Article

The ETCBC Database of the Hebrew Bible

Cody Kingham
https://orcid.org/0000-0002-8625-6773
Vrije Universiteit Amsterdam
codykingham@icloud.com

Wido Van Peursen
https://orcid.org/0000-0002-3142-5752
Vrije Universiteit Amsterdam
w.t.van.peursen@vu.nl

Abstract

We provide a brief introduction to the history, methodology, and tools of the Eep Talstra Centre for Bible and Computer (ETCBC). The ETCBC maintains a searchable database of morphology, syntax, and text-level features for the Hebrew Bible, Hebrew inscriptions, Dead Sea Scrolls, the Peshitta, and one of the Targumim. The ETCBC follows a form-to-function approach, in which surface-level features are registered first and functional labels second. Linguists and exegetes can use the database's freely accessible query tools for pattern searches and analysis of the text's structure to address their research questions

Keywords: ETCBC; Eep Talstra; Hebrew Bible; Hebrew database; linguistics; textlinguistics

Introduction

The ETCBC's now forty-year existence might be characterised as an ongoing practicum in methodology. Founded in 1977 as the Werkgroep Informatica Vrije Universiteit (WIVU), the small research centre in Amsterdam set out to build simple concordance and text searching programs to aid the process of biblical interpretation.1 Within a few years, though, the focus had developed into a wider examination of the process of interpretation itself (Talstra 1986). Automatic processing of the biblical texts stimulated reflection on the way information from a text is naturally conveyed. Just what is the best procedure to follow when reading an ancient text? Talstra and his colleagues answered broadly that the particular should precede the general.2 For interpretation, this meant

1 For a fuller description of the ETCBC's history, see Oosting (2016).

2 "One may expect that the exegete and the linguist can, each on his own way, benefit from the use of these materials. An exegete, interested in the explanation of a special text, will ask for general linguistic data, lexical and syntactical parallels that may clarify his text. The general information contributes to

UNISA university of south africa PRESS

Journal for Semitics
https://upjournals.co.za/index.php/JSEM
Volume 27 | Issue 1 | 2018 | #2974 | 13 pages

https://doi.org/10.25159/1013-8471/2974
ISSN 0256-8853 (Print)

that good reading begins with good linguistics. For linguistics, it meant beginning with the smallest units of meaning and meticulously analysing the text up to its largest units. It is that intensive focus on sequence that kept the research centre linked to the computer.

The computer enabled the WIVU to approach linguistic analysis of the Hebrew Bible as a formal protocol. Programs were written which, when provided with input, could parse Hebrew words and constructions based on rules and pattern recognition (Talstra 1986). The WIVU stored the output of its programs in a growing database that eventually covered the entire Hebrew Bible. The process enabled the exposure of workings and tendencies in the language that were previously neglected. Traditional grammars, for instance, had answered many questions about the functional categories of words, phrases, and clauses. Yet seldom did they explain the details of how the components combine into the larger, discrete units, such as phrases into clauses (Talstra 2004, 9). By essentially teaching the computer to parse Hebrew constructions, the WIVU produced a database in which the resulting data was linked to the mechanisms of the language itself.

In 2013, the WIVU was renamed the Eep Talstra Centre for Bible and Computer (ETCBC) in honour of its founder (Oosting 2016, 206). In the years since, it has continued to expand and update its database and analytical tools. The active reflection on methodology also continues, only encouraged by a world where every discipline is now being transformed under the digital revolution. The quest for patterns, truly at the heart of both linguistics and exegesis, is enabled as never before (Bod 2013). For the remainder of this essay, we describe the present state of the ETCBC database, its coverage, its methodology, and the representations of data available to researchers. Finally, we conclude with some reflections on the future direction of the ETCBC.

