

Drayton C. Benner  
Stephen Johnson  
SBL Annual Meeting  
Computer Assisted Research section  
Bible Software Shootout session  
November 21, 2009

## Introduction

Good evening. Before we begin, we would like to thank the Computer Assisted Research section for inviting us to present alongside our friends who develop Bible software for PCs and Mac, and we would also like to thank all of you for coming today to hear us speak about our Bible software for mobile devices. We are delighted to be with you.

**[To be included if the chair of the session does not introduce us:**

Let me first introduce us. This is Stephen Johnson. A software engineer by education and vocation, Stephen is Olive Tree's Chief Technology Officer. My name is Drayton Benner. By day--or perhaps more technically by academic year, I am a PhD student in Northwest Semitic Philology at the University of Chicago. By night--or really by summer, I do software engineering for Olive Tree.]

I will be doing the talking, but I am not very good at doing two things at once, and Stephen is much more nimble at the controls than I am, anyway, so he is going to be doing the driving in this software demonstration.

At Olive Tree, we make Bible software for a variety of handheld devices: iPhone and iPod Touch, Windows Mobile, Blackberry, Android, Symbian, and Palm. Our demo today will focus on the iPhone and iPod Touch, and many of the features you will see are brand new, having just been released this week. At the end of the demonstration, we will compare and contrast our resources and features on each platform that we support. Unfortunately, to my knowledge, there is no good way of hooking up an iPhone to a projector, so we are going to demo the software on Stephen's Mac, using the iPhone emulator produced by Apple. It will look basically the same as on an actual device, but of course the marvelous touch functionality of the iPhone has to be mimicked in other ways on the emulator.

Without further ado, let us jump into the specific items we have been asked to demonstrate.

## **SBL-Generated Questions**

**#1** (Give the parsing of a word and its meaning from a standard source.)

The first thing we have been asked to show you is the parsing of a word and its meaning from a standard source. We will show this for each of the Greek New Testament, the Septuagint, and the Hebrew Bible, for which we have the UBS, LEH, and BDB lexica, respectively. We hope that this will also give you a feel for the basics of the graphical user interface, and I will also highlight several things along the way.

First, let us show you how to do it for the Greek New Testament. Stephen will choose GNT-Mounce from the library. This is our parsed Greek New Testament. The text is that of NA27, with the parsing having been done by William Mounce and Rex Koivisto (my apologies if I mispronounced that). Stephen will navigate to the

beginning of Matthew. To bring up lexical and morphological information on a word, one simply taps on the word. Let's have Stephen tap on the first verb, ἐγεννησεν. This brings up a popup window on the main screen with our information. This popup window is placed out of the way of the word that Stephen selected, and the little triangular arrow indicates the word to which it corresponds. It begins with the lexical information and then gives the morphological information. It tells us that the lexeme to which ἐγέννησεν corresponds is γεννάω. It gives us a link to the appropriate entry in the UBS Greek lexicon, to which we'll come back momentarily, and it also gives us a gloss. Then, it gives the morphological codes together with an English interpretation of them. Now, if we click on the link to the UBS lexicon, we see that it takes us straight to the entry for γεννάω in the UBS lexicon. Having gained the desired information, we can simply click outside of the popup box to dismiss it.

Now, let us move on to the Septuagint. Stephen will choose LXX Parsed from the library. This is the standard scholarly electronic text, with the text coming basically from Rahlfs' edition and the parsing coming from the Kraft/Taylor/Wheeler morphology database. For

the sake of variety, we'll have Stephen navigate away from the beginning and go to Zechariah 4:2, where he will click on the word λαμπαδεῖον. We see the same kind of lexical and morphological information as we did with the parsed Greek New Testament. We see that the Kraft/Taylor/Wheeler database indicates that the lexical form is λαμπαδεῖον. However, when we click on our link to LEH, that is, the lexicon of the Septuagint by Lust, Eynikel, and Hauspie (my apologies if I am mispronouncing any of their names), we see that LEH has the lexical form as λαμπάδιον. Yet, this has caused us no trouble: we have resolved the discrepancy and gone straight to the appropriate entry in LEH. We see that the popup window does not contain the entire entry for λαμπάδιον. If we want to see the rest of it, we have a couple options available to us. We can simply scroll down in the popup window to see the rest of the entry, or, with the click of a button, we can choose either to replace LXX Parsed in the main window or bring the lexicon up in a second window. Stephen will do the former.

