbal stem and not for the paradigmatic information. E.g.: g_vbs ~ '\\[N][IE]}' (this would for example search for all cases in which we would find the stem signs for the niph'al ך and ך).

//section06:

here we define the presence of pronominal suffixes. The features "pfm" (paradigmatic) and "g_pfm" (phenomenological/graphical) are available. If the paradigmatic "pfm" features is used the following values apply:

Here follows a list of the paradigmatic forms of the pronominal suffix (Hebrew version)

```
1sgC="NJ", "J"
2sgM="K"
2sgF="K="
3sgM="W", "HW"
3sgF="H"
1plC="NW"
2plM="KM"
2plF="KN"
3plM="HM", "M", "MW"
3plF="HN", "N"
```

A possible query could look like this:

```
[word FOCUS sp IN (verb,subs,prep) AND prs ~ '[JKWHMN]'] // this finds all suffixes that come
with at least one consonant (two consonants are possible)
OR
[word FOCUS sp IN (verb,subs,prep) AND prs = "J"] // this finds all suffixes that have only J as
consonant and no other consonant following.
OR
[word FOCUS sp IN (verb,subs,prep) AND prs ~ '[K][=]'] // this finds the homographic 2sg suffix
OR
[word FOCUS sp IN (verb,subs,prep) AND prs ~ '[JKWHMN][JWMN ]'] // this finds all suffixes that
come with two consonants
```

2.2 Searching on the PHRASE-level

a) Searching "Abraham" as a subject

```
select all objects where
[book book = Genesis
  [clause
    [phrase FOCUS function = Subj
      [word lex = ">BRHM/"]
    ]
  ]
]
```

Link to this query: <https://shebanq.ancient-data.org/hebrew/query?version=4b&id=1367>

The full list of phrase functions can be found in the ETCBC feature doc

(<https://etcbc.github.io/bhsa/features/hebrew/2017/function.html>):

function data

phrase function

The phrase function denotes the syntactic function of the phrase.

This feature is present on objects of type *phrase*.

Adju	Adjunct	
Cmp1	Complement	
Conj	Conjunction	
EPPr	Enclitic personal pronoun	!
ExsS	Existence with subject suffix	
Exst	Existence	
Frnt	Fronted element	
Intj	Interjection	
IntS	Interjection with subject suffix	
Loca	Locative	
Mod1	Modifier	
ModS	Modifier with subject suffix	
NCop	Negative copula	
NCoS	Negative copula with subject suffix	
Nega	Negation	
Objc	Object	
PrAd	Predicative adjunct	
PrCS	Predicate complement with subject suffix	
PrC	Predicate complement	
Pred	Predicate	
PreO	Predicate with object suffix	
PreS	Predicate with subject suffix	
PtcO	Participle with object suffix	
Ques	Question	
Re1a	Relative	
Subj	Subject	
Supp	Supplementary constituent	
Time	Time reference	
Unkn	Unknown	
Voct	Vocative	

Note

I was unable to locate a definition of this feature.

Examples needed, or longer descriptions.

The ! appears where the QUEST documentation has a cross mark. I do not know why these values have been marked.

b) Phrase advanced: Searching for the predicates that have Abraham as subject.

```
select all objects where
[book book = Genesis
  [clause
    [phrase FOCUS function = Subj
      [word FOCUS lex_utf8 = "אברהם/"]
    ]
    ..
    [phrase FOCUS function = Pred]
  ]
OR
[clause
```

```
[phrase FOCUS function = Pred]
..
[phrase FOCUS function = Subj
  [word FOCUS lex_utf8 = "אברהם/"]
]
]
```

Link to this query: <https://shebanq.ancient-data.org/hebrew/query?version=4b&id=1368>

2.3 Searching on the Clause/Sentence-level

a) Searching for a sentence that contains two clauses in Genesis 20

```
select all objects where
[chapter book = Genesis AND chapter IN (20)
  [sentence
    [clause FOCUS]
    [clause FOCUS]
  ]
]
```

Link to this query: <https://shebanq.ancient-data.org/hebrew/query?version=4b&id=1369>

b) Complex Sentence advanced:

i) Searching for a sentence that contains an independent clause and a dependent attributive/relative clause.

```
select all objects where
[chapter book = Genesis AND chapter IN (20)
  [sentence
    [clause FOCUS]
    [clause FOCUS rela = Attr]
  ]
]
```

Link to this query: <https://shebanq.ancient-data.org/hebrew/query?version=4b&id=1370>

The full list of rela values can be found in the ETCBC feature doc (go to

https://etcbc.github.io/bhsa/features/hebrew/2017/0_home.html):