Corpora and Contents

The ETCBC database encompasses a growing number of Hebrew, Aramaic, and Syriac corpora for which linguistic data is stored. The largest analysed corpus is the complete Hebrew Bible, including the Aramaic portions. The database has recently added the Qumran documents of 1QM (War Scroll), 1QS (Community Rule), some Hebrew epigraphic texts, including the Mesha Stela, the Siloam Inscription, the Balaam text of Deir ʿAlla, the inscriptions from Kuntillet ʿAjrud, the ostraca of Arad, Lachish, and Meṣad Ḥashavyahu, and the amulet of Ketef Hinnom, and a few Tannaitic texts: the Mishnah tractate *ʾAbot* and the *Parasha Shirata* from Mekilta d-Rabbi Ishmael.3 There

the explanation of the particular. On the contrary, a linguist, interested in the grammatical functions of a set of clauses, will try to establish general rules on the basis of the observed linguistic phenomena. This analysis goes from the particular to the general" (Talstra et al. 1983, 28).

3 The extra-biblical Hebrew materials were prepared during the ongoing project, "Does Syntactic Variation Reflect Language Change? Tracing Syntactic Diversity in Biblical Hebrew Texts"

is also data for the Aramaic Qumran fragment 4Q246 ("Son of God").[4] For Syriac, the database contains the Peshitta to Kings, Judges, Ben Sira, Prayer of Manasseh, Epistle of Baruch, and the Book of the Laws of the Countries.[5] Initial work in the Targumim has produced data for the Targum of Jonathan on Judges. Forthcoming analyses include 1QH[a] (Hodayot), 1QpHab (Pesher to Habbakuk), and 11QT[a] (Temple Scroll).[6]

For each corpus, the ETCBC stores data for the word, phrase, clause, sentence, and text levels.[7] Data on the word level includes orthography, morphology, part of speech, basic lexical categories, and phrase-level function. Phrase and clause units bear functional and relational descriptions. The clause contains further designations for clause type (based on the main verb form and the order of clause constituents), domain and text (narrative, discourse, and embedded variants), and kind (verbal, nominal, and non-predication). The only data stored for sentences is their boundaries.[8] The database also contains a number of experimental, text-level features. Clause hierarchy contains proposed relations between clauses based on the inclusion of conjunctions or similarities in kind or structure.[9] Another, experimental feature is the paragraph, stored on the clause as a number, which reflects units identified based on the continuation of actants (Talstra 1997, 92, 97–100).

Methodological Principles

The ETCBC employs two guiding principles in its encoding of ancient texts: form-to-function and bottom-up analysis. The form-to-function method foregoes commitment

---

supervised by Janet Dyk and Wido Van Peursen, with the researchers Dirk Bakker, Marianne Kaajan, and Martijn Naaijer. The project will conclude in the fall of 2017 with a synthesis of the three monographs. For a sample result from this project, see Naaijer and Roorda (2016).

4 4Q246 was prepared during Cody Kingham's 2016 independent research project, "The Syntactic Structure and Message of 4Q246, the 'Son of God' Manuscript", with the eventual goal of developing it further into an article.

5 The Peshitta material was encoded during the 1999–2005 project titled "Computer Assisted Linguistic Analysis of the Peshitta" (CALAP) led by Konrad Jenner and Eep Talstra with the researchers Janet Dyk, Percy Van Keulen, and Wido Van Peursen (Van Peursen 2005). On the project methodology, see Van Peursen (2007, 159–75). The project was succeeded by the 2005–2011 project titled, "Turgama: Computer-Assisted Analysis of the Peshitta and the Targum: Text, Language and Interpretation", led by Wido Van Peursen with the researchers Dirk Bakker and Percy Van Keulen, during which the Aramaic Targum to Judges was encoded alongside the Syriac Book of the Laws of the Countries (Peursen 2005). Constantijn Sikkel and Hendrik Jan Bosman worked as scientific programmers for the projects.

6 Prospectively to be finished during the final months of the syntactic variation project.

7 For a full listing of the features alongside some brief descriptions, see https://etcbc.github.io/text-fabric-data/features/hebrew/etcbc4c/0_overview.html

8 The sentence often agrees with the clause boundaries, but, unlike the clause, it is able to accommodate larger constructions such as clauses further expanded by infinitive clauses or relative clauses (Talstra 2004, 17).

