

Export options in Flex

Ken Zook

January 12, 2010

Contents

| | | |
|------|--|----|
| 1 | Introduction..... | 2 |
| 2 | FieldWorks XML..... | 2 |
| 3 | Dictionary | 2 |
| 3.1 | Configured Dictionary XML | 2 |
| 3.2 | Configured Dictionary SFM | 3 |
| 3.3 | Configured Dictionary Simple XML for Index Cards..... | 3 |
| 3.4 | Configured XHTML Dictionary | 5 |
| 3.5 | Full Lexicon LIFT XML..... | 6 |
| 3.6 | Full Lexicon (root/stem-based) SFM..... | 8 |
| 4 | Lists..... | 9 |
| 4.1 | Lists SFM..... | 9 |
| 4.2 | Lists Tab-delimited | 10 |
| 4.3 | Lists XML..... | 11 |
| 5 | Reversal Indexes | 12 |
| 5.1 | Reversal Index XML..... | 12 |
| 5.2 | Reversal Index SFM | 13 |
| 5.3 | Reversal Index Detail SFM..... | 13 |
| 6 | Wordforms | 14 |
| 6.1 | Wordforms SFM..... | 14 |
| 6.2 | Wordforms Tab-delimited | 14 |
| 6.3 | Wordforms XML | 14 |
| 7 | Interlinear text..... | 15 |
| 7.1 | HTML | 15 |
| 7.2 | Interlinear XML | 16 |
| 7.3 | Microsoft Word 2007 XML..... | 17 |
| 7.4 | Microsoft Word XML..... | 18 |
| 7.5 | Open Office Writer ODT | 19 |
| 7.6 | Pretty HTML..... | 20 |
| 7.7 | Verifiable generic XML..... | 21 |
| 7.8 | XLingPaper as examples, align morphemes XML | 23 |
| 7.9 | XLingPaper as examples, align words XML..... | 23 |
| 7.10 | XLingPaper, align morphemes XML | 23 |
| 7.11 | XLingPaper, align words XML | 24 |
| 7.12 | XLingPaper, as a single example, align words XML | 25 |
| 8 | Grammar sketch..... | 25 |
| 9 | Phonology Assistant..... | 25 |
| 10 | Lexique Pro..... | 26 |
| 11 | Python | 26 |

1 Introduction

Flex provides a number of ways to export all or part of your data. In addition to the built in options, other options to export any part of the FieldWorks data include the following:

- Data Notebook, Translation Editor, and List Editor--some built-in export options
- Custom FXT exports (see FXT Export options.doc)
- SQL queries (see FieldWorks database model.doc)
- Python or some other programming language that can access the database directly or through FDO (See Python database access.doc)

Note this document needs to be updated to include current export options which differ somewhat from these option.

FieldWorks XML

You can dump the entire FieldWorks project to a FieldWorks XML file. This is a single file that includes the lexicon, wordform inventory, interlinear texts, scripture, data notebook, and all lists. You can include this in a FieldWorks backup by checking "Include a human readable (XML) backup". Also see the dump and load commands for the DB program in Databases and backups.doc for information on dumping and loading, and FieldWorks XML model.doc for details on the XML format.

2 Dictionary

In the Lexicon area you can choose File...Export to bring up an Export dialog with various options. Options that do not apply to the current view are grayed out. The options listed in this dialog come from the c:\Program Files\SIL\FieldWorks\Language Explorer\Export Templates directory. It is possible to add new export options by adding FXT control files to this directory. See FXT Export options.doc for details.

2.1 Configured Dictionary XML

This option is only available when you are in the Dictionary view. It uses the current sort, filters, and view as defined in Tools...Configure Dictionary to determine what to export. The output is an XML file. Here is a sample of the format dumped to this file:

```
<?xml version="1.0" encoding="utf-8"?>
<ExportedDictionary>
<LexEntry id="hvo6232">
  <LexEntry_HeadWord><MoForm><MoForm_Form>
    <AStr ws="xkal">
      <Run ws="xkal">himbakosa</Run>
    </AStr>
  </MoForm_Form></MoForm></LexEntry_HeadWord>
  <LexEntry_Senses>
    <LexSense number="1" id="hvo6234">
      <MoStemMsa_PartOfSpeech>
        <Link><Alt ws="en" abbr="bitrans" /></Link>
      </MoStemMsa_PartOfSpeech>
      <LexSense_Gloss>
        <AStr ws="en">
```

```

        <Run ws="en">to.sell</Run>
    </AStr>
    </LexSense_Gloss>
</LexSense>
</LexEntry_Senses>
</LexEntry>
<LexEntry id="hvo6166">

```

2.2 Configured Dictionary SFM

This option is only available when you are in the Dictionary view. It uses the current sort, filters, and view as defined in Tools...Configure Dictionary to determine what to export. This is an example of the output:

```

\lx ztestmain
\va ztestminor
\ve first person plural
\va variant
\lf cf
\lv ztestsub
\sn 1
\ps N
\den This is the English definition
\xxkal Vernacular example
\xen English translation
\xxkal Second vernacular example
\xen Second English translation
\een This is English encyclopedic information.
\rsten These are English restrictions.
\lf pt
\lv pal
\lv pus1

```

This file is similar to MDF, but has a number of differences because FieldWorks knows nothing about national and regional languages. Instead, there are any number of vernacular and any number of analysis writing systems. Thus, for definitions Flex uses \d plus the writing system code (e.g., \den English, \des Spanish). Any field that is a MultiString or MultiUnicode is dumped in this way. The exported file is UTF-8 data with NFC normalization.

2.3 Configured Dictionary Simple XML for Index Cards

This option is only available when you are in the Dictionary view. It uses the current sort, filters, and view as defined in Tools...Configure Dictionary to determine what to export. It produces an XML file to produce Mara index cards. This is an example of the output:

```

<?xml version="1.0" encoding="utf-8"?>
<?xml-stylesheet type="text/xsl" href="C:\Program Files\SIL\FieldWorks\Language
Explorer\Export Templates\IndexCards.xslt"?>
<!--Card display of FW configured view-->

```

```

<flex-configured-lexicon>
<headword name="ztestmain" ws="xkal">
  <etymology>
    <form ws="xkal">etymology</form>
    <gloss ws="en">English gloss</gloss>
    <comment ws="en">English comment</comment>
  </etymology>
  <lex-function-abbr ws="en"> cf </lex-function-abbr>
  <lex-ref ws="xkal"> ztestsub </lex-ref>
  <sense num="1">
    <part-of-speech ws="en">N</part-of-speech>
    <definition ws="en"> This is the English definition </definition>
    <definition ws="xkal"> Vernacular example </definition>
    <translation ws="en"> English translation </translation>
    <definition ws="xkal"> Second vernacular example </definition>
    <translation ws="en"> Second English translation </translation>
    <enc-info ws="en"> This is English encyclopedic information. </enc-info>
    <restrictions ws="en"> These are English restrictions. </restrictions>
    <lex-function-abbr ws="en"> cal </lex-function-abbr>
    <lex-ref ws="xkal"> pal </lex-ref>
    <lex-ref ws="xkal"> ztestmain 1 </lex-ref>
    <lex-ref ws="xkal"> yalo </lex-ref>
    <lex-function-abbr ws="en"> ant </lex-function-abbr>
    <lex-ref ws="xkal"> nadra </lex-ref>
    <lex-function-abbr ws="en"> syn </lex-function-abbr>
    <lex-ref ws="xkal"> yalo </lex-ref>
    <lex-ref ws="xkal"> palti </lex-ref>
    <lex-function-abbr ws="en"> spec </lex-function-abbr>
    <lex-ref ws="xkal"> ponesa </lex-ref>
    <lex-ref ws="xkal"> pal </lex-ref>
    <lex-function-abbr ws="en"> pt </lex-function-abbr>
    <lex-ref ws="xkal"> pal </lex-ref>
    <lex-ref ws="xkal"> pus1 </lex-ref>
    <sci-name ws=""> Scientific name </sci-name>
    <sem-domain-abbr ws="en"> 1 </sem-domain-abbr>
    <sem-domain ws="en"> Universe, creation </sem-domain>
    <sem-domain-abbr ws="en"> 7 </sem-domain-abbr>
    <sem-domain ws="en"> Physical actions </sem-domain>
    <domain ws="en"> education</domain>
    <domain ws="en"> graphology</domain>
    <usage-abbr ws="en"> hon </usage-abbr>
    <usage-abbr ws="en"> oldfash </usage-abbr>
  </sense num="1.1">
    <part-of-speech ws="en">conc</part-of-speech>
    <definition ws="en"> English subsense definition for 1.1. </definition>
  </sense>

