Caves of the ETCBC:
Exposing and Evaluating the Data Creation Process at the Eep Talstra Centre for Bible and Computer

An Internship Report

By

Cody Kingham

Mentor: Constantijn Sikkel
Supervisor: Maarten Wisse

Vrije Universiteit Amsterdam
Periods 4-5

1. Introduction

The purpose of this internship was to "expose the processes behind generating the Hebrew syntactic data" at the Eep Talstra Centre for Bible and Computer at the Vrije Universiteit Amsterdam (VU). The Eep Talstra Centre (ETCBC) is a research center at the VU which has, since 1977, developed a database of Biblical Hebrew syntax for linguistic and exegetic research. The unique focus of the ETCBC as opposed to other databases of ancient Hebrew is its emphasis on form-to-function encoding (Van Peursen 2015: 302–303). The form-to-function method emphasizes the priority of formal patterns of linguistic units in determining syntactic function. It contrasts with a functional approach, which aims to interpret patterns through a broader theoretical framework (Van der Merwe 1994: 16–17).

The emphasis on registering formal patterns in the language also means that the ETCBC places more attention on the method of text encoding than alternatives. For instance, some databases add parsing data about Hebrew words onto the surface of the word so that the individual morphemes within the word are simply evaluated by the researcher who is doing the encoding work (Van Peursen 2015: 299–300). The ETCBC, on the other hand, employs a set of files that contain both morpheme patterns and the explicit rules which govern their evaluation into a complete parsing. By analyzing the surface text against these rules and patterns, the parsing can be automatically generated and evaluated by the researcher. Not only is this method conducive for encoding texts more quickly, it aligns with, and is motivated by, the centre's concern for describing formal patterns before moving to the linguistic theory.

The fact that the ETCBC places so much weight on the data creation procedure as an important part of the process of understanding grammar means that describing those processes is

of special importance. Yet, while researchers have published several resources on the form-to-function theory of the ETCBC over the years (see especially Talstra 2004), there is as of yet no full accounting of the actual procedures used to create the data. This oversight is surprising in light of the centre's goal of emphasizing data discovery before theory. The literature itself greatly emphasizes the *theory* of form-to-function, and even describes some practical aspects of the data creation process. But by and large, the explicit details have not yet been described for a broader audience, perhaps in part since biblical scholars might find such details as tedious.

While biblical scholars are not known for their technological prowess, the lack of explicit documentation for data creation creates tension for those who might otherwise use the database for their own research. How can a researcher use data for which they have no idea whence it came? Much of the data model itself has been formed based on practice, even trial and error (a point which Talstra has embraced and commended, see Talstra 2016: 242). Such a model produces unique data features not found in other databases such as the "atom", i.e. the phrase, clause, and sentence atoms in the ETCBC database model. It is difficult to understand the purpose of these unique linguistic objects without first obtaining a decent understanding of how they came to be—through practice not theory.

In a recent (2015) workshop on tools in digital humanities, Traub and Ossenbruggen highlight another problem with the lack of data creation documentation:

> It is vitally important to make explicit how the data in the ETCBC database has been encoded. Who has done it by what methods? Especially when the same researcher draws conclusions from the database as the one who has contributed relevant parts of the encoding. That is not necessarily bad, as long as his/her method of encoding is well described and can be subject to criticism. (2015: 3)

Indeed, the ETCBC's aim to emphasize the methodological integrity of its results means that the processes used to encode the texts must fully and carefully be explained. Only then can the

methods be questioned and, in light of critical evaluation, be improved. Not only would its exposure allow for constructive criticism, it would also open the door for wider acceptance for the methods and findings.

2. Goals of the Internship

This internship aimed to provide a starting point for documenting the ETCBC data creation processes. It focused on three goals (in addition to the production of this report) as outlined in the internship plan:

1) "Document the workflows and programs that build the ETCBC data in a reference-guide format with hyperlinks and easily accessible descriptions/definitions."
2) "Submit an article for submission to the Journal of Semitics introducing and describing the activities of the ETCBC."
3) "Compare the ETCBC theory and methodology with other database approaches."

The mentor selected for the project was Constantijn Sikkel, who is the research officer at the ETCBC and who has designed and overseen most of the data creation processes over the last *27 years. Additional cooperation with the director of the ETCBC, Wido van Peursen, provided an opportunity to submit an article which details the corpus coverage, encoding methods, and future direction of the centre. Finally, an assistantship with the Computational Lexicology and Terminology Lab (VU) for a course on text-mining provided the student with an opportunity to compare methods of text-processing with those of the ETCBC.