9 See the end of the following section for a fuller description.

to a linguistic theory until after the data is collected and analysed (Van der Merwe 1994, 16–7; Van Peursen 2015, 302–3). The theory about a construction's function, then, can develop from the data (so "form-to-function"). Of course, the commitment to no theory is in itself a kind of theory, one which only works if structure and function in a text are directly connected.[10] That connection need not exclude more cognitive or semantic sources for function. But it does mean that structure must first be fully exploited before advancing to other forms of analysis (Talstra and Van der Merwe 2002). By emphasising pattern recognition and withholding judgment on linguistic functions, the ETCBC is also able to provide a database that is simultaneously useful to adherents of various methods (Talstra 2004, 1–2).

The practical application of the form-to-function approach is achieved through bottom-up analysis. This method first processes smaller constructions and then utilises their combinations in the calculations of larger units. The process thus moves progressively from words to phrases, from phrases to clauses, from clauses to sentences.

Beginning at the word level, morphological data is encoded rather than tagged. Tagging implies labelling the surface of a word or construction with a morphology code; encoding, on the other hand, registers the individual morphemes in a given word: prefixes, lexeme, distinguishing vowels, and suffixes (Verheij 1994, 32–40). To do this, words in the ETCBC are transliterated. Verbal prefixes are marked with enclosing exclamation marks, e.g., !J! is the prefixed *yodh* in יקטל. Verbal stem prefixes are marked with enclosing brackets. Verbal or nominal suffixes are marked after the lexeme with an opening bracket or forward slash, respectively, e.g., "[WN" as the suffix in יקטלון, or "/JM" as in דברים. A complete example, using the 3MP *yiqtol* of the verb קטל, would appear as !J!QVL[WN,[11] or the masculine plural of דבר as DBR/JM (Verheij

10 The theoretical foundation of the ETCBC method ties back to Wolfgang Schneider and Harald Weinrich (Talstra 1992a), both of whom could be classified as structuralists. Structuralism has indeed fallen on hard times since the rise of Chomsky's generative grammar. Yet structuralism continues its influence through textbooks and a handful of researchers (Matthews 2001, 142–53). Recently, Lazard has called for a return to Saussure's distinctions between *langue* (language) and *parole* (speech), so that language as a system of signs, and the conditions of its use in discourse, are considered separately (Lazard 2012, 242). His call does not discount the value of cognitive approaches. Rather, following Granger, he casts Saussure's program as an intentional reduction of the complexity of language, which allows it to be studied as an "abstract 'scientific object'" (Lazard 2012, 243). "The decisive step in any scientific approach to an experience, and the *sine qua non* requirement for starting it, is to form concepts determining the most general shape of the objects of that experience, in such a way that it will be precise, but only approximate, knowledge" (Granger 1992, 30–1 as translated by Lazard 2012, 243). See also Croft (2001) and Haspelmath (2007) and their recent rejections of the generativist concept of general language categories.

11 See https://shebanq.ancient-data.org/shebanq/static/docs/ETCBC4-transcription.pdf for a transliteration table for converting the Hebrew characters into ETCBC representation. The morphological encoding also includes an option to include distinguishing vowels which are necessary for the parsing.

1994, 28–31). The program then applies rules to the encoded words to calculate the parsing from their aggregate (see Verheij 1994, 48–56 for the rules). A final step looks up the part of speech data for the word in a lexicon file.[12]

A similar process of applying rules to registered patterns enables the encoding of phrases, clauses, and sentences. A phrase program examines the morphology and lexical data of a string of words, compares them against a list of phrase combinations (a user-made phrase set), and proposes boundaries based on recognition of a listed pattern, e.g., preposition + definite article + noun = phrase (Talstra 2004, 15–6). As the user encounters and defines new patterns, they are added to the phrase set to be used if the pattern is re-encountered. For clauses, the program deploys a list of recognised phrase patterns. For instance, the program might define the boundaries of a clause by examining the grammatical type of each phrase on either side of a *waw* conjunction. If they are different, e.g., prepositional phrase + *waw* + nominal phrase, the program can demarcate the end of one clause and the beginning of another (Talstra 2004, 16; see Gen 22:1).