```

```

<sense num="1.2">
  <definition ws="en"> English subsense definition for 1.2. </definition>
</sense>
</sense>
<sense num="2">
  <definition ws="en"> This is the English definition for 2. </definition>
</sense>
</headword>
</flex-configured-lexicon>

```

2.4 Configured XHTML Dictionary

This option is only available when you are in the Dictionary view. It uses the current sort, filters, and view as defined in Tools...Configure Dictionary to determine what to export. This is being developed as a path for publishing a dictionary. The dictionary information and a css style sheet are exported. One way to view the output of this is to load the xhtml and css files into the Prince program (free download from www.princexml.com). It will convert this to a PDF file ready for printing. This document needs to be updated to include the Pathway export option which is now the primary export option. See <https://software.sil.org/pathway/>.

An example of the output:

```

<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="utf-8" lang="utf-8">
<!--
There are no spaces or newlines between <span> elements in this file because
whitespace is significant. We don't want extraneous spaces appearing in the
display/printout!
-->
<head>
<title /><link rel="stylesheet" href="x.css" type="text/css" /></head>
<body class="dicBody">
<div class="letHead">
<div class="letter">B b</div>
</div>
<div class="letData">
<div class="entry" id="hvo6166">
<span class="headword" lang="x-kal">-bi</span><span class="MoForm-
publishStemLexemeForm-FormPub" lang="x-kal"><MoMorphType_Prefix><AUni
ws="en">-</AUni></MoMorphType_Prefix>bi<MoMorphType_Postfix><AUni
ws="en" /></MoMorphType_Postfix></span><span class="senses"><span
class="sense" id="hvo6169"><span class="grammatical-info"><span
class="partofspeech" lang="en">sta</span><span class="slots"><span class="slot-
name" lang="en">Tense</span></span></span><span class="definition"
lang="en">Past</span><span class="semantic-domains"><span

```

```

class="xitem"><span class="semantic-domain-abbr" lang="en">6.5.1.1</span><span
class="semantic-domain-name" lang="en">House</span></span><span
class="xitem"><span class="semantic-domain-abbr" lang="en">9</span><span
class="semantic-domain-name" lang="en">Grammar</span></span><span
class="xitem"><span class="semantic-domain-abbr" lang="en">9.1</span><span
class="semantic-domain-name" lang="en">General words</span></span><span
class="xitem"><span class="semantic-domain-abbr" lang="en">9.6.2.5</span><span
class="semantic-domain-name"
lang="en">Cause</span></span></span></span></span>
</div>
</div>
</body>
</html>

```

This leaves ExportStage%.txt files in the output directory that should be deleted when done.

2.5 Full Lexicon LIFT XML

This option exports the lexical database into an XML file for import into the WeSay dictionary collection program or Lexique Pro. Here is an example of the output:

```

<?xml version="1.0" encoding="UTF-8"?>
<lexicon>
<entry id='1a705846-a814-4289-8594-4b874faca6cc' dateCreated='2004-10-18'
dateModified='2004-10-18'>
<lexicalForm ws="Kal">underlying form</lexicalForm>
<sense>
<pos ws="Eng">N</pos>
<gloss ws="Eng">English gloss</gloss>
<definition ws="Eng">This is the English definition</definition>
<exampleGroup>
<example ws="Kal">Vernacular example</example>
<translation ws="Eng">English translation</translation>
</exampleGroup>
<exampleGroup>
<example ws="Kal">Second vernacular example</example>
<translation ws="Eng">Second English translation</translation>
</exampleGroup>
<nt ws="Eng">These are English general notes.</nt>
<semanticDomainAbbr ws="Eng">1</semanticDomainAbbr>
<semanticDomainAbbr ws="Eng">7</semanticDomainAbbr>
</sense>
<sense>
<gloss ws="Eng">English gloss2</gloss>
<definition ws="Eng">This is the English definition for 2.</definition>
</sense>
</entry>

```

Complex entries and variant entries have information linking them to their main entries. Here is a main entry with allomorph 'stem allomorph' Note that in LIFT the <variant> element is used for allomorphs, not variant entries. Note, there is nothing in the main entry that indicates it has subentries or variants. This can only be detected by checking the _component-lexeme relations that point to the current entry.

```
<entry dateCreated="2004-10-19T02:42:02Z" dateModified="2007-01-17T17:23:06Z" guid="1a705846-a814-4289-8594-4b874faca6cc" id="main_1a705846-a814-4289-8594-4b874faca6cc">
  <lexical-unit>
    <form lang="x-kal"><text>main</text></form>
  </lexical-unit>
  <variant>
    <form lang="x-kal"><text>stem allomorph</text></form>
    <trait name="environment" value="/ [V] _"></trait>
    <trait name="morph-type" value="stem"></trait>
  </variant>
  <variant>
  </entry>
```

A second main entry for testing: newmain

```
<entry dateCreated="2010-01-12T17:38:13Z" dateModified="2010-01-12T17:39:06Z" guid="42368b9d-c1f1-49ef-b155-2f255e88b952" id="newmain_42368b9d-c1f1-49ef-b155-2f255e88b952">
  <lexical-unit>
    <form lang="x-kal"><text>newmain</text></form>
  </lexical-unit>
  <trait name="morph-type" value="stem"></trait>
</entry>
```

A subentry contains at least one relation element with a type of _component-lexeme and a ref of the id for the main entry. It has a trait element with a name of complex-form-type and a value that indicates the type of subentry which comes from the Flex ComplexForm Types list. In this example, it is classified as a Compound and Derivative, illustrating that you can have more than one type. The trait is-primary indicates this subentry should be shown as a subentry in the main entry. In this example subentry is a complex form of main and newmain, but it is only listed in the dictionary under the main entry since the newmain reference does not have is-primary set to true. The relation element could also have a hide-minor-entry trait to indicate the subentry is not shown as a separate minor entry in the root-based view of the lexicon.

```
<entry dateCreated="2004-10-30T01:03:28Z" dateModified="2007-01-17T17:24:04Z" guid="157edd55-886f-4d91-b009-8e6b49991c85" id="subentry_157edd55-886f-4d91-b009-8e6b49991c85">
  <lexical-unit>
    <form lang="x-kal"><text>subentry</text></form>
  </lexical-unit>
  <trait name="morph-type" value="stem"></trait>
```

```

<relation type="_component-lexeme" ref="main_1a705846-a814-4289-8594-
4b874faca6cc" order="0">
<trait name="complex-form-type" value="Compound"></trait>
<trait name="complex-form-type" value="Derivative"></trait>
<trait name="is-primary" value="true"/>
</relation>
<relation type="_component-lexeme" ref="newmain_42368b9d-c1f1-49ef-b155-
2f255e88b952" order="1">
</relation>