Clause

Adju	Adjunctive clause
Attr	Attributive clause
CmpI	Complement clause
Coord	Coordinated clause
Objc	Object clause
PrAd	Predicative adjunct clause
PreC	Predicative complement clause
ReVo	Referral to the vocative
Resu	Resumptive clause
RgRc	Regens/rectum connection
Spec	Specification clause
Subj	Subject clause

2.4 Miscellaneous searches

- i) Searching for simple clauses with simple vocabulary (only possible on SHEBANQ): <https://shebanq.ancient-data.org/hebrew/query?version=4b&id=1106>

```
select all objects where
[clause_atom FOCUS
  [phrase FIRST AND LAST
    NOTEXIST [word rank_lex > 200]
    //NOTEXIST [phrase function IN (EPPr, ExsS, Exst, Frnt, IntS, Intj,
ModS, Modi, NCoS, NCop, Nega, Objc, PrAd, PrcS, Conj, Rela, Supp, Unkn)]
    [word]
  ]
]*{10-}
OR
[clause_atom FOCUS
  [phrase FIRST
    NOTEXIST [word rank_lex > 200]
    //NOTEXIST [phrase function IN (EPPr, ExsS, Exst, Frnt, IntS, Intj,
ModS, Modi, NCoS, NCop, Nega, Objc, PrAd, PrcS, Conj, Rela, Supp, Unkn)]
    [word]
  ]
  [phrase
    NOTEXIST [word rank_lex > 200]
    //NOTEXIST [phrase function IN (EPPr, ExsS, Exst, Frnt, IntS, Intj,
ModS, Modi, NCoS, NCop, Nega, Objc, PrAd, PrcS, Conj, Rela, Supp, Unkn)]
    [word]
  ]*
  [phrase LAST
    NOTEXIST [word rank_lex > 200]
    //NOTEXIST [phrase function IN (EPPr, ExsS, Exst, Frnt, IntS, Intj,
ModS, Modi, NCoS, NCop, Nega, Objc, PrAd, PrcS, Conj, Rela, Supp, Unkn)]
    [word]
  ]
]*{10-}
```

3 Utilizing Operators for Complex queries

3.1 UnorderedGroup

Every element that is contained directly under the UnorderedGroup can appear in any order. For example:

This query

```
select all objects where
[clause
  [UnorderedGroup
    [word FOCUS lex = "MCH="/]
    [word FOCUS lex = ">HRN/"]
  ]
]
```

Does exactly the same thing as this query:

```
select all objects where
[clause
  [word FOCUS lex = "MCH="/]
  ..
  [word FOCUS lex = ">HRN/"]
]
OR
[clause
  [word FOCUS lex = ">HRN/"]
  ..
  [word FOCUS lex = "MCH="/]
]
```

3.2 PLACE HOLDERS

In some cases, one wants to include place holders in the query. In order to use placeholders, the simple dot-sign “.” is used. However, the placeholder sign only functions when using in the context of regular expressions. In case one wants to search for , but only wants to use the consonants for the query since one is not familiar with the codes for the cantillation signs and vowel transliterations one could run the following search:

```
select all objects where
[word FOCUS g_word ~ "^N.*W.*V.*J.*H.*M.*"]
```

See: <https://shebang.ancient-data.org/hebrew/query?version=2017&id=2934>

In the above case we activate the regular expression by using “~” instead of “=”. With “^” we identify the “N” as the first sign of the word and with “.” we create potential spaces between the consonants.

In the following query I look for any lexeme that has the consonants “DBR” in its word. This would catch words like “MDBR/”, “LDBR/”, “DBR[”, “DBR/”, “DBR=”, etc.

```
select all objects where
[word FOCUS lex ~ ".DBR.*"]
```