Processing grammatical units in the manner described above requires a distinction between distributional and functional units. Distributional units are identifiable through formal pattern recognition; functional units are the traditional grammatical units of phrases, clauses, or sentences, which require a level of human interpretation. The need for distributional units can be illustrated by the example below (adapted from Talstra 1992b, 136):

Gen 3:11b	1	המן העץ
Gen 3:11b	2	אשר צויתיך
Gen 3:11b	3	לבלתי אכל ממנו
Gen 3:11b	4	אכלת

A parsing program might recognise the beginning of the enclosing clause, the embedded אשר-clause, the embedded לבלתי-clause within the אשר-clause, and the end of the enclosing clause. But the intervention of the אשר and לבלתי clauses separates part 1 (המן) from the last part of the sentence (part 4, אכלת). In this case, all the parts must be independently registered before they can be assimilated into a completed sentence. To solve this issue, the ETCBC introduced the distributional units of "atoms". So, in the example, the numbered "parts" become "clause atoms". The atoms are then connected

12 For this task, the program uses a lexical file derived from KBL (Koehler and Baumgartner, second edition). The ETCBC encoding method also includes a way to distinguish between paradigmatic letters and realised letters in a lexeme, in order to account for irregular forms. Non-paradigmatic realised letters are indicated with an ampersand (&) while the non-realised paradigmatic letters are indicated with an opening parenthesis mark. Homographic morphemes are distinguished by adding any number of equal signs, depending on the number of homographs (Verheij 1994, 31, 58).

in a subsequent analysis to form the clause. The ETCBC database model contains not only clause atoms, but also phrase atoms and sentence atoms. The atoms are preserved alongside their functional aggregates, the phrases, clauses, and sentences.[13]

At the top of the bottom-up progression is the clause relations (also clause atom relations), which is a hierarchy of those units based on features derived from the lower level analyses (Talstra 1997). The ETCBC labels independent clauses as "mother", while their dependents are called "daughters". The less-experimental aspect of clause relations depends on the presence of conjunctions or relative particles that explicitly relate the clause back to the previous clause. But the ETCBC program also searches for similarities between relationships that are less obviously marked, including closeness of verb forms, continuity of an indicator for person, number, and gender, and the frequency of previous clause connections made between the two candidate clause types (Talstra 1992b, 141–2).

Versions and Representations

The database of the ETCBC is available through a growing number of representations, all derived from the core database stored in the ETCBC server at the Vrije Universiteit Amsterdam. The elementary representation in the server is a series of plain text analysis files containing word, phrase, or clause data.[14] These files are created and utilised by the data creation programs.[15] The files generate a larger plain text file that contains the analysis for an entire book (Quest Data File or QDF).[16] The data is stored in rows and columns, in which the rows represent individual words in the book, and the columns contain the linguistic data, including which phrase or clauses the words belong to.17 From the QDF files, the ETCBC generates a database dump for the entire corpus, encoded in extended monad-dot-feature (EMdF) model using software called Emdros, and queried using Mini Query Language (MQL).[18]

---

13 Functional units and distributional units ("atoms") often overlap in the cases where the distributional units are uninterrupted by embedding elements.

14 The analysis files are columnar, often containing numbered codes for the analysis data.

15 These files and programs are the subject of the first author's current master's internship, in which he will prepare a detailed explanation of the more technical aspects of how they function. See, though, Van Peursen (2007), for a detailed description of the adapted programs for Syriac.

16 The Quest Data File Format was originally developed for exchanging data with the University of Greifswald (Germany) during the Quest II project in the late 90s (see Sikkel (2014) for full description of the form). Originally intended to be a Bible study tool for a broader audience, the project never got off the ground (Oosting 2016, 200).

17 QDF does not directly store data on units larger than words. Rather, each word has a unique identifier that corresponds to an abstract unit that contains it. By combining all the words with that same identifier, a phrase, for example, can be constructed. See Sikkel (2014).