The learning objectives for the student's own research were to obtain a better understanding of the data creation methods used by the ETCBC, as well as to practice critical evaluation of methodology. The student's thesis, for instance, focuses on using the database to identify time markers in the Hebrew Bible so that they can be compared against the verbal forms found within. The thesis is an evaluation of contemporary tense and aspect theories in the

Hebrew Bible as well as an evaluation of the so-called text linguistic approach to tense/aspect favored by Talstra, Niccacci, and other form-to-function linguists. An understanding of the data creation model is crucial for understanding the data model itself, and it will help better exploit the data for the research goals.

3.1. Results: Data Creation Documentation

After consulting with the project mentor, Constantijn Sikkel, it was determined that whatever documentation would be made on the data creation process should be easily extensible or modifiable in the likely event that pipeline would grow or change. This led to further reflection on an ideal format for the documentation itself. Traditional paper or PDF publication is limited by its static nature, and as soon as an update to the documentation is made, the physical copies or digital files become obsolete. Based on this reflection, it was decided that a digital resource would be best suited for the documentation. This would allow the work to be easily distributed online with hyperlinks to other relevant resources. The documentation could be instantly updated in light of changes, and it could be extended later to include even more details on the processes.

The student opted to compose the documentation in markdown format, a language designed for easily composing documents that can then be exported to HTML for online distribution.[1] For instance, in a normal HTML document, one needs to enclose bolded text with brackets and add instructions to the brackets to stylize the text; but in markdown, bolding a text is

[1] "Markdown is a text-to-HTML conversion tool for web writers. Markdown allows you to write using an easy-to-read, easy-to-write plain text format, then convert it to structurally valid XHTML (or HTML)." https://web.archive.org/web/20040402182332/http://daringfireball.net/projects/markdown/

as simple as enclosing it in ** double asterisks.** Similar short hand conventions in markdown allow for easily creating hyperlinks or inserting images.

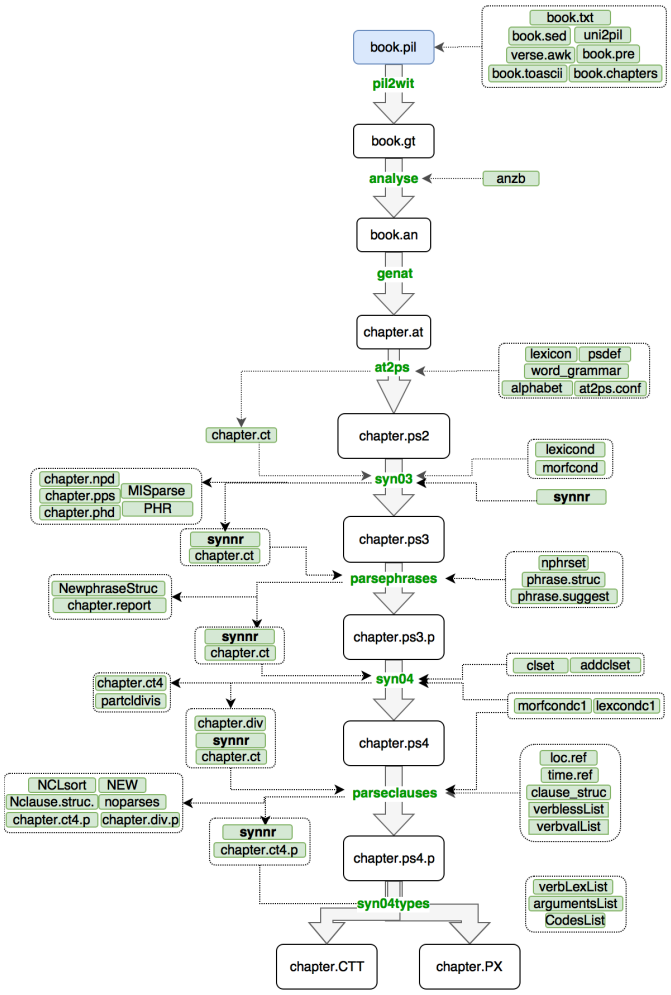
The decision to utilize markdown for the documentation had several unexpected benefits to the final product. One profit is the ease of introducing graphic media into the running text, so that the documentation could readily include illustrative diagrams for each of the analysis files described therein. Another is the option to add additional formatting options that simply would not be possible in a traditional setting. Many of the files produced in the data creation process contain more columns than can be displayed on a usual page size. But the HTML/CSS template used allowed for code/text to overflow into a scrollable box without interrupting the flow of the descriptions. Finally, the digital markdown/HTML document is easily extensible, since new sections might be added and instantly pushed to the online sphere it serves. Since it was simply impossible to document *all* of the processes of data creation, this last fact is especially relevant. The resource can later be updated with more and more refined definitions and explanations.

The final product is primarily published on the student's personal website under the link www.codykingham.com/etcbc/datacreation.html.[2] Example files and manual pages from the programs in the ETCBC central server are separately published in a Github repository at https://github.com/codykingham/ETCBC_DataCreation. The repository allows for those who have no access to the ETCBC central server to still obtain a good sense of the programs and files produced by them.

