

Fieldworks Language Explorer (FLEX) 3.0

from SIL International

Reviewed by CHRIS ROGERS, *University of Utah*

1. OVERVIEW. One of the most important goals in language documentation is to produce reference materials as well as to create long-lasting (i.e., archivable) materials for the languages. Fieldworks Language Explorer (FLEX) 3.0 is software for organizing and analyzing linguistic data and is produced for free download by SIL International (SIL)¹. This review has three goals: to show how the current version of FLEX addresses problems in earlier versions of the program, to highlight some of the more useful features of FLEX 3.0, and to indicate how it compares to other database programs.

Like earlier versions of the program and its predecessors (Shoebox/Toolbox), FLEX consists of components designed to organize information for a given language, such as its lexicon, grammar, and texts. The broad categories of language organization form separate components of the program: LEXICON, TEXTS & WORDS, and GRAMMAR. Each of these component parts consists of a number of options and data fields, most of which can be customized to the needs of a given language project. When taken together the program's components can be used to produce outputs useful in language conservation, such as: wordlists and dictionaries, collections of texts and examples sentences, and interlinear morphological analyses; though these outputs are of varying sophistication (see below). This review is concerned with the new features in FLEX (3.0) and with the usefulness of this program to both seasoned field workers and those looking for an ideal program for their first field experiences.

By way of general summary, I have found this program to be useful for its described purposes and an important asset in any language documentation effort. The updated version provides solutions to many of the difficulties present in past editions. However, there are some aspects of the program that limit its use and render it less than suitable for some language projects.

2. ISSUES WITH PREVIOUS VERSIONS OF FLEX (UPDATES AND IMPROVEMENTS).

An earlier version of FLEX was reviewed in *LD&C* Vol. 1, Issue 1 (Butler and van Volkingburg 2007), and since then there have been few general changes in the organization and purpose of the program. However, a number of new features have been added and past features updated to satisfy the needs of individual language projects. It is worth mentioning how these updates compare to the previous edition. Butler and van Volkingburg (2007), indicate that there are at least thirteen issues with FLEX that make its use problematic in language documentation projects. Some of these have been addressed in the current version of the program while others remain unresolved. My review focuses on the issues highlighted

¹ <http://www.sil.org/computing/fieldworks/FLEX/index.htm>

in the 2007 review that have since been updated or that have received some attention in the current version of FLE_x.

One issue that has been addressed but not completely resolved is the output of what FLE_x terms “reversal indexes”. Butler and van Volkingburg (2007:102) write that “one major flaw in FLE_x...[is] the lack of a full-featured gloss to vernacular dictionary output.” This requirement would allow FLE_x to output a dictionary with terms in the contact, national, or analysis language as the headwords in a dictionary with the gloss being a lexical item in the vernacular language being analyzed, documented, or described. While FLE_x was always able to produce such outputs, the amount of information in a gloss-to-vernacular dictionary was limited because parts of speech, example sentences, cross-references, and other relevant information was not displayed. Butler and van Volkingburg are correct in that this was a major problem with the design of FLE_x. The current version seeks to remedy this issue.

In the updated version it is possible to include the linguistic information associated with vernacular language with entries in the gloss, or analysis, language. For example, “reversal indexes” can be organized according to the headword in the glossing language and can furthermore include data on the variants of the vernacular form, grammatical categories (parts of speech), subentries of the headword, example sentences, cross-references in the vernacular language (though not of the gloss language), and semantic information about the entry. This means that reversal indexes are treated very similarly to the vernacular-to-gloss dictionary FLE_x can create.