18 Crist-Jan Doedens developed EMdF during his 1994 dissertation, "Text Databases. One Database Model and Several Retrieval Languages". The model is now most prominently maintained by Ulrik

EMdF is the primary data representation for storing and querying linguistic data in the ETCBC database. In EMdF, linguistic constructions, or "objects", are defined as ranges of words, or "monads". In the table below, the monad numbers correspond to words:

Table 1: Gen 1:1 in EMdF with ETCBC transliteration

Clause	1										
Phrase	1		2	3	4						
Words	B	R>CJT	BR>	>LHJM	>T	H	CMJM	W	>T	H	>RY
Monad	1	2	3	4	5	6	7	8	9	10	11

Linguistic objects in the database are defined using ranges of monad numbers. Thus, in the example above, clause object 1 consists of monads 1–11; phrase object 4 consists of monads 5–11; etc. When paired with a query language (MQL), the data model allows a user to enter a search that is isomorphic with its appearance in the text. For instance, to look for a clause that contains two proper nouns separated by a *waw* conjunction, one can enter:[19]

```
SELECT ALL OBJECTS WHERE
   [clause FOCUS
      [word pdp = nmpr]
      [word lex = "W"]
      [word pdp = nmpr]
   ]
```

While this example is intentionally simple, the search capability afforded by MQL combined with the numerous features of the ETCBC database allow for powerful searches. Some examples are: finding all first-person verbs within narration,[20] all one-

Peterson's Emdros software (www.emdros.org). See Peterson (2004) for a helpful introduction of Emdros and its query language, MQL.

19 "SELECT ALL OBJECTS WHERE" syntactically marks the beginning of the query. The boundaries of the linguistic objects being queried are represented with the opening and closing brackets. The first word after the opening bracket defines the object. The "FOCUS" tag tells the query which element to highlight in the results. Other tags that follow the object tag, such as "pdp" (phrase dependent part of speech) or "lex" (lexeme) are optional features; the equal sign specifies what value these features should hold. The results of this simple query can be viewed at https://shebanq.ancient-data.org/hebrew/text?iid=1684&page=1&mr=r&qw=q.

20 C. Kingham, https://shebanq.ancient-data.org/hebrew/text?iid=1050&page=1&mr=r&qw=q.

word clauses within discourse,[21] all instances of the negation אין with a pronominal suffix followed by a participle,[22] or all uses of "Abraham" as a subject.[23] MQL additionally allows for gapping (of specific or non-specific distance) and other specifications. With these tools, the combinations of search patterns one can find are only limited by the imagination.

Based on the core database, the 2014 SHEBANQ project brought the ability to run MQL queries in the Hebrew Bible to an interactive website.[24] The website also made the ETCBC data accessible online for the first time. The tool focuses on Hebraists, biblical scholars, or students who are interested in querying or learning to query the ETCBC's representation of the Hebrew Bible. SHEBANQ presents the results of a query within a text viewer. With a simple toggle, users can also see others' queries.25 This not only provides illustrative examples for learning MQL, but it also gives opportunities for serendipity, where one researcher might come across a query that inspires further thought. Each search in SHEBANQ receives a persistent identifier. Through it, scholars can cite their data specifically (and the MQL code that found it); their reviewers can then access that data directly.

Some research, though, requires more advanced searches for which there are no limits on how data is processed and sorted. The ETCBC's Text-Fabric brings linguistic analysis of ancient Hebrew into the cutting-edge of data science. It provides a full representation of the Hebrew database (including the coveted Dead Sea Scrolls and inscriptions) in a Python package.[26] The implementation in Python allows searches in the data to be combined with an abundant library of packages created specifically for data science. These tools enable advanced statistical analysis and the visualisation of data through illuminative graphs, tables, and charts. Jupyter notebooks let researchers

---

[21] M. Kaajan, https://shebanq.ancient-data.org/hebrew/text?iid=562&page=1&mr=r&qw=q.

[22] M. Naaijer, https://shebanq.ancient-data.org/hebrew/text?iid=1410&page=1&mr=r&qw=q.

[23] O. Glanz, https://shebanq.ancient-data.org/hebrew/text?iid=490&page=1&mr=r&qw=q.

[24] SHEBANQ (pronounced "she-bank") stands for "System for Hebrew Text: Annotations for Queries and Markup". The project was funded via CLARIN (https://portal.clarin.nl/node/4180). The website is https://shebanq.ancient-data.org.

[25] See https://shebanq.ancient-data.org/hebrew/text, select a text, and click on "show queries".