```

A variant entry contains at least one relation element with a type of `_component-lexeme` and a ref of the id for the main entry. It has a trait element with a name of `variant-type` and a value that indicates the type of variant entry which comes from the Flex Variant Types list--`Inflectional Variant` in this case. The trait `is-primary` indicates this subentry should be listed as part of the main entry. The `hide-minor-entry` trait indicates that in a root-based view of the lexicon, there will not be a minor entry outside of main for this variant.

```

<entry dateCreated="2004-10-30T01:05:18Z" dateModified="2007-01-
17T17:23:36Z" guid="a6babda6-6830-4ec0-a363-c7fba14268eb"
id="variantentry_a6babda6-6830-4ec0-a363-c7fba14268eb">
<lexical-unit>
<form lang="x-kal"><text>variantentry</text></form>
</lexical-unit>
<trait name="morph-type" value="stem"></trait>
<relation type="_component-lexeme" ref="main_1a705846-a814-4289-8594-
4b874faca6cc">
<trait name="variant-type" value="Inflectional Variant"></trait>
<trait name="hide-minor-entry" value="-1"/>
</relation>
</entry>

```

Some areas of LIFT have not been nailed down enough to be very helpful. At this point `_component-lexeme` is common across all FLEX LIFT files, but I'm not sure WeSay uses this concept, and I doubt anyone producing a LIFT file from from a SFM dictionary via SOLID will have this. There are a number of gray areas like this that should probably be nailed down as part of the standard.

2.6 Full Lexicon (root/stem-based) SFM

This process exports the entire dictionary, independent of any settings in Flex. The output is standard format unsorted UTF-8 data with NFC normalization. Here is an example of the output:

```

\lx underlying form
\lxKal underlying form
\lcKal ztestmain
\va ztestminor

```



```

\va variant
\lf cf
\lv ztestsub
\sn 1
\ps noun
\gEng English gloss
\dEng This is the English definition
\reEng house
\reEng bungalo
\xvKal Vernacular example
\xEng English translation
\xvKal Second vernacular example
\xEng Second English translation
\lf pt
\lv pal
\lf pt

```

This file is similar to MDF, but has a number of differences because FieldWorks knows nothing about national and regional languages. Instead, there are any number of vernacular and any number of analysis writing systems. Thus, for definitions Flex uses `\d` plus the writing system abbreviation set in the Writing System Property dialog (e.g., `\dEng English`, `\dSpa Spanish`). Any field that is a MultiString or MultiUnicode is dumped in this way.

The stem-based version will export subentry bodies in separate `\lx` entries with a reference to it from the main entry. The root-based version will export subentry bodies appended to the main entry. It should also include a separate `\lx` entry that simply refers to the main entry, but this is not working in FieldWorks 5.0.

If you need to export custom fields, you will need to modify `c:\Program Files\SIL\FieldWorks\Language Explorer\Export Templates\mdf.xml` (stem-based) or `RootBasedMDF.xml` (root-based) to specify how to dump the custom fields. See `FXT export options.doc` for help.

3 Lists

Most lists can be exported in a single file using different formats.

3.1 Lists SFM

This exports lists in Standard Format. This is a small portion of the file showing the first two semantic domains. All checked analysis writing systems are exported.

```

\lna Semantic Domains
\lab Sem
\ a_Eng 1
\ n_Eng Universe, creation
\ d_Eng Use this domain for general words referring to the physical universe. Some
languages may not have a single word for the universe and may have to use a phrase
such as 'rain, soil, and things of the sky' or 'sky, land, and water' or a descriptive

```

phrase such as 'everything you can see' or 'everything that exists'.

\q_Eng (1) What words refer to everything we can see?

\w_Eng universe, creation, cosmos, heaven and earth, macrocosm, everything that exists

\s_Eng In the beginning God created <the heavens and the earth>.

\a_Eng 1.1

\n_Eng Sky

\d_Eng Use this domain for words related to the sky.

\q_Eng (1) What words are used to refer to the sky?

\w_Eng sky, firmament, canopy, vault

\q_Eng (2) What words refer to the air around the earth?

\w_Eng air, atmosphere, airspace, stratosphere, ozone layer

\q_Eng (3) What words are used to refer to the place or area beyond the sky?

\w_Eng heaven, space, outer space, ether, void, solar system

\q_Eng (4) What words describe something in the sky or something that happens in the sky?

\w_Eng aerial, heavenly, atmospheric, stratospheric, celestial,

\q_Eng (5) What words describe the appearance of the sky?

\w_Eng blue, clear, black (night), cloudy, cloud filled, brilliant, dark, night sky, foreboding, stormy, dark, star studded

\q_Eng (6) What words refer to the edge of the sky where it meets the ground?

\w_Eng horizon, skyline

\q_Eng (7) What words refer to something in the sky?

\w_Eng heavenly body, celestial body, luminary

\q_Eng (8) What words refer to the lights that appear in the northern (or southern) sky?

\w_Eng aurora borealis, northern lights

\q_Eng (9) What words refer to something being in the sky?

\w_Eng aloft, up in the sky, up in the air,

3.2 Lists Tab-delimited

This exports lists as tab-delimited text. This is a small portion of the file showing several semantic domains.

Semantic Domains

| Abbr. | Name |
|---------|--------------------|
| 1 | Universe, creation |
| 1.1 | Sky |
| 1.1.1 | Sun |
| 1.1.1.1 | Moon |
| 1.1.1.2 | Star |
| 1.1.1.3 | Planet |
| 1.1.2 | Air |
| 1.1.2.1 | Blow air |
| 1.1.3 | Weather |
| 1.1.3.1 | Wind |

3.3 Lists XML

This exports lists as XML. This is the only form that maintains hierarchy in list records. This is a small portion of the file showing the first two semantic domains. All checked analysis writing systems are exported.

```

<list>
<name ws="Eng">Semantic Domains</name>
<abbr ws="Eng">Sem</abbr>
<item>
<abbr ws="Eng">1</abbr>
<name ws="Eng">Universe, creation</name>
<descr ws="Eng">Use this domain for general words referring to the physical
universe. Some languages may not have a single word for the universe and may have
to use a phrase such as 'rain, soil, and things of the sky' or 'sky, land, and water' or a
descriptive phrase such as 'everything you can see' or 'everything that exists'.</descr>
<questionGroup>
<question ws="Eng">(1) What words refer to everything we can see?</question>
<words ws="Eng">universe, creation, cosmos, heaven and earth, macrocosm,
everything that exists</words>
<sentences ws="Eng">In the beginning God created &lt;the heavens and the
earth&gt;.</sentences></questionGroup>

<item>
<abbr ws="Eng">1.1</abbr>
<name ws="Eng">Sky</name>
<descr ws="Eng">Use this domain for words related to the sky.</descr>
<questionGroup>
<question ws="Eng">(1) What words are used to refer to the sky?</question>
<words ws="Eng">sky, firmament, canopy, vault</words>
</questionGroup>
<questionGroup>
<question ws="Eng">(2) What words refer to the air around the earth?</question>
<words ws="Eng">air, atmosphere, airspace, stratosphere, ozone layer</words>
</questionGroup>
<questionGroup>
<question ws="Eng">(3) What words are used to refer to the place or area beyond the
sky?</question>
<words ws="Eng">heaven, space, outer space, ether, void, solar system</words>
</questionGroup>
<questionGroup>
<question ws="Eng">(4) What words describe something in the sky or something
that happens in the sky?</question>
<words ws="Eng">aerial, heavenly, atmospheric, stratospheric, celestial, </words>
</questionGroup>
<questionGroup>
<question ws="Eng">(5) What words describe the appearance of the sky?</question>
<words ws="Eng">blue, clear, black (night), cloudy, cloud filled, brilliant, dark,