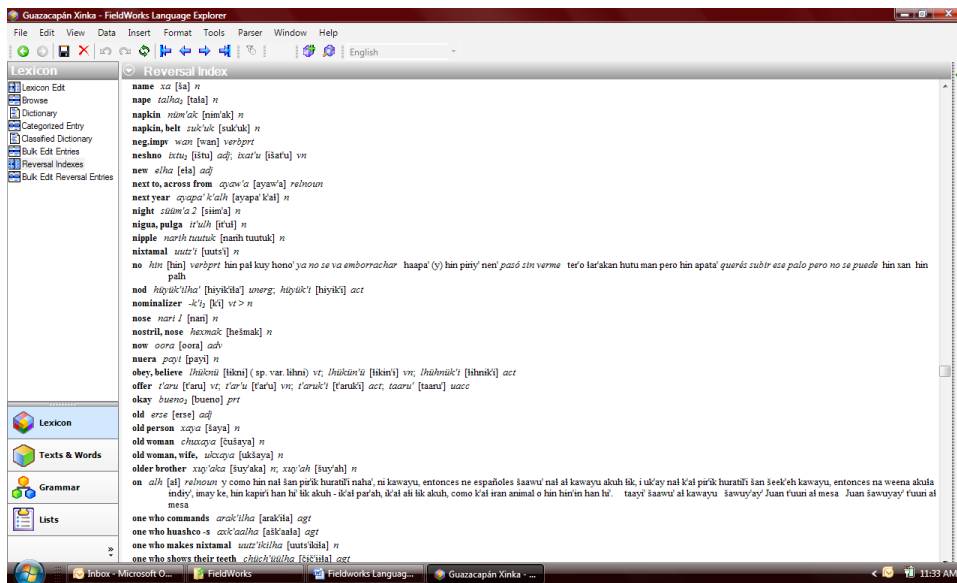


FIGURE 1. Reversal Index view

Another issue is the lack of output options for both configured dictionaries and analyzed texts. In Butler and van Volkingburg’s review they comment that they would like a .txt format output option (2007:103). The newer version of FLE_x still only allows exports

into .xml or .db formats. However, beneficial to some users is the new integration with Microsoft Word 2007. Specifically, an .xml file can be exported from FLEx and read by Word 2007. The exported file largely preserves the formatting of text data used in FLEx. One drawback, though, is that modifications to the export are difficult, requiring knowledge of XML style sheet creation. Many language projects do not have participants with this type of knowledge, so exporting the data is not customizable and is limited to users who have the required additional programs (i.e., Word) to produce decent dictionaries. That said, I have created three vernacular-to-gloss dictionaries with custom formatting and split columns in Word 2007. Though the output has been satisfactory, and this update was a step in the right direction, the proprietary nature of MS Word limits the overall functionality of FLEx.

Sometimes FLEx is unbearably slow. This has been an issue since the first release of the program and continues to be so with no resolution in sight. This slowness is directly proportional to the size of the database; so, for example, very small databases have faster response times than very large ones (i.e., databases containing tens of thousands of lexical entries). This is especially true when the user switches among the main components of the program (LEXICON, WORDS & TEXTS, GRAMMAR, and LISTS) or when the view is toggled within a single component (e.g., INTERLINEAR TEXTS and CONCORDANCES in the TEXTS & GRAMMAR component, or between BASELINE VIEW and ANALYSIS VIEW in the INTERLINEAR TEXT option).

Delayed program response time is a problem especially when working with texts. Unlike Toolbox or other database programs (e.g., FileMaker Pro, see below), corrections or modifications of a text or example sentence must be done in the BASELINE view which does not show the analysis window or the lexical information stored in the database. Often while working in the ANALYZE view in the TEXT & WORDS component, I have needed to make changes directly to the text I am working on (e.g., punctuation or spelling errors) and have needed to change the view back to the BASELINE. This process is slow and time-consuming. However, even more difficult is the simultaneous consultation of lexical entries and texts/examples sentences. In Toolbox this was straightforward; two windows were opened, one containing the lexical information and one containing the texts/sentences. In FLEx the user must switch between the LEXICON and TEXT & WORDS components, which cannot be viewed simultaneously.