[26] Text-Fabric is the successor to LAF-Fabric (Linguistic Annotation Framework), both developed by Dirk Roorda. It currently contains a version of the data that is more up-to-date than the one in SHEBANQ (ETCBC 4c). However, there are plans to create a direct link between the Text-Fabric data and the core database, so that the data is as updated as the internal server data. See https://github.com/ETCBC/text-fabric; see also Roorda, Kalkman, Naaijer, and Van Craneburgh (2014).

write and present their work in an interactive format that provides direct access to the code in a modular way (via interconnected cells of code interspersed with written notes in mark-up, graphs, tables, or other output). These state-of-the-art abilities offer the opportunity for complex pattern searches and analysis at a level previously only seen in the so-called hard sciences.[27]

The introduction of the open-source ETCBC database materials to the online sphere has fostered a number of other useful implementations. Bible Online Learner is a learning tool based on SHEBANQ for students of Biblical Hebrew.[28] The tool follows a principle of corpus-based learning from the Hebrew Bible, wherein the students receive vocabulary, parsing, and translation exercises within the context of a biblical passage ("text as tutor"). The utility includes a number of helpful tools for Hebrew teachers to track the progress of their students; and it brings the possibility of flipped classroom instruction, where the class time focuses on actively practicing and enjoying the material while the theory-end is learned out of class through instructive videos or reading.[29] Other undertakings worth mentioning are the Tiberias Project at Bar Ilan University, which is a computational authorship attribution and stylistic analysis tool for the Hebrew Bible,[30] the implementation of the data in Paratext (for Bible translators), and the ETCBC database packages now available in Logos and Accordance for general use.[31]

Recently, individual users have begun to insert the data into their own software applications, with user-friendly access to the morphological and syntactic data. Examples include Eliran Wong's Bible Reader (text viewer, mouse-over morphology),[32] ETCBC Remix (an Android port),[33] and qBible (text viewer with word association queries and point and click morphology).[34] The accessibility to SHEBANQ and Text-Fabric, along with the recent user-created applications, has generated a small online community, via a Slack (messenger) group, where users share their work or

---

27 For illustrative examples on the kind of data that Text-Fabric is already being used to process, see Roorda and Naaijer (2016) and Kingham (2017).

28 See the Bible Online Learner page: http://bh.3bmoodle.dk.

29 See http://bh.3bmoodle.dk/mod/page/view.php?id=622.

30 Tiberias is by Joshua Berman and Moshe Koppel. The project relies on the ETCBC data archived at Data Archiving and Networked Services' (DANS) Easy service at https://doi.org/10.17026/dans-2z3-arxf.

31 See Logos, https://www.logos.com/product/18617/german-bible-society-bundle; and Accordance, https://www.accordancebible.com/ETCBC/.

32 Eliran Wong, http://eliranwong.com.

33 Eliran Wong, https://github.com/eliranwong/ETCBC-remix.

34 James Cuénod, http://qbible.tk.

receive help in MQL or Text-Fabric.[35] The members of the community range from those who have no familiarity with coding to those who are professional programmers.

Future of the ETCBC

With 2017 being the fortieth anniversary of our founding, we have occasion for reflecting briefly on the future of the ETCBC. Our upcoming celebration of that founding, for example, has received the name "ETCBC≥40", with the "greater than or equal to" putting attention on looking forward. What, from this point, seems to be the future direction of the Eep Talstra Centre?

From a methodological perspective, ongoing research and additions to the data suggest a greater emphasis on semantics. Recent research on verbal valency, or the tendency of certain verbs to attract certain modifiers (objects or adjuncts), has sought to show how the form-to-function approach can be utilised successfully in determining which sense a verb carries.[36] Often translators or exegetes will select a meaning for a verb, without being aware of the syntactic patterns that condition it (Dyk, Glanz, and Oosting 2014, 1–2). This research has also indicated shortcomings in the current data model, such as the need to distinguish between valence relations, e.g., complements and adjuncts, and grammatical relations, like subject, predicate, or direct object (Dyk, Glanz, and Oosting 2015, 40–1). Another sign of greater semantic emphasis includes current research on participant tracking, which aims to answer how actants are conveyed on and off the "stage" of the text (Talstra 2016, 239–242, and Talstra and Erwich, forthcoming).[37] One of the goals of the ETCBC has always been to exploit the data to the furthest extent possible. As research moves increasingly towards the realm of semantics, it remains to be seen where those farthest extents may lie.