```

```

night sky, foreboding, stormy, dark, star studded</words>
</questionGroup>
<questionGroup>
<question ws="Eng">(6) What words refer to the edge of the sky where it meets the
ground?</question>
<words ws="Eng">horizon, skyline</words>
</questionGroup>
<questionGroup>
<question ws="Eng">(7) What words refer to something in the sky?</question>
<words ws="Eng">heavenly body, celestial body, luminary</words>
</questionGroup>
<questionGroup>
<question ws="Eng">(8) What words refer to the lights that appear in the northern (or
southern) sky?</question>
<words ws="Eng">aurora borealis, northern lights</words>
</questionGroup>
<questionGroup>
<question ws="Eng">(9) What words refer to something being in the
sky?</question>
<words ws="Eng">aloft, up in the sky, up in the air, </words>
</questionGroup>
...
</item>
</item>
</list>

```

4 Reversal Indexes

The reversal indexes can be exported in several ways.

4.1 Reversal Index XML

This option is only available when you are in the Reversal Indexes view. It exports an XML file of the reversal, including nested reversal items and sense references. If the index has more than one index writing system enabled, they are all supposed to be exported, but that is currently not working in FieldWorks 5.0.

```

<?xml version="1.0" encoding="utf-8"?>
<ReversalIndex>
<ReversalIndexEntry id="hvo59163">
  <ReversalIndexEntry_ReversalForm><AUni ws="en">a little</AUni>
  </ReversalIndexEntry_ReversalForm>
  <ReversalIndexEntry_RelatedForm />
  <ReversalIndexEntry_ReferringSenses>
    <Link><Alt ws="ii" sense="ἡ ἄστὴρ" /></Link>
  </ReversalIndexEntry_ReferringSenses>
  <ReversalIndexEntry_Subentries>
  </ReversalIndexEntry_Subentries>

```

```

</ReversalIndexEntry>
<ReversalIndexEntry id="hvo59203">
  <ReversalIndexEntry_ReversalForm><AUni ws="en">a pity</AUni>
</ReversalIndexEntry_ReversalForm>
  <ReversalIndexEntry_RelatedForm />
  <ReversalIndexEntry_ReferringSenses>
    <Link><Alt ws="ii" sense="𐌆𐌆 2" /></Link>
  </ReversalIndexEntry_ReferringSenses>
  <ReversalIndexEntry_Subentries>
  </ReversalIndexEntry_Subentries>
</ReversalIndexEntry>
.....

```

4.2 Reversal Index SFM

This option is only available when you are in the Reversal Indexes view. It exports a standard format file of the reversal, including nested reversal items and sense references. If the index has more than one index writing system enabled, they were all supposed to be exported, but that is currently not working in FieldWorks 4.0. The output is UTF-8 with NFC normalization. Note that record end markers (\-reen) are used to allow hierarchical indexes:

```

\reen a little
\sn 𐌆𐌆𐌆𐌆
\-reen

```

```

\reen a pity
\sn 𐌆𐌆 2
\-reen

```

4.3 Reversal Index Detail SFM

This exports all reversal indexes into a single SFM file. For every sense reference, it exports all of the normal MDF sense fields instead of just the headword.

```

\ind en
\ws_Eng English
\rev_Eng house
\sn haus
\ps noun
\g_Eng house
\d_Eng This is the English definition
\is_Eng 1
\sd_Eng Universe, creation
\sd_Frn L'univers physique

\rev_Eng bungalo
\sn haus
\ps noun

```

```

\g_Eng house
\d_Eng This is the English definition
\is_Eng 1
\sd_Eng Universe, creation
\sd_Frn L'univers physique

```

5 Wordforms

There are several ways to export the words area. These exports are quite slow because of the time to calculate the counts.

5.1 Wordforms SFM

This exports the wordforms, analyses, and counts in SFM format.

```

\w_Kal nihimbilira
\g_Eng I see
\g_Eng I perceive
\g_Eng I understand
\ps verb
\fc 0
\uc 1
\pc 1

```

```

\w_Kal tihindoksa
\fc 0
\uc 0
\pc 1

```

5.2 Wordforms Tab-delimited

This exports the wordforms, analyses, and counts in tab-delimited format.

| Wordforms | Gloss | Cat | NumInCorp | UserAnals | PredictedAnals |
|-------------|-----------------|------|-----------|-----------|----------------|
| nihimbilira | I seeI perceive | verb | 0 | 1 | 1 |
| tihindoksa | | | 0 | 0 | 1 |

5.3 Wordforms XML

This exports the wordforms, analyses, and counts in XML: format.

```

<?xml version="1.0" encoding="UTF-8"?>
<wordform PredictedAnalyses='1' UserAnalyses='1'>
<form ws="Kal">nihimbilira</form>
<analysis><gloss ws="Eng">I see</gloss>
<gloss ws="Eng">I perceive</gloss>
<gloss ws="Eng">I understand</gloss>
<category>verb</category></analysis>
<analysis></analysis>
</wordform>
<wordform PredictedAnalyses='1' UserAnalyses='0'>

```

```
<form ws="Kal">tihindoksa</form>  
<analysis></analysis>  
</wordform>
```