Now we are ready to see the Hebrew Bible. Stephen will choose BHS-Parsed. This is the Michigan-Claremont-Westminster database, so I suppose that BHS is a bit of a misnomer since it includes a

number of changes that will be reflected in BHQ. The first thing I want you to notice is the clarity of the text. The iPhone, like most mobile devices, does not handle complex scripts like Hebrew well. Yet, we have invested quite a bit of time and energy into making the Hebrew look beautiful. In my opinion, the two most aesthetically pleasing UNICODE Hebrew fonts available are the smart fonts Ezra SIL and SBL Hebrew. The iPhone does not actually support either of these fonts, but we have gotten around that limitation, and we are using Ezra SIL, the glyphs of which look almost identical to those of the print version of BHS. Unfortunately, we have to display it here on Stephen's Mac, but it really does look the same on an actual iPhone or iPod Touch. If you do not want to take our word for it, you will simply have to come by our booth to see.

Next, let us pick a word from Genesis 1:1, namely השמים. We see lexical and morphological information for both the noun שמים and also the definite article. Note that even though the definite article appears first in the text, we display the information for the noun שמים first because we think it is unlikely that you were trying to find

out about the definite article. In general, we place the information for particles after the information for verbs and nouns.

Before clicking on our link to BDB, let me make a couple brief comments about the gloss information. First, I have generally been unenthused by the quality of the Hebrew glosses I have seen in the past. These Biblical Hebrew and Aramaic glosses we are using are brand-new glosses that we recently commissioned from Chip Hardy, a fellow PhD student of mine in Northwest Semitic Philology at the University of Chicago, and they are of superb quality. Second, when showing glosses for verbs, in order to save some of the limited screen space available to us on mobile devices, we are showing only the portion of the gloss for the appropriate stem.

Now, let us click on the link to BDB. We see that though the Westminster parsing gave the lexeme using the form שָׁמִים, the only attested form, the editors of BDB filed it under the hypothetical form שָׁמִי, reflecting their belief that שָׁמִים is not a dual form but rather a plural form of a noun from a final-weak root. More generally, Westminster usually follows HALOT, which often differs

from BDB in the lexical form, either in terms of the actual form or in terms of the order of homographs, but we have reconciled them, and as you see here, we take you to the correct entry. As Stephen opens the entry in the split window, let me just note that our version of BDB is not one of the many electronic abridgements that are out there. We are using the same version digitized by Scribe that some of the others presenting here today are using. Our version is nearly unabridged; I simply have not gotten around to converting all of the languages (Arabic, Ethiopic, etc.) in the etymological sections.

**#2** (Show all the occurrences of a word in the NT and LXX and show the Hebrew word which corresponds with the Greek in the LXX (if there is a correspondence).)

We are now ready to move on to question #2, in which we were asked to show all the occurrences of a word in the New Testament and the Septuagint and then show the correspondence between the word in the Septuagint and the underlying Hebrew word it is translating. For this task, we have chosen the word *παρθένος*. Stephen will open the parsed Greek New Testament in the main window and the parsed Septuagint in the second window. With the

focus on the main window, he will then click the magnifying glass that brings up the search screen. We will show more of the advanced features of searching later, but for the time being, all we need to do to search the morphology database, rather than the text itself, is type in  $\pi\alpha\rho\theta\epsilon\nu\sigma$ , not worrying about accents or the final sigma. To do this, one can either use the built-in iPhone Greek keyboard, which was designed with modern Greek in mind, or one can use our Greek keyboard, which uses a layout more familiar to people in the English-speaking world. At present, we do not allow you to search multiple versions at once, so this will search just the Greek New Testament. This performs the search rapidly and displays the 15 hits, through which one can scroll, as Stephen will demonstrate. For each hit, it shows the entire verse, highlights the matching word, and provides a link to that verse in the parsed Greek New Testament.

Now, to search for the same word in the Septuagint, all we need to do is tap on the second window to give it the focus, then bring up our search screen again. It is already populated with the contents of the last search, so we simply hit "Search." This quickly fetches 67 hits and displays them. We can now see the results for both the New

Testament and the Septuagint, separated in different windows. Now, suppose we were interested in this search because of Matthew 1:23, our first hit in the main window, with its mention of a virgin, παρθένος, conceiving and giving birth to a son named Immanuel. We know that this is an allusion to Isaiah 7:14, so let us have Stephen scroll through the Septuagint search results all the way to Isaiah 7:14. Perhaps we want to see more context, and we want to see the Hebrew to which it corresponds. To do this, we will click on the link to Isaiah 7:14 in the search results list. Then, in the main window, we will bring up BHS-Parsed, which will automatically open to the same verse. Now, we do not have the Septuagint and the Hebrew Bible linked in a way that we can tell you explicitly which word corresponds to which. Those who do not know Hebrew can click on words to see glosses. However, for this scholarly crowd, that is not a problem. If you know Hebrew, you can read:

לֵבָן יִתֵּן אֲדֹנָי הוּא לָכֶם אֹת הַנְּהַה הָעֵלְמָה הָרָה וְיִלְדֶת בֵּן וְקָרָאת שְׁמוֹ עִמָּנוּ אֵל:

Clearly, *העלמה* corresponds to *παρθένος*. We can click on *העלמה* to see the gloss "young woman" or click through to the BDB entry to get a fuller treatment.

**#3** (Find all the occurrences of *οἱ δὲ* in Mathew's gospel followed by a finite verb within the clause.)

We are now ready to move on to the third question posed to us, namely the finding of all the occurrences of *οἱ δὲ* in Mathew's gospel followed by a finite verb within the clause. This one is a bit trickier in that it is asking for a search not only on the text or morphology but rather for one including the syntax. We do not presently have a syntax database for the Greek New Testament, although we are working on obtaining one, so we cannot give you the perfect search, but we can nonetheless show you how we would find all of the appropriate occurrences without the syntax database, anyway.

There are a few ways in which we could do this. Once we bring the parsed Greek New Testament back up in the main window, we proceed to the search screen. The easiest way we could do the search is simply to type in *οι δε*, set the option to search on the text

rather than the morphology, and click "search." This will bring up all of the occurrences of  $\text{οι δε}$ , through which we could look for those with a finite verb in the same clause. Note that at present, we do not allow the user to limit a search to a single book such as Matthew. One can limit the search by testament, but obviously that is not useful here. Under the hood, we actually do have the ability to set fine-grained limits on our search engine, but we have not yet updated the graphical user interface to allow the user to specify those limits.

However, if we are going to have to look through some verses by hand, we would at least like to have the finite verbs highlighted for us. So, let us return to the parsed Greek text and then to the search screen and enter a more complex search. While Stephen enters it, let me make a few comments about the challenges of developing Bible software specifically for mobile devices. Mobile devices have a number of limitations that desktops and laptops do not have. As mentioned earlier, mobile devices have rudimentary capabilities with regard to handling complex scripts, like those of Hebrew and Aramaic. We have faced that challenge and conquered it, producing aesthetically pleasing Hebrew and Aramaic. Mobile devices have far

less memory and processing power than desktops and laptops, which can be a problem with doing complex searches. We have faced this challenge and conquered it, allowing for quite complex searches that are fast not just on the emulator on Stephen's Mac but also on the actual mobile device. Finally, mobile devices have far less screen space than a typical desktop or laptop, and this presents challenges for a variety of display issues. We at Olive Tree have conquered many of these display issues, but we have not yet conquered these issues when it comes to search. We recognize that we need a better, graphical way of allowing users to specify complex searches, and we need a better, graphical way of allowing users to organize the search results. In time we fully expect to rise to this challenge, but we are not there yet. Truth be told, I am no good at designing and implementing graphical user interfaces. Thus, while I am pleased with the power and speed of the search engine I wrote, the graphical user interface leaves much to be desired. Thankfully, Stephen and other Olive Tree developers, who are all much better at designing and implementing graphical user interfaces than I am, have plans to rectify the situation.

((o@r%nmp oσ@spr%nmp)o δε)aow1 (v%??i\* v%??m\* v%??s\*  
v%??o\*)o)aov0

With that caveat, let me explain what Stephen has entered. This is a search on the morphological database; Stephen has indicated that in the search options. The first two terms correspond to the different ways in which we could have οἱ in the text: either from the article ο or from the relative pronoun ος. Of course, we never actually have οἱ δε with the οἱ coming from the relative pronoun ος, but we have included it here for the sake of completion. The lexeme is specified first, followed by an @ sign to separate it from the morphology codes. The "r" stands for "article." The % sign that follows is used to separate major sections of the morphology codes. Then, the "nmp" stands for nominative masculine plural. Note that the Mounce-Koivisto text only marks vocatives as such when they are morphologically distinguishable, so we do not need to worry about the vocative option. The relative pronoun is similar, with the "spr" standing for "substantive, pronoun, relative." These are enclosed in parentheses, followed by an "o", which indicates an OR search on the items inside the parentheses. Next comes the word δε. If we only wish to specify the lexeme and not the morphology codes, we