Finally, the implementation of data processing tools such as Python and Jupyter notebooks, along with the introduction of the data to the internet, suggests a more interactive future for the ETCBC, and one more thoroughly in the realm of data science. No longer is the database confined to the servers of a university storeroom. Access to it enables replicability for linguistic and exegetical findings, necessary for the proliferation of research, and helpful for moving beyond outdated, intuition-driven research in biblical studies. The ability to apply complex algorithms and procedures in the processing of data brings the process of data discovery (and perhaps, in the future, data creation!) to a level previously reserved for the more mathematical sciences. These

35 For an invitation to the Slack group, e-mail shebanq@ancient-data.org.

36 See Dyk, Glanz, and Oosting (2014 and 2015); and Femke Siebesma-Mannens is conducting an NWO funded PhD project between the ETCBC and VU Faculty of Humanities, called "Verbal Valence in the Dead Sea Scrolls: Syntactic Variation and Linguistic Change".

37 Christiaan Erwich is actively researching participant tracking through his ongoing NWO funded PhD project at the ETCBC titled "Who is Who in the Psalms? A Computational Analysis of Participants and Their Networks".

tools will no doubt continue to push the ETCBC's active reflection in methodology, and hopefully lead to a better understanding of the languages it studies and the corpora that contain them.

References

Bod, R. 2013. "Who's Afraid of Patterns? The Particular versus the Universal Meaning of Humanities 3.0." *BMGN Low Countries Historical Review* 128, 171–80. https://doi.org/10.18352/bmgn-lchr.9351.

Croft, W. 2001. *Radical Construction Grammar: Syntactic Theory in Typological Perspective*. Oxford: Oxford University Press. https://doi.org/10.1093/acprof:oso/9780198299554.001.0001.

Doedens, C-J. 1994. *Text Databases. One Database Model and Several Retrieval Languages.* Language and Computers 14. Amsterdam and Atlanta, GA: Rodopi.

Dyk, J., Glanz, O. and Oosting, R. 2014. "Analyzing Valence Patterns in Biblical Hebrew: Theoretical Questions and Analytic Frameworks." *JNSL* 40 (1), 43–62.

Dyk, J., Glanz, O. and Oosting, R. 2015. "Valence Patterns in Biblical Hebrew: Classical Philology and Linguistic Patterns." *JNSL* 41 (2): 31–55.

Granger, G.G. 1992. *La Vérification*. Paris: Odile Jacob.

Haspelmath, M. 2007. "Pre-established Categories Don't Exist: Consequences for Language Description and Typology." *Linguistic Typology* 11:119–132. https://doi.org/10.1515/LINGTY.2007.011.

Kingham, Cody 2017. "Time Spans in Genesis." https://github.com/codykingham/tfNotebooks/blob/master/timeSpans/Time_Spans.ipynb.

Lazard, G. 2012. "The Case for Pure Linguistics." *Studies in Language* 36:241–259. https://doi.org/10.1075/sl.36.2.02laz.

Matthews, P. 2001. *A Short History of Structural Linguistics*. Cambridge: Cambridge University Press. https://doi.org/10.1017/CBO9780511612596.

Naaijer, M. and Roorda, D. 2016. "Syntactic Variation in Masoretic Hebrew: The Object Clause Reconsidered." *JSem* 25 (2): 961–971. https://doi.org/10.25159/1013-8471/2566.

Oosting, R. 2016. “Computer-Assisted Analysis of Old Testament Texts: The Contribution of the WIVU to Old Testament Scholarship.” In *The Present State of Old Testament Studies in the Low Countries: A Collection of Old Testament Studies Published on the Occasion of the Seventy-fifth Anniversary of the Outtestamentisch Werkgezelschap*, edited by K. Spronk, 192–209. OTS 69. Leiden: Brill.

Petersen, U. 2004. “Emdros – A Text Database Engine for Analyzed or Annotated Text.” In ACL, *COLING 2004 Geneva, 20th International Conference on Computational Linguistics, August 23rd to 27th, 2004. Volume II. Proceedings*, edited by L. Lemnitzer, D. Meurers, and E. Hinrichs, 1190–1193. Sroudsburg, PA: Association for Computational Linguistics.