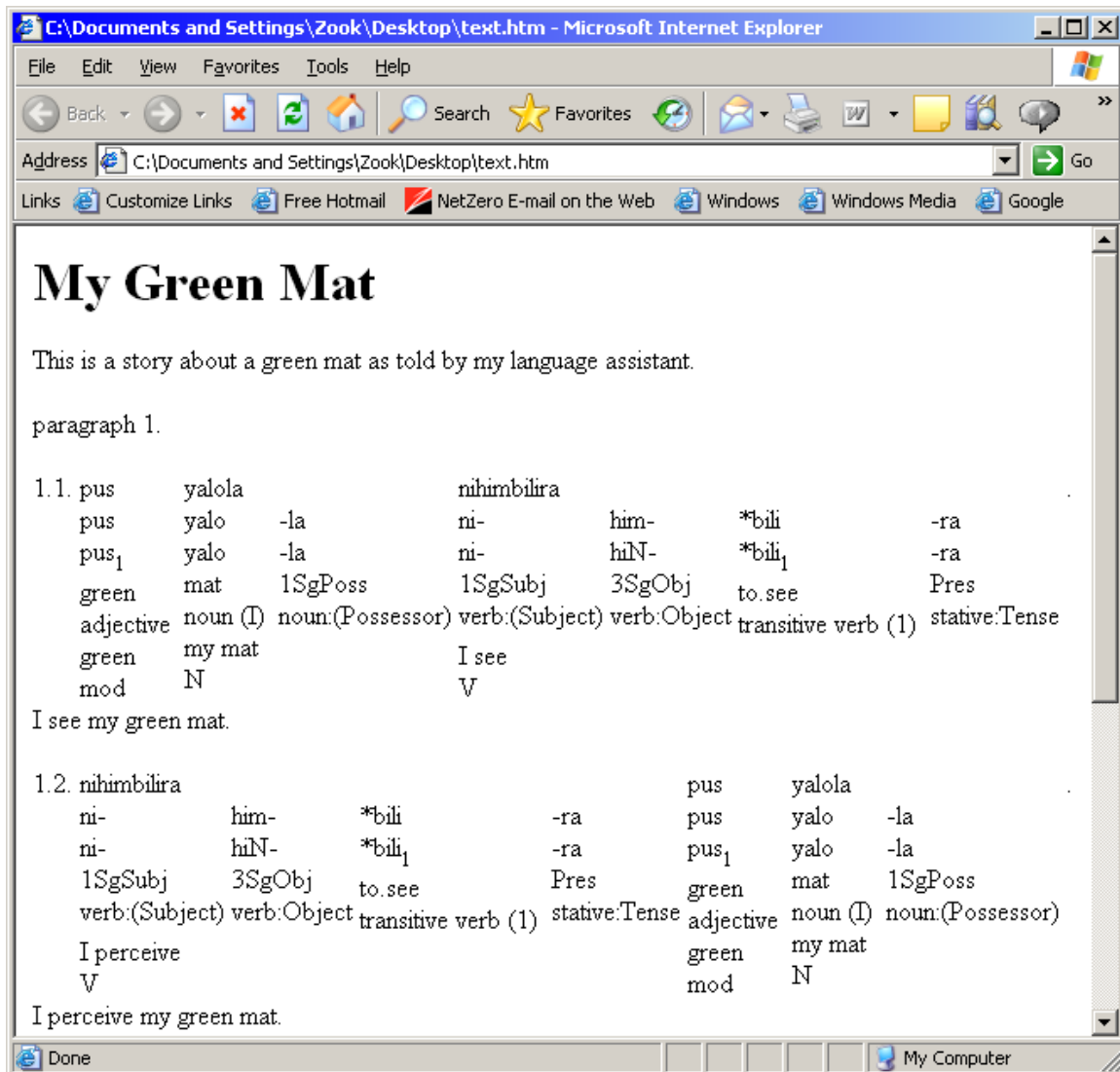
6 Interlinear text

There are various ways to export interlinear text. When you are in the Texts & Words area, you can choose File...Export Interlinear, and then choose one of the options in the Export Interlinear dialog. The export uses the current layout of the text based on Tools...Configure Interlinear.

These options leave a temporary Phase1-*.htm file in the output directory that need to be deleted.

6.1 HTML

This option exports an interlinearized text to a simple HTML format using piles that will display properly in Internet Explorer or FireFox. This is a form suitable for web publishing.



6.2 Interlinear XML

This option exports an interlinearized text to a generic XML file suitable for transforming to other formats with XSLT or some other process. This is an example of the output as viewed in Internet Explorer:

```
<?xml version="1.0" encoding="utf-8" ?>
- <document>
- <interlinear-text>
- <languages>
  <language lang="xkal" font="Times New Roman" vernacular="true" />
  <language lang="" font="Times New Roman" />
  <language lang="en" font="Times New Roman" />
</languages>
<item type="title" lang="en">My Green Mat</item>
<item type="description" lang="en">This is a story about a green mat as told
```



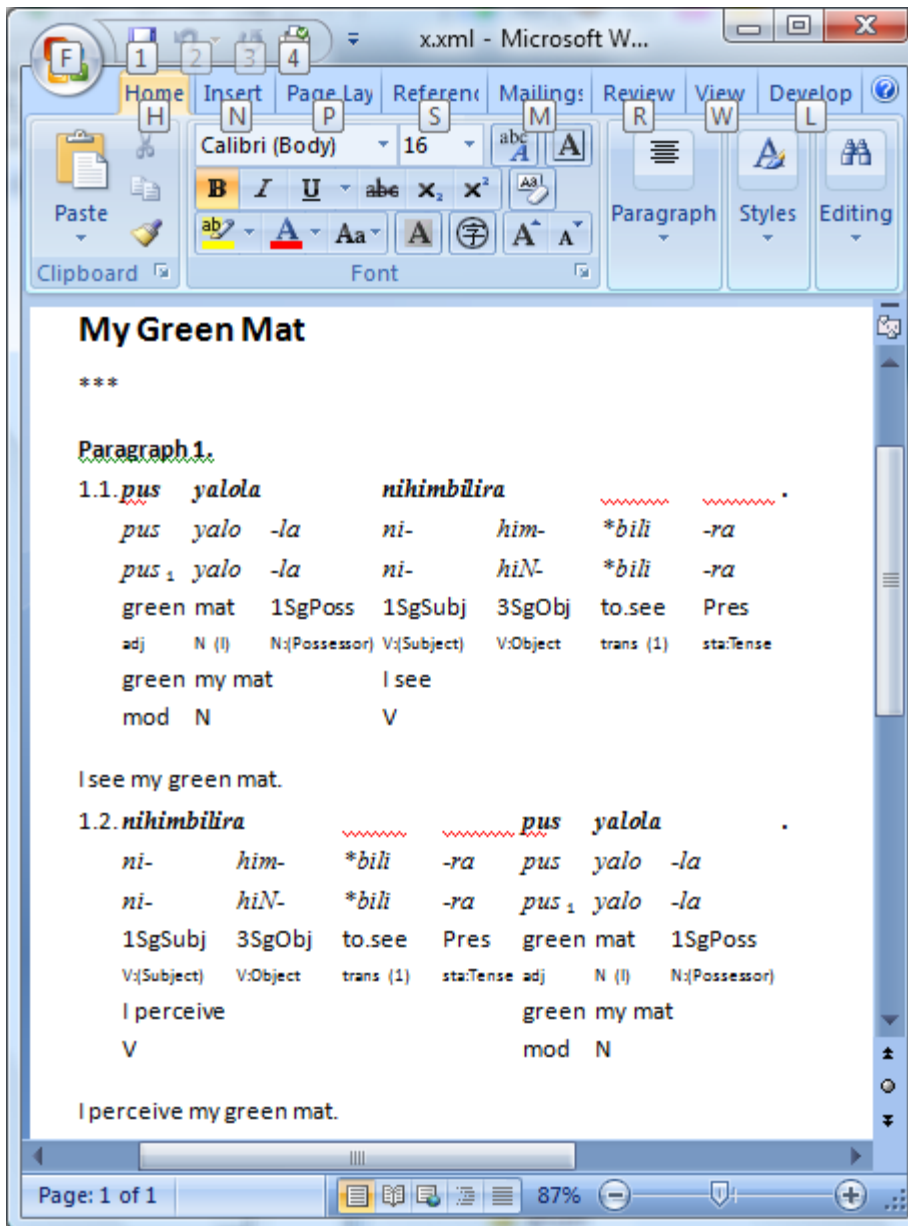
```

by my language assistant.</item>
- <paragraphs>
- <paragraph>
- <phrases>
- <phrase>
- <words>
- <word>
  <item type="txt" lang="xkal">pus</item>
- <morphemes>
- <morph type="root">
  <item type="txt" lang="xkal">pus</item>
  <item type="cf" lang="xkal">pus</item>
  <item type="hn" lang="en">1</item>
  <item type="gls" lang="en">green</item>
  <item type="msa" lang="en">adjective</item>
  </morph>
  </morphemes>
  <item type="gls" lang="en">green</item>
  <item type="pos" lang="en">mod</item>
  </word>
+ <word>
+ <word>
+ <word>
  </words>
  <item type="gls" lang="en">I see my green mat.</item>
  </phrase>
+ <phrase>
+ <phrase>
+ <phrase>
  </phrases>
  </paragraph>
  </paragraphs>
  </interlinear-text>
  </document>