An improvement, on the other hand, from past versions of the program is the search functionality. It is now possible to search a text using regular expressions. This means that strings can be specified for many different environments with a word or text. However, most beneficial is the addition allowing searches to be made of texts, translations, notes, glosses, and lexical entries (see Butler and van Volkinburg 2007:104). In addition, though the lack of this option posed only a minor issue with the program, example sentences can now be displayed and edited for bulk editing (see Butler and van Volkinburg 2007:101). All other issues highlighted in the last review of FLEx have been left unaddressed.

3. OTHER UPDATES AND IMPROVEMENTS. Besides the updates to the past issues with the program, FLEx 3.0 provides other additions and upgrades to each of the component parts of the program. Some of these updates do not apply to the projects I have worked on and so are not evaluated at any length there, though a complete list of new features is available on the SIL website.

In the LEXICON view there are improvements in the organization and representation of the lexical data as well as in the analysis options of this data. Specifically, the improvements in the organization of variant forms and complex forms are more intuitive and functional. For example, it is now possible to choose whether a variant is a dialectal variant, a free variant, an irregularly inflected variant, a spelling variant, or some combination of any of these. It is also advantageous that these lists can be added to or customized for the specific needs of the project. This is an improvement as data management is now less confined to a pre-set mold.

Another useful and functional feature is that in the dictionary view, the variants are listed with the “regular” entries and are appropriately cross-referenced. This feature is especially useful in my current project as there are a large number of variant forms in the data.

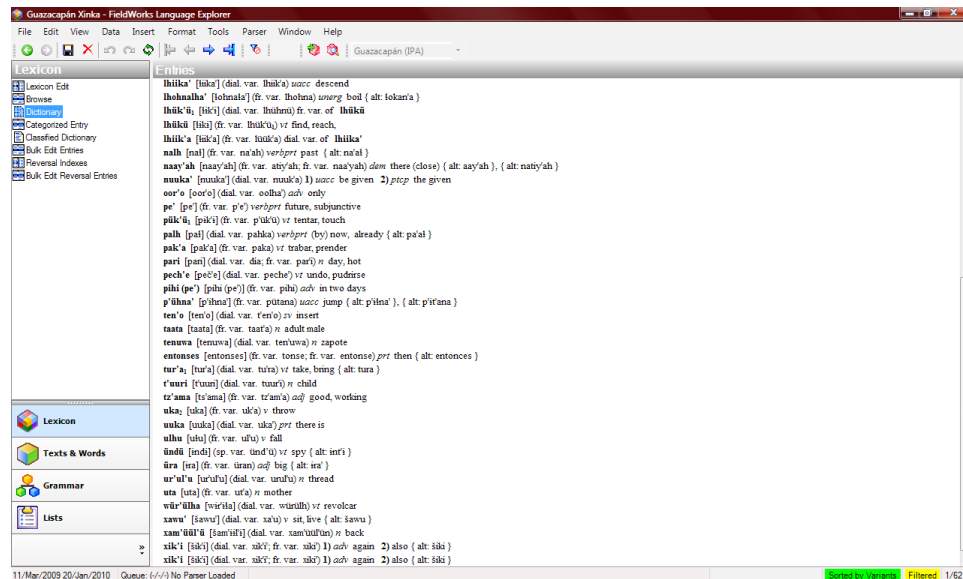


FIGURE 2. Lexicon Dictionary view

Other useful updates to the LEXICON component include the treatment of complex forms and the configuration of reversal indexes or wordlists. It is now possible to indicate that a lexical form is a complex form (e.g., a compound word) while creating a new entry. This reduces the amount of time and effort needed to review and appropriately organize the data.

In the TEXT & WORDS component, improvements include the ability to import interlinearly glossed texts (IGT) from other programs and a syntactic tagger, among other things (for a full list see the SIL website). Previous editions did not support the importing of IGT and this was a major drawback for those wanting to migrate from other programs (e.g., Shoebox/Toolbox) to FLEx. In order to accomplish the import there are a number of steps that ensure the original text is formatted correctly, and these steps are clearly outlined in the technical documents on the SIL website. However, I have not had any success in migrating

my texts from Toolbox (or any other program) into FLEx. The process is time consuming and clunky. So, while this is a welcome new addition, it is has not been developed sufficiently to provide satisfactory results. These problems are especially true for texts stored in programs that are not SIL-endorsed (e.g., MS Excel, FileMaker Pro, or MS Access).