See: <https://shebang.ancient-data.org/hebrew/query?version=4b&id=2934>

3.3 NOTEXIST

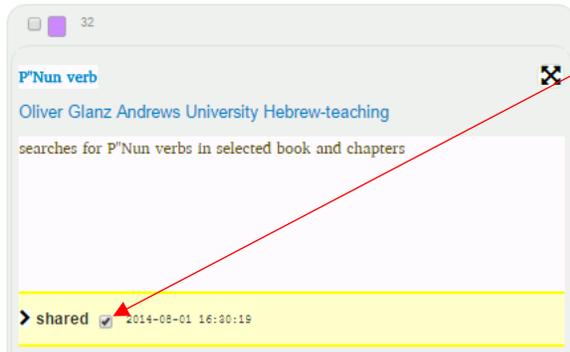
3.4 AS

See: <https://shebang.ancient-data.org/hebrew/query?version=4b&id=1465>

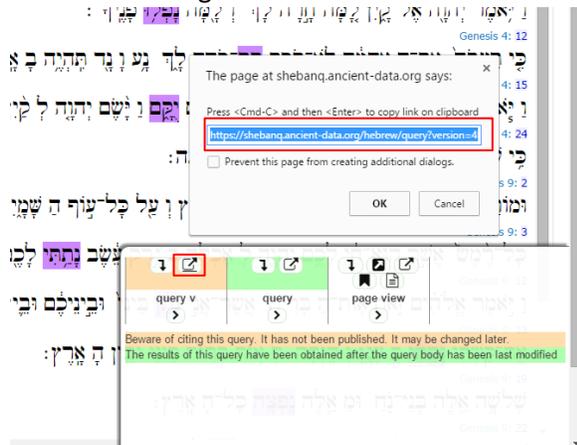
```
select all objects where
[clause kind=NC
  [word AS same_png FOCUS sp=prps]
]
[clause typ IN (WQt0, WQtX)
  [word FOCUS sp=verb
    AND ps=same_png.ps
    AND nu=same_png.nu
    AND gn=same_png.gn
  ]
]
```

4 Publishing your query in an academic paper

- a) Once you have run your query you can publish your query and use a PID for reference purposes in your paper/article/book/etc.
- b) Click the share box under your query description:



- c) In the bottom right corner click on "cite" and chose the right box over "query v":



- d) Now you can copy your PID and use it wherever you wish. I share my query no 491 with you here:
<https://shebanq.ancient-data.org/hebrew/query?version=4b&id=491>

2) Going wild:

- a) Using operators: OR, AND, NOTEXIST, First AND Last, etc.
 - i) Search P'Nun: <https://shebanq.ancient-data.org/hebrew/query?version=4b&id=32>
 - ii) Search same lexeme in different clauses: <https://shebanq.ancient-data.org/hebrew/query?version=4b&id=36>
- b) Check MQL query guide: <https://github.com/ETCBC/shebanq/wiki/Documents/MQL-Query-Guide.pdf>

5 Advanced Queries

A couple of advanced queries can be found here. You can study them and learn from them how to translate your exegetical questions into electronic queries.

The concrete phenomenon triggering the exegetical question	The query	The result
<p>A. Word level (Jer 1:11-12): Really a rhetoric device?</p> <p>וַיְהִי דְבַר יְהוָה אֵלַי לֵאמֹר מַה-אַתָּה רֹאֶה יְרֵמְיָהוּ וְאָמַר מִקֵּץ שָׂדֶה אֲנִי רֹאֶה׃ וַיֹּאמֶר יְהוָה אֵלַי הִיטַבְתָּ לְרֹאֹת כִּי-שָׂדֶה אֲנִי עַל-דְּבָרִי לַעֲשׂוֹת׃</p>	<pre>select all objects where [clause [word AS gcl NOTEXIST [word sp IN (art,conj,inrg,prep)] [word AS samelex]]] [clause]*{1-2} //allows for any clause to be repeated 2 times [clause [word FOCUS g_cons = gcl.g_cons NOTEXIST [word sp IN (art,conj,inrg,prep)] NOTEXIST [word lex = samelex.lex]]]</pre>	<p>SHEBANQ query no1002</p>
<p>B. Phrase level (Gen 4:1): Did Eve bore YHWH?</p> <p>וַתֹּאמֶר קָנִיתִי אִישׁ אֶת־יְהוָה</p> <p>יהוה as complement: "I gained a man with the LORD." יהוה as apposition to אִישׁ: "I gained a man, the LORD."</p>	<pre>select all objects where [clause [phrase function = Objc [phrase_atom FOCUS rela = NA] [phrase_atom FOCUS rela = Appo [word lex = ">T"]]]]</pre>	<p>SHEBANQ queries no946 no947 no948 no1003</p>
<p>C. Clause level: To whom does YHWH speak?</p>	<pre>select all objects where [clause [UnorderedGroup [phrase function = Subj [word lex_utf8 = "אלהים/"] OR [word lex_utf8 = "יהוה/"] OR [word lex_utf8 = "אל/"]] [phrase function = Pred [word lex = "דבר/"] OR [word lex_utf8 = "אמר/"] OR [word lex_utf8 = "ענה/"] OR [word lex_utf8 = "קרא/"]]] [phrase function = Cmpl [[word lex_utf8 = "ל/"] OR [word lex_utf8 = "ל/"]] [word FOCUS sp = nmpr]]]</pre>	<p>SHEBANQ query no448</p>

D. Level of text-linguistics (Gen 20:9-10): Is Abraham silent?

Genesis 20:9-10 (NRSV)

9 Then Abimelech called Abraham, and said to him, "What have you done to us? How have I sinned against you, that you have brought such great guilt on me and my kingdom? You have done things to me that ought not to be done."