can leave the morphology codes off. After the close parenthesis, we see the codes "aow1". This means that we want our two words to be contiguous and ordered as specified. More specifically, the "a" stands for an AND search, over against the OR search we did earlier. The "o" stands for "ordered," as opposed to "n" for NOT ORDERED. The next part indicates the proximity for the elements within the parentheses. The "w" stands for "word," as opposed to "v" for verses. The "1" indicates that the last word can be no more than one word after the first. That is, they must be contiguous. The next set of elements are an OR search, as indicated by the "o" after the parentheses surrounding them, and they correspond to the different types of finite verbs. Verbs all start with v%, then they give the tense, voice, and mood. The "?" we are using is a single-character wildcard, so by using two of them we are indicating that we want to allow any tense or voice. The next character is either "i" for indicative, "m" for imperative, "s" for subjunctive, or "o" for optative. Thus, this OR search will find any finite verb. Finally, we have our searches for οὐ δε and finite verbs in parentheses with "aov0" after it. This means that it is an AND search, it is ordered, and they have to all be at most zero verses apart. That is, they need to be in the same verse.

This is complicated to enter, but it does accomplish the purpose of highlighting any finite verbs later in the verse after we have  $\omicron\iota\ \delta\epsilon$ . We still have to check by hand to see whether they are part of the same clause, but our job is easier now. Note also that the figure of 336 hits signals 336 matches to our search, not 336 separate verses; a verse with 5 finite verbs after  $\omicron\iota\ \delta\epsilon$  counts as 5 toward that total.

Let me show you one more way in which we could do this search. Instead of requiring that the finite verb be in the same verse after  $\omicron\iota\ \delta\epsilon$ , we could have specified that the finite verb needed to be within a certain number of words of  $\omicron\iota\ \delta\epsilon$ . For this, all we need to do is change the very end of our search query to indicate the number of words for this AND search rather than the number of verses. Let's allow 10 words to separate the beginning of  $\omicron\iota\ \delta\epsilon$  and the finite verb under the assumption that the clause is usually not longer than that.

```
((o@r%nmp oσ@spr%nmp)o δε)aow1 (v%??i* v%??m* v%??s*
v%??o*)o)aow10
```

There are a bit fewer results now, but notice that some results have actually been added. This type of search ignores verse boundaries, a feature that can be extremely useful. When doing searches, I very often care about proximity but often do not care at all about the proximity in terms of often arbitrary verse divisions. Notice, for example, as Stephen scrolls through the results, how οὐ δε in Matthew 9:14 matches a finite verb later in Matthew 9:14 and also matches a finite verb at the beginning of verse 15. We could also require the finite verb to be both in the same verse and within 10 words of the start of οὐ δε, but we will not take the time to show that now.

**#4** (I want to study a part of speech, e.g., demonstrative pronouns or interjections. How do I get all of the lemmas for that part of speech, get all the occurrences of those lemmas, and the results organized in such a way that I could write an article/monograph on that part of speech from the data?)

Rather, we will move on to the fourth question that we have been asked. The fourth and fifth questions that we have been asked are designed to help you write an article or a monograph after analyzing

the results from a search. To be honest, while we at Olive Tree are excited about what mobile devices can do in the present and are even more optimistic concerning what they will do in the future, we presently concede that you will probably be doing your primary research in preparation for writing an article or a monograph using Bible software on your laptop, using the software written by our friends at the other Bible software companies that have already presented here today. However, we do not see that as an excuse for not giving you tremendous power at your fingertips on the go, anyway!

For this fourth question, we were asked to study a part of speech. Since the next question is Hebrew-based, we will stick with Greek for this one. However, since I am a Semitist, I could not resist throwing in some Hebrew and Aramaic. Let us imagine that we want to write an article on words in the New Testament that are transliterated from Hebrew or Aramaic. Stephen will punch in the search:

q%h q%a

We just need to specify the morphology codes. The letter "q" stands for "particle," and the "h" and "a" stand for "Hebrew" and "Aramaic," respectively. Then, we need to adjust our search options in two ways. First, we will make it an OR search. We can do this without going to the search options page by using parentheses and an "o," but it is perhaps easier to select it from the options screen. We also want to change another setting. Heretofore, we have been displaying our search results in the order in which they appear within the canon. However, in this case, it would be more helpful to arrange them by lexeme. Stephen will check "Hit Word" under "Sort By" to accomplish just that. This will actually sort the results first by lexeme, second by morphology codes, and third by location in the canon. When we do so, we see 175 results, starting with  $\alpha\beta\beta\alpha$ , then moving on to  $\alpha\lambda\lambda\eta\lambda\omicron\upsilon\iota\alpha$ ,  $\alpha\mu\eta\nu$ , etc.