```

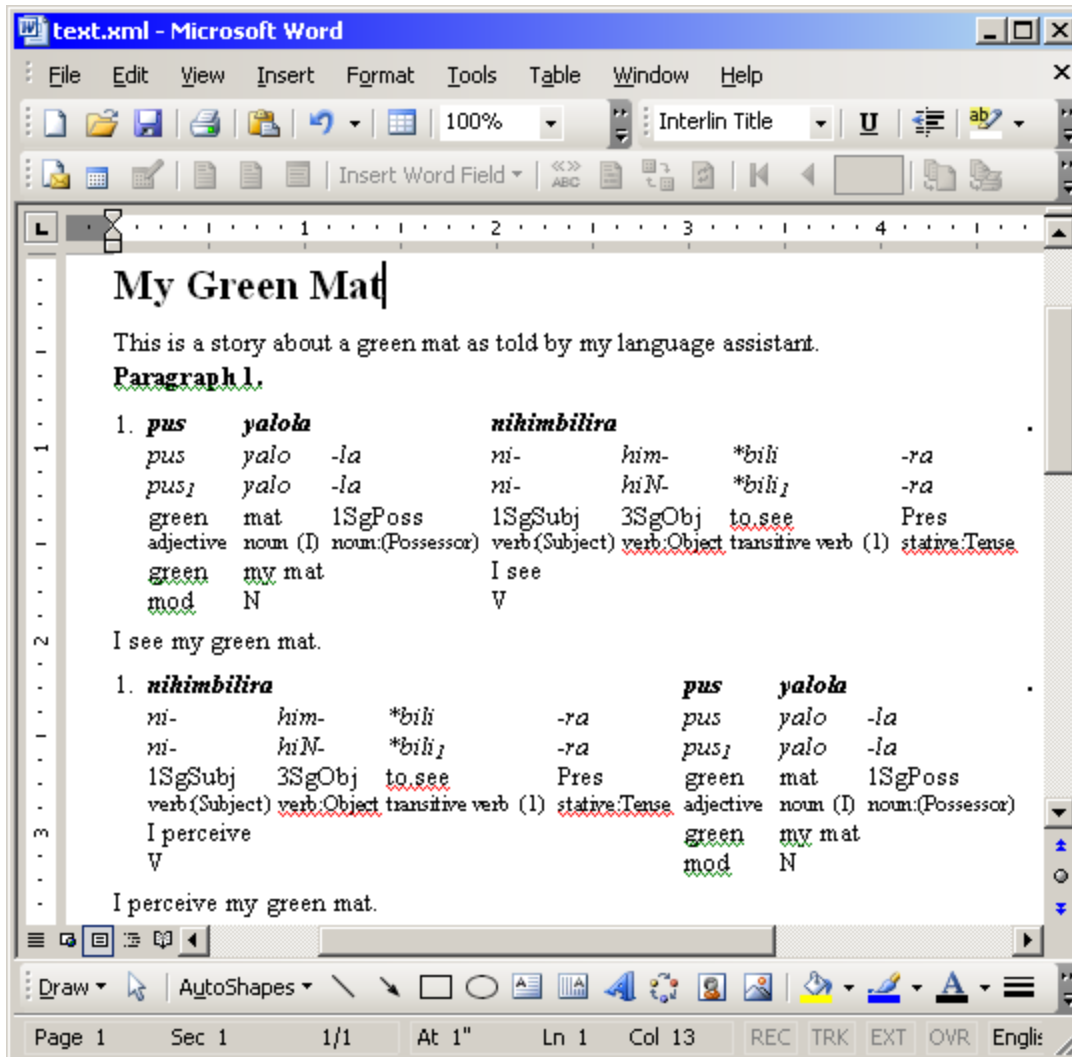
6.3 Microsoft Word 2007 XML

This option exports an interlinearized text to an XML file that can be read by Microsoft Word 2007. Each part of the interlinear bundle is a different style, so you can modify the Word stylesheet for different appearances.