The documentation primarily focuses on 1) the pipeline processes encoded in a representative `Makefile`, which is similar to a Unix script but contains interlinking

[2] The page can easily be moved to the ETCBC's upcoming website, www.etcbc.nl, later on.

dependencies between files and the processes that produce them; and 2) the analysis files yielded by the pipeline processes. The latter goal required extra attention since the file formats are crucial to understanding the process as a whole, and also because the documentation is most lacking in this area.[3] The diagram below is used at the beginning of the documentation to fulfill the first goal:



White boxes represent analysis files produced by the programs and processes. Green represents part of a process in the pipeline; green words between broad arrows represent the programs

[3] The manual pages in the ETCBC server offer some help, but they are still not sufficient for describing the files for the uninitiated. There are also several gaps in the manual pages.

themselves and the green boxes stand for the auxiliary files used by them. Some files are omitted from the diagram since they play only a minor role in the overall process. Files on the right of the pipeline contain grammatical or structural data needed for the programs to analyze the data. Files on the left are those which the programs produce as a result of the analysis.

To fulfill the second goal, documenting the analysis files created by the programs, the presentation describes the files in the logical progression of their creation. It organizes them into one of six sections: 1) raw text analysis files, 2) morphological analysis files, 3) word level analysis files, 4) phrase level analysis files, 5) clause level analysis files, and 6) text level analysis files. The documentation offers a profile of each file format, with an introduction, location of documentation in the server,[4] the source of the file (i.e. its program of origin), a sample of the file in a scrollable box, and an illustrative diagram of the file with extensive explanation on how to understand its various parts. Sample screen shots of a file profile are attached in the appendix. Other, larger examples are of course available in the document.

3.2. Results: Journal of Semitics Article

The opportunity to submit a joint-article on the ETCBC was offered to the student by Wido van Peursen, who was invited to submit an article on a presentation he had given at the IOSOT conference in Stellenbosch, South Africa. The student was responsible for composing the material based on the presentation given by Van Peursen. The article served the goal of providing the student with an opportunity to read literature and describe the theoretical side of the data creation processes. In this way, the article also functions as a companion product to the data

[4] Again, it is limited. The documents, i.e. "man" pages, are uploaded to the project Github repository.

creation documentation since it provides a soft introduction to the methods and resources at the centre. Writing the article also imparted to the student the professional experience of publishing. A copy of the article has been attached to the appendix of this report. The abstract is presented below:

> 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 in order to address their research questions.

The article has been submitted and is currently undergoing the review process.

3.3. Results: Comparison of the ETCBC with Other Approaches

The final goal of offering comparison between the ETCBC database methods with other approaches was primarily achieved through the documentation already described, the article, and also through the student's assistantship with a bachelor's course on text-mining. The documentation allows one to fully see the entire pipeline process from start to finish, and thus to easily compare with other methods that rely less on rules-based processing, and more on researcher intuition (such as the Forbes database, see Van Peursen 2015). Researching literature, and explaining in the article the distinctive features of the ETCBC allowed for the additional comparison. Finally, the course on text mining required the student to be prepared to field student questions on the text-mining topic and to become more thoroughly familiar with the data pipelines used by the Computational Lexicology and Terminology Lab. Those pipelines, which also utilize Unix tools common to the ETCBC, revealed to the student that the ETCBC method is

surprisingly modern, being comparable to some techniques being used in artificial intelligence research, despite the processes being now several decades old.

4. Evaluation of the Results

In view of the internship goals outlined in the internship plan and their implementations as described above, the internship can be deemed a success. The data creation documentation provides a full overview with illustrations. The documentation has been prepared not only for the research needs, but also in hopes that it can be a useful tool in the future for new students and researchers who will actually use the ETCBC tools to encode new texts. The materials produced by the internship meets these needs.

In terms of documenting the specific behavior of the programs themselves, i.e. the code that actually drives their output, more work is needed. Importantly, the documentation produced during this internship builds a platform upon which the documentation can expand. It also provides a place, via the newly created Github repository, wherein the programs' code itself might eventually be annotated and uploaded so that the technically-minded can evaluate it and perhaps even improve upon it. The growing online community of ETCBC database users (through platforms like SHEBANQ and Text-Fabric) provides a resource in itself where the research goals of the ETCBC might be enhanced. Specifically, funding for projects that would help better document, clean up, or optimize the code is especially limited or non-existent. Those needs can instead be met by competent enthusiasts who give their time freely to improve the database.

The article submitted to the Journal of Semitics has also met the goal of providing the student with the opportunity to study additional literature on the ETCBC and gain experience in

publishing academic work. In addition to the internship itself, the article provides the student with a positive mark on their curriculum vitae.