**#5** (I want to study the inflections of the Hebrew middle weak verb, and I want to see what the range of possible variations are for each of the conjugations (perfect, imperative, etc.), person, number, gender, stem. This means I need to find all the middle weak verbs, find all their occurrences, and organize them in such a way that the variations of their inflections are immediately apparent. The goal of

the data organization would be to allow me to write an article about variations of the Hebrew middle weak verb.)

The final question we have been asked to demonstrate concerns finding all of the middle weak verbs in Hebrew and organizing them in some way corresponding to their morphology. Stephen will take us back to our parsed Hebrew Bible, and he will enter the appropriate search string:

ו?@v\* י?@v\*

Recall that the "?" is a single character wildcard. The "@" separates the lexeme from the morphology codes. The "v" in the morphology codes stands for "verb," and the "\*" at the end of each term is a generic wildcard. On the options screen, he will double-check that we have this set as an OR search, and he will also change the sorting method. For this search, we will sort by "Hit Morphology." This will sort first by the morphology codes, second by lexeme, and third by location within the canon.

When we do the search, we get 13,854 results. Our first hit is מוֹבְאֹת, a hophal feminine plural participle from the root בּוֹא. The morphology codes are sorted by alphabetical order, and the code for hophal is the first alphabetically among the stems. Then, the code for the participle comes alphabetically before that of the perfect, imperfect, etc., and on it goes down the line. As Stephen scrolls through the first results, all of which are hophal participles, of course, I'll mention that we hope to provide better ways for you to arrange these results in the future.

### Self-Generated Question

**[If time permits:** Since we have a little more time, we will show one more search that shows a bit of the flexibility of the search engine. In this search, we'll look for some cosmic imagery near a verb of creation. More specifically, we will look for אָרֶץ or שָׁמַיִם within 15 words of שָׁמַשׁ, יָרַח, or כּוֹכָב, and then both of these need to be in the same verse or a verse neighboring either the verb בָּרָא or the verb עָשָׂה. However, we will also suppose that we do not know all of the

lexical forms precisely. Suppose we do not know whether the lexical form for שמים is שמי or שמים, and suppose we do not know whether the *mater lectionis* ם is present in the lexical form כוכב. We will make use of wildcards for these. Further, suppose that we are having a bad day as Anglophones and cannot even remember whether the verb עשה begins with an ע or an א. We will use a single-character wildcard for the first letter of this verb. For the sake of time, we will let Stephen copy the search into the search entry field. Typing mixed right-to-left and left-to-right can confuse the iPhone a bit, but as long as we type it in the logical order, the search will be done correctly.

```
((@nc*שמ*)o (כ*כב ירח שמש)o)anw15 (?@v*ברא)anv1
```

Stephen is actually going to do this search not only on the emulator but also on an actual iPhone to show that even on the iPhone, the search is fast. We typed it in beforehand on the iPhone. You see that we get 23 results in no time flat on the emulator, and then Stephen will give us a signal when he starts the search on the iPhone and again when it finishes.]

## **Availability of Greek, Hebrew, and Aramaic Resources and Features**

This concludes our iPhone demonstration. We at Olive Tree are always working to improve our product and deploy it across as many platforms as possible. We'll put up a chart displaying the Greek, Hebrew, and Aramaic resources we have available for each platform and also which features are available on each. Basically, we have highlighted what is available on iPhone and iPod Touch. Right now, none of this is available on Blackberry or Android, but all of it will be available soon. On Windows Mobile and Palm, we have most of the same Greek, Hebrew, and Aramaic resources available as on iPhone and even have the Qumran non-biblical texts available on them, which we do not yet have available for iPhone. Of course, we also have Bibles in English and other languages, reference works, and e-books for your theological library. All of these resources, on all of the mobile platforms, run locally on the device itself. You do not need an Internet connection for any of them, meaning that they are accessible wherever you are, and you do not have to wait for a potentially slow Internet connection. Now we will put up another chart showing the features available on the various platforms. The

Windows Mobile and Palm versions do not have all of the same features as the iPhone version. Most notably, morphological searching is not available on Windows Mobile or Palm.

## **Conclusion**

In conclusion, we hope that you have enjoyed seeing this demonstration of Olive Tree's scholarly resources and that you will come talk to us afterwards or come by our booth, booth #808, to see a demonstration on actual mobile devices. Mobile devices are taking on more and more functions every year, and now one of these new functions is the advancement of biblical scholarship.