Another very exciting feature is the newly implemented syntactic tagger component. This is found in the TEXT & WORDS view and is useful in indicating linguistically relevant information at the sentence level. With this tagger the constituents of a sentence can be labeled using one of four general categories: grammatical relations-function (e.g., A, P, S, Subject, Direct Object, Indirect Object, Ergative, Absolute, and Oblique), syntactic descriptive (e.g., phrase type), semantic roles (e.g., actor, undergoer), or syntactic roles (e.g., core syntactic role versus peripheral syntactic role). However, no constituent can be labeled using more than one of these general categories. For example, a fully inflected verb cannot be labeled a verb phrase using the syntactic descriptive labels and simultaneously be labeled as core to the sentence according to the syntactic roles labels. Fortunately, though, these lists can be customized for the needs of a specific project through the LISTS view in the FLEx database.

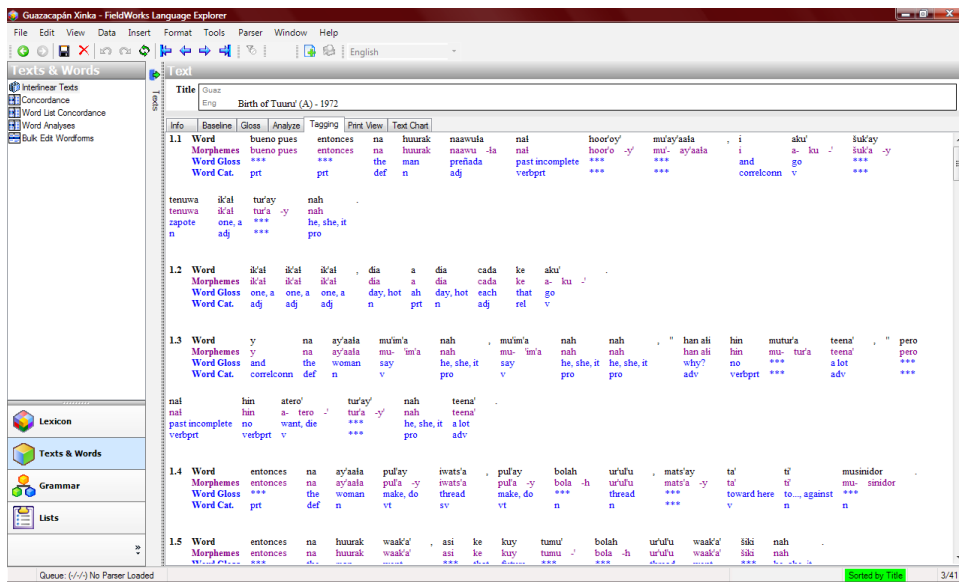


FIGURE 3. Text & Words Syntactic Tagger View

There is also a new option of copying interlinear sentences from FLEx in the text & words view and pasting them into a word processor in tab-delimited format. I have very successfully used both of these features.

4. ISSUES AND DRAWBACKS. I have used FLEx for the past year and a half, and I have recently upgraded to version 3.0. Using this, I have created a number of small databases of the four endangered or extinct Xinkan languages of southeastern Guatemala. None of these

FLEx projects exceed 4,000 lexical entries. Previously, I used Toolbox and FileMaker Pro to create the databases of these languages.

I have found that, in general, I like FLEx better than Toolbox because the organization of the different components is more intuitive and user-friendly. Most of the many components, lists, and menus in FLEx are customizable, making it a better language-specific databasing tool than Toolbox. Furthermore, many of the needed fields are preset in FLEx—advantageous when compared to database tools such as FileMaker Pro in which an entire database must be created from scratch. However, FLEx is limited in its use because it does not support multi-lingual databases, nor does it provide the complete set of applications sometimes necessary to a fieldwork project.