Roorda, D., Kalkman, G., Naaijer, M. and Van Canenburgh, A. 2014. “LAF-Fabric: A Data Analysis Tool for Linguistic Annotation Framework with an Application to the Hebrew Bible.” *Computational Linguistics in the Netherlands* 4, 105–120.

Roorda, D. and Naaijer, M. 2016. “Parallel Passages in the MT.” https://shebanq.ancient-data.org/shebanq/static/docs/tools/ parallel/parallels.html.

Sikkel, C. 2014. “Description of Quest II Data File Format.” https://shebanq.ancient-data.org/shebanq/static/docs/qdf1.33-2014-03-26.pdf.

Talstra, E. 1986. “An Hierarchically Structured Data Base of Biblical Hebrew Texts. The Relationship of Grammar and Encoding.” In *Proceedings of the First International Colloquium. Bible and Computer: The Text*, 337–349. Champion-Slatkine: Paris-Genève.

Talstra, E. 1992a. “Text Grammar and Biblical Hebrew: The Viewpoint of Wolfgang Schneider.” *Journal of Translation and Textlinguistics* 5, 269–297.

Talstra, E. 1992b. “Text Grammar and Computer. The Balance of Interpretation and Calculation.” In *Proceedings of the Third International Colloquium. Bible and Computer: Interpretation, Hermeneutics, Expertise*, 135–149. Champion-Slatkine: Paris-Genève.

Talstra, E. 1997. “A Hierarchy of Clauses in Biblical Hebrew Narrative.” In *Narrative Syntax and the Hebrew Bible: Papers of the Tilburg Conference 1996*, edited by E. Van Wolde, 85–118. BibInt 29. Leiden: Brill.

Talstra, E. 2004. “Text Segmentation and Linguistic Levels. Preparing Data for SESB.” In *Handbuch/Instruction manual SESB (Stuttgart Elektronic Study Bible)*, edited by C. Hardmeier, E. Talstra, and B. Salzmann, 23–31. Stuttgart: Deutsche Bibelgesellschaft.

Talstra, E. and Erwich, C. Forthcoming. “The Text as Our Teacher. Participant Tracking in Psalm 64.” In *Multiple Teachers and Disciples in Biblical Texts*, edited by B. Koet and A.Van Wieringen. Forthcoming.

Talstra, E. and Van der Merwe, C.H.J. 2002. "Analysis, Retrieval and the Demand for more Data. Integrating the Results of a Formal Textlinguistic and Cognitive Based Pragmatic Approach to the Analysis of Deut 4:1–40." In *Bible and Computer. The Stellenbosch AIBI-6 Conference. Proceedings of the Association Internationale Bible et Informatique*, edited by J.A. Cook, 43–78 Leiden: Brill.

Talstra, E., Postma, F. and Vervenne, M. 1983. *Exodus: Materials in Automatic Text Processing. Part I: Morphological Syntactical and Literary Case Studies*. Instrumenta Biblica 1. Amsterdam: VU Boekhandel.

Van der Merwe, C.H.J. 1994. "Discourse Linguistics and Biblical Hebrew Grammar." In *Biblical Hebrew and Discourse Linguistics,* edited by R.D. Bergen, 13–49. Winona Lake: Eisenbrauns.

Van Peursen, W.T. 2005. *Computer-Assisted Linguistic Analysis of the Peshitta*. DANS. https://doi.org/10.17026/dans-zv9-w9d2.

Van Peursen, W.T. 2007. *Language and Interpretation in the Syriac Text of Ben Sira: A Comparative Linguistic and Literary Study*. Monographs of the Peshitta Institute Leiden 16. Leiden: Brill. https://doi.org/10.1163/ej.9789004163942.i-475.

Van Peursen, W.T. 2015. "Mathematical Rigour and Scholarly Intuition. Some Reflections on Andersen's and Forbes' Biblical Hebrew Grammar Visualized." *ANES* 52, 298–307.

Verheij, A.J.C. 1994. *Grammatica Digitalis: The Morphological Code in the "Werkgroep Informatica" Computer Text of the Hebrew Bible*. Amsterdam: VU University Press.