Finally, the comparison with other database models has also been successful, but perhaps not in the precise format outlined in the internship plan. This aspect was the smallest portion of the internship, and it was deemed more important to focus on producing quality, in-depth documentation of the data creation programs. That documentation, combined with the experience gained through the text mining assistantship, provided both tangible and qualitative reflection on how the ETCBC method compares with others.

5. Final Reflections on the Internship as a Whole

The internship plan, in accordance with the guidelines for the internship course (THE_G_INTERN_2016_141), called for fifteen weeks of work beginning on 20 February and extending until 2 June. The workload requirements for each of the goals outlined above, including this present report, total to 12 EC credits.

The student has met goals as outlined in the internship plan, and the workload as expressed in the outline are accurate to how the internship was carried out. Since the ETCBC article took more time to produce than anticipated, the present results are a few weeks delayed. Likewise, studying the data creation pipeline at the ETCBC took place throughout the fifteen weeks as a whole, with some extra initiative exerted towards the last 3 weeks of the project.

In terms of the student's own challenges, weaknesses, and hardships to overcome throughout the project, the student was able to grow in some new areas. One of those areas is in the discipline of note taking. Even if the material is written out on a computer screen, it helped the student to physically write out, step-by-step, the processes before assimilating them into a full

description. Secondly, the student had to overcome some initial lack of competency in navigating the Unix environment of the ETCBC server. After the project, though, the student is now well-initiated, and feels prepared to apply the new skills for his further research.

Works Cited

Andersen, Francis & Forbes, Dean 2012. *Biblical Hebrew Grammar Visualized*. Winona Lake: Eisenbrauns.

Merwe, Christo H J van der 1994. Discourse Linguistics and Biblical Hebrew Grammar, in Bergen 1994:13–49.

Ossenbruggen, Jacob & Myriam, Traub 2015. 'Workshop on Tool Criticism in the Digital Humanities' *CWI Techreport* 2015: 1-7.

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

Talstra, Eep 2016. Data, Knowledge and Tradition: Biblica Scholarship and the Humanities 2.0: Exodus 19 as a Laboratory Text, in Spronk 2016: 228–247.

— 2004. Text segmentation and linguistic levels. Preparing data for SESB, in Hardmeier, C, Talstra, Eep & Salzmann, B, *Handbuch/Instruction manual SESB* (*Stuttgart Elektronic StudyBible*). Stuttgart: Deutsche Bibelgesellschaft, 23–31.

Appendix 1 – PIL Analysis File

2.1 Raw Text Analysis

.pil

The .pil ("Peshitta Institute of Leiden") file is the first step in entering an ancient text into a computer-readable format. It contains the chapter and verse boundaries, transcription of the document text, and notations of variants/reconstructions found in other witnesses. The file also supports the notation of lacunae and fragmentary readings in a text.

The .pil file is generated by processing a plain text document, perhaps with the original utf8 characters copied and pasted from an original source (`book.txt` in the diagram above).

Source

plain text file, perhaps in unicode, processed into .pil format with `sed`, `awk`, `cat`, or other UNIX text processing commands

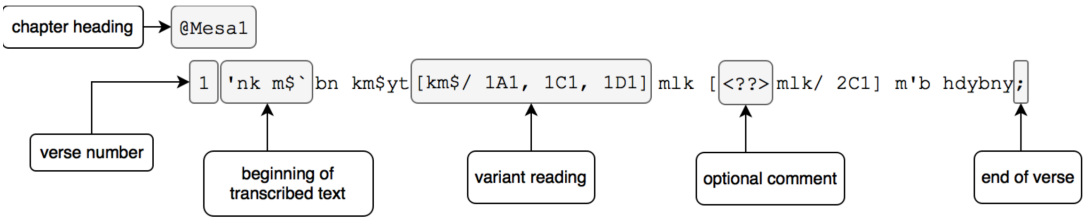
Documentation

Format of a PIL Running Text File or `/projects/calap/doc/format/format.pdf`

Sample

```
@Mesa1 1 'nk m$` bn km$yt  [km$/ 1A1, 1C1, 1D1] mlk [<??>mlk/
2C1] m'b hdybny; 2 'by mlk `l m'b $l$n $t w'nk mlkty 'Hr 'by; 3 w'`$ hbmt z't
lkm$ bqrHh; 4 bmt y$` {bm<???>$`/ 1A1, 1D1} {b<????>$`/ 1C1} ky
h$`ny mkl hmlkn wky hr'ny bkl $n'y; 5 `mry mlk y$r'l wy`nw 't m'b ymn rbn ky
y'np km$ b'rSh; ...
```

Parts



lines 1-2 of Mesa.pil

A more detailed description for each of these elements and some others, as well as the formatting rules for a .pil file, is in the documentation, Format of a PIL .