10 And Abimelech said to Abraham, "What were you thinking of, that you did this thing?"

Genesis 20:9-10 (BHW 4.18)

9 ויקרא אבימלך לאברהם ויאמר לו מה עשית לנו ומה חטאתי לך כי הבאת עלי ועל ממלכתי חטאה גדלה מעשים אשר לא יעשו עשית עמדי:

10 ויאמר אבימלך אל אברהם מה ראית כי עשית את הדבר הזה:



```
select all objects where
[
  [clause domain = "N"
    [phrase function = Pred
      [word
        [word lex = "DBR["]
        OR
        [word lex = ">MR["]
        OR
        [word lex = "QR>["]
      ]
    ]
    ..
    [phrase FOCUS function = Subj
      [word AS samesubject]
    ]
    ..
    [phrase FOCUS function = Cmpl
      [word AS samecomplement]
    ]
  ]
  [clause domain = "N"]* {0-1}
  [clause domain = "Q"]* {1-50}
  [clause domain = "N"
    [phrase function = Pred
      [word
        [word lex = "DBR["]
        OR
        [word lex = ">MR["]
        OR
        [word lex = "QR>["]
      ]
    ]
    ..
    [phrase FOCUS function = Subj
      [word lex = samesubject.lex]
    ]
    ..
    [phrase FOCUS function = Cmpl
      [word lex = samecomplement.lex]
    ]
  ]
]
]
OR
[
  [clause domain = "N"
    [phrase function = Pred
      [word
        [word lex = "DBR["]
        OR
        [word lex = ">MR["]
        OR
        [word lex = "QR>["]
      ]
    ]
    ..
    [phrase FOCUS function = Cmpl
      [word AS samecomplement2]
    ]
    ..
    [phrase FOCUS function = Subj
      [word AS samesubject2]
    ]
  ]
]
[clause domain = "N"]* {0-1}
```

SHEBANQ query
[no491](#)

```

]
OR
[
[clause domain = "Q"* {1-50}
[clause domain = "N"
  [phrase function = Pred
    [word
      [word lex = "DBR["
      OR
      [word lex = ">MR["
      OR
      [word lex = "QR>["
    ]
  ]
  ..
  [phrase FOCUS function = Cmpl
    [word lex = samecomplement2.lex]
  ]
  ..
  [phrase FOCUS function = Subj
    [word lex = samesubject2.lex]
  ]
]
]
OR
[
[clause domain = "N"
  [phrase function = Pred
    [word
      [word lex = "DBR["
      OR
      [word lex = ">MR["
      OR
      [word lex = "QR>["
    ]
  ]
  ..
  [phrase FOCUS function = Subj
    [word AS samesubject3]
  ]
  ..
  [phrase FOCUS function = Cmpl
    [word AS samecomplement3]
  ]
]
]
[clause domain = "N"* {0-1}
[clause domain = "Q"* {1-50}
[clause domain = "N"
  [phrase function = Pred
    [word
      [word lex = "DBR["
      OR
      [word lex = ">MR["
      OR
      [word lex = "QR>["
    ]
  ]
  ..
  [phrase FOCUS function = Cmpl
    [word lex = samecomplement3.lex]
  ]
  ..
  [phrase FOCUS function = Subj
    [word lex = samesubject3.lex]
  ]
]
]
]
OR

```

```

[
  [clause domain = "N"
    [phrase function = Pred
      [word
        [word lex = "DBR["]
        OR
        [word lex = ">MR["]
        OR
        [word lex = "QR>["]
      ]
    ]
    ..
    [phrase FOCUS function = Cmpl
      [word AS samecomplement4]
    ]
    ..
    [phrase FOCUS function = Subj
      [word AS samesubject4]
    ]
  ]
  [clause domain = "N"* {0-1}
  [clause domain = "Q"* {1-50}
  [clause domain = "N"
    [phrase function = Pred
      [word
        [word lex = "DBR["]
        OR
        [word lex = ">MR["]
        OR
        [word lex = "QR>["]
      ]
    ]
    ..
    [phrase FOCUS function = Subj
      [word lex = samesubject4.lex]
    ]
    ..
    [phrase FOCUS function = Cmpl
      [word lex = samecomplement4.lex]
    ]
  ]
]

```

6 ETCBC in Accordance

7 ETCBC in Logos

8 ETCBC in Paratext

⇒ Download here: <https://1drv.ms/u/s!Al0u8U0ZQlv3ho4cRCXy-01qroquSQ>