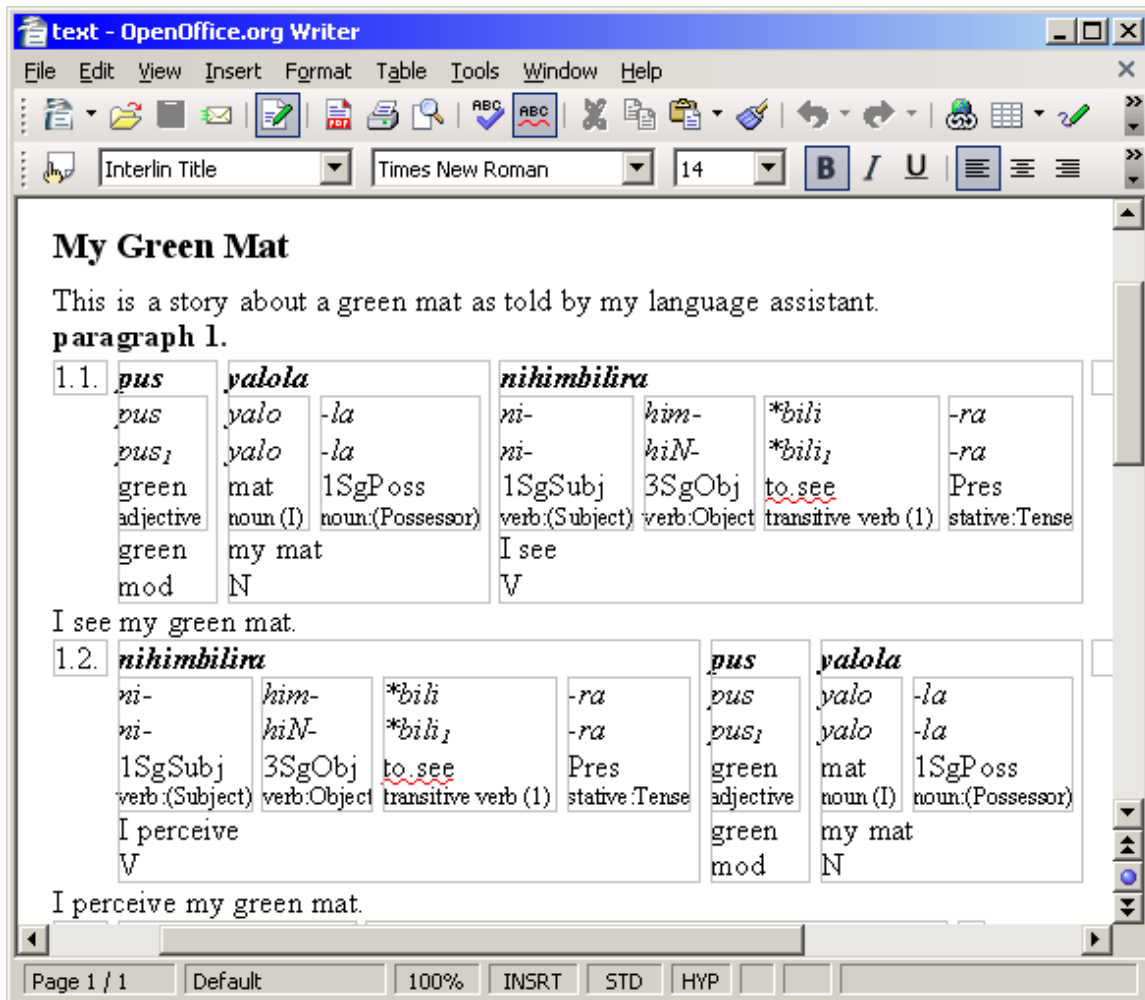
6.4 Microsoft Word XML

This option exports an interlinearized text to an XML file that can be read by Microsoft Word (2003). Before the text is formatted properly, you need to run a Word macro. The instructions for doing this are in c:\Program Files\SIL\FieldWorks\Language Explorer\Export Templates\Interlinear\Interlinear Macros and Instructions.doc. Each part of the interlinear bundle is a different style, so you can modify the Word stylesheet for different appearances.



6.5 Open Office Writer ODT

This option exports an interlinearized text to an Open Office 2.0 ODT file.



6.6 Pretty HTML

This option exports an interlinearized text to an html file that is can have various lines turned on and off in the view. This is not available in the current release, but can be sent to anyone that would like it.

Interlinear text: "My Green Mat" - Windows Internet Explorer

C:\Documents and Settings\Zook\Desktop\lx.htm

Search by Yahoo!

File Edit View Favorites Tools Help

Interlinear text: "My Green Mat"

Display Interlinear Lines: Morpheme Group MG morpheme MG ref. MG gloss
 MG gram. info word gloss pos free translation

Best viewed with Firefox 3

Text Title: My Green Mat

paragraph 1.

1.1. pus yalola nihimbilira .

| | | | | | | |
|-------|--------|----------------|--------------|-----------|-----------|------------|
| pus | yalola | | nihimbilira | | | |
| pus | yalo | -la | ni- | him- | *bili | -ra |
| pus | yalo | -la | ni- | hiN- | *bili | -ra |
| green | mat | 1SgPoss | 1SgSubj | 3SgObj | to.see | Pres |
| adj | N (I) | N: (Possessor) | V: (Subject) | V: Object | trans (1) | sta: Tense |
| green | my mat | | I see | | | |
| mod | N | | V | | | |

Free translation: I see my green mat.

1.2. nihimbilira pus yalola .

| | | | | | | |
|--------------|-----------|-----------|------------|-------|--------|----------------|
| ni- | him- | *bili | -ra | pus | yalola | -la |
| ni- | hiN- | *bili | -ra | pus | yalo | -la |
| 1SgSubj | 3SgObj | to.see | Pres | green | mat | 1SgPoss |
| V: (Subject) | V: Object | trans (1) | sta: Tense | adj | N (I) | N: (Possessor) |
| I perceive | | | | green | my mat | |
| V | | | | mod | N | |

Free translation: I perceive my green mat.

Done

My Computer

100%

6.7 Verifiable generic XML

This option exports an interlinearized text to a generic XML file that can be verified by an XSD schema. This may be useful for further transforms that require a verifiable input, but does not preserve the order of the interlinear lines. This is an example of the output as viewed in Internet Explorer:

```
<?xml version="1.0" encoding="utf-8" ?>
- <document xsi:noNamespaceSchemaLocation="file:FlexInterlinear.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
- <interlinear-text>
  <item type="title" lang="en">My Green Mat</item>
  <item type="title-abbreviation" lang="en">***</item>
```

```

- <paragraphs>
- <paragraph>
- <phrases>
- <phrase>
- <words>
- <word>
  <item type="txt" lang="xkal">pus</item>
  <item type="gls" lang="en">green</item>
  <item type="pos" lang="en">mod</item>
- <morphemes>
- <morph type="root" guid="d7f713e5-e8cf-11d3-9764-00c04f186933">
  <item type="txt" lang="xkal">pus</item>
  <item type="cf" lang="xkal">pus</item>
  <item type="hn" lang="en">1</item>
  <item type="gls" lang="en">green</item>
  <item type="msa" lang="en">adj</item>
</morph>
</morphemes>
</word>
- <word>
  <item type="txt" lang="xkal">yalola</item>
  <item type="gls" lang="en">my mat</item>
  <item type="pos" lang="en">N</item>
- <morphemes>
- <morph type="root" guid="d7f713e5-e8cf-11d3-9764-00c04f186933">
  <item type="txt" lang="xkal">yalo</item>
  <item type="cf" lang="xkal">yalo</item>
  <item type="gls" lang="en">mat</item>
  <item type="msa" lang="en">N (I)</item>
</morph>
- <morph type="suffix" guid="d7f713dd-e8cf-11d3-9764-00c04f186933">
  <item type="txt" lang="xkal">-la</item>
  <item type="cf" lang="xkal">-la</item>
  <item type="gls" lang="en">1SgPoss</item>
  <item type="msa" lang="en">N:(Possessor)</item>
</morph>
</morphemes>
</word>
- <word>
</words>
  <item type="gls" lang="en" />
</phrase>
</phrases>
</paragraph>
</paragraphs>

```

```
- <languages>
  <language lang="en" font="Times New Roman" />
  <language lang="xkal" font="Times New Roman" vernacular="true" />
</languages>
</interlinear-text>
</document>
```