The goal of FLEx is to “attempt to provide the best functionality of [the many software tools use to gather and publish data] in one application” (SIL website). However, while FLEx does provide good support for dictionary organization and creation as well as a means for the interlinear glossing of texts and example sentences, this is far from all that is needed in language documentation and description. Documentation projects still need to look elsewhere for software to annotate and archive digital recordings (video and voice), analyze phonetic characteristics of a language, and organize advanced syntactic constituents. This software tool is only one of many different software packages that may be needed.

Of course, the usefulness of FLEx cannot be judged by what it is not designed to do. In fact, it is an ideal program to organize lexical information. FLEx can also export decent dictionaries if the user has the ability to interact with the XML format (e.g., by using Microsoft Word 2007, see above). But the goals quoted above in relation to language documentation and description are misleading. The outputs of this program are not yet useful to many language projects, and they are definitely not the only ones needed for language documentation and description.

Another issue specific to the Xinkan projects mentioned above is the absence of multilingual dictionary capabilities. One goal of the Xinkan database is to create a comparative dictionary of the four Xinkan languages with cognates cross-referenced and language specific idiosyncrasies independent of the other languages in the project. While each language can be used in an independent database, there is no functional way to combine the information for all four Xinkan languages. The FLEx website provides some technical documentation, one of which (“ICU and writing systems”) is specifically devoted to this issue. This document openly acknowledges that the lack of multilingual capabilities might be problematic for some projects and provides some possible solutions. The solutions are based on the writing system organization in FLEx. Specifically, the website suggests that independent languages might be used as different writing systems though this requires a headword related to but independent of each language. Furthermore, these writing systems cannot be used independently as the headwords can (i.e., they cannot be specified for variants, they are not available for the highest level of syntactic analysis, nor do they appear as independent entries in the dictionary view).

After trying all of the suggestions in the technical documentation, I found that none of the solutions really support multilingual capabilities. I recommend that this issue be addressed in further developments of the program. This might be accomplished, for example, through a language drop-down menu of headwords that must be indicated for each new

lexical entry with the language choice indicated in the dictionary view. This feature might be turned on or off depending on the needs of the project.

5. CONCLUSIONS. A quality documentation of a language can be made easier with the correct tools. I have found FLEx to provide useful tools for language documentation, and I encourage others to investigate its functionality. While it may not fit every language project perfectly, it is useful as a data organization tool. [Author's note - Subsequent to the completion of this review FieldWorks 6.0.3 was released in April 2010 (see http://fieldworks.sil.org/stable_download.htm).]

Primary function:	“Field linguists use a variety of software tools to gather and publish their data. In FieldWorks Language Explorer, we are attempting to provide the best functionality of those tools in one application. Beyond that, we are seeking to bring a new level of power and support to both traditional approaches as well as promising, new techniques. Finally, as part of the FieldWorks suite of applications, Language Explorer brings the power and efficiency of integrating with other language program data, such as translation and anthropology research.” (SIL website)
Pros:	Organization and archiving capabilities; morphological analysis; and customization. Relatively easy to learn; more functionality with variants forms, complex forms, and reversal indexes. Interlinearly glossed text import functions and syntactic tagger.
Cons:	Switching between glossing views and lexical lists is slow. Multi-language dictionary creation appears to be impossible. No Mac version.
Platforms:	PC
Open Source:	Yes. Available from http://fieldworks.sil.org/License/SIL_open_source_license_frame.htm
Proprietary:	Shareware
Available from:	SIL International (http://www.sil.org/computing/fieldworks/FLEx/)
Cost:	Free
Reviewed Version:	3.0.0; Fieldworks 6.0
Application Size:	293 MB
Documentation:	Introductory tutorials, class handouts, and technical documentation. (http://fieldworks.sil.org/techsupportdocs.htm)

Chris Rogers
chris.rogers@utah.edu