6.8 XLingPaper as examples, align morphemes XML

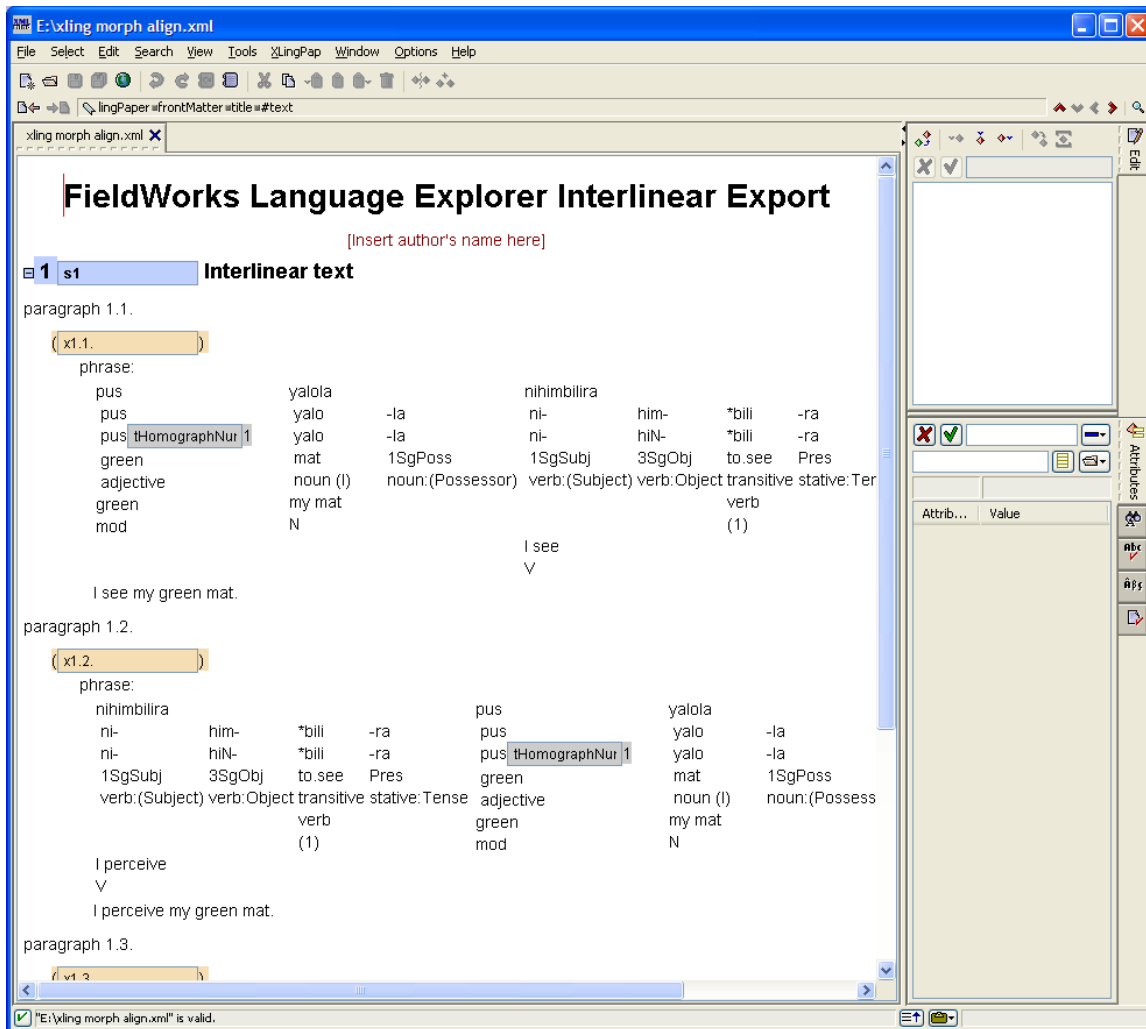
This option exports an interlinearized text as a sequence of examples to a basic XLingPaper document which can then be used to cut and paste into another XLingPaper document (e.g., the XLingPaper form of the Grammar Sketch). The final printed view shows morphemes as they are shown in Flex.

6.9 XLingPaper as examples, align words XML

This option exports an interlinearized text as a sequence of examples to a basic XLingPaper document which can then be used to cut and paste into another XLingPaper document (e.g., the XLingPaper form of the Grammar Sketch). The final printed view concatenates all the morphemes in a word together, so there are not separate morpheme columns within a word.

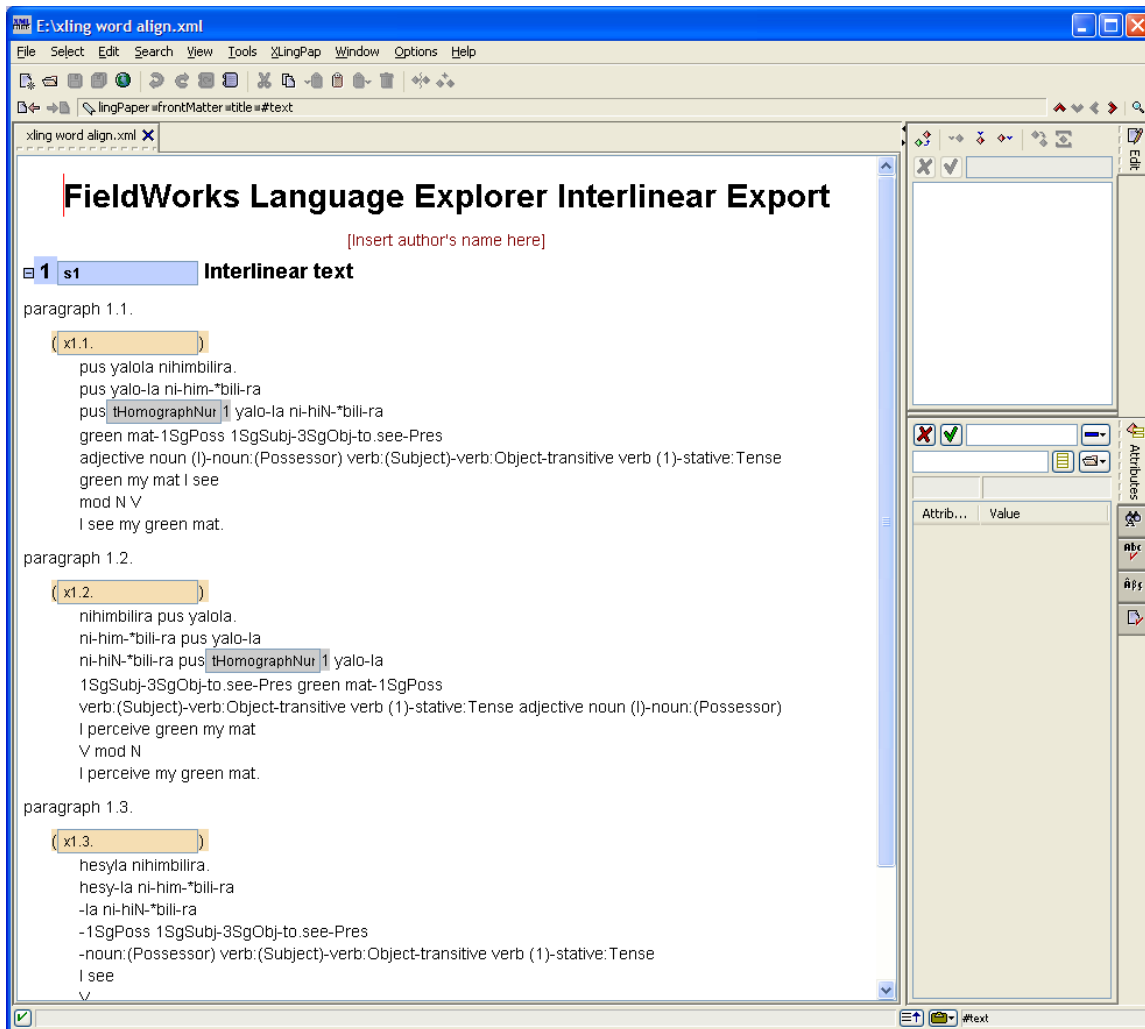
6.10 XLingPaper, align morphemes XML

This option exports an interlinearized text to a basic XLingPaper document that can then be used to cut and paste into another XLingPaper document (e.g., the XLingPaper form of the Grammar Sketch). XLingPaper is a way to write linguistic papers using XML technologies in an XML editor such as the freely available XMLmind XML Editor with the XLingPaper configuration files. See <http://www.sil.org/~blacka/xlingpap/index.htm> or the installation CD-ROM at D:\FieldWorks\XML Utilities\XLingPap.htm for more information. The final printed view is similar to what you see in Flex.



6.11 XLingPaper, align words XML

This option exports an interlinearized text to a basic XLingPaper document. The final printed view concatenates all the morphemes in a word together, so there are not separate morpheme columns within a word.



6.12 XLingPaper, as a single example, align words XML

This option exports the interlinear text as an example containing a sequence of listInterlinear elements to a basic XLingPaper document which can then be used to cut and paste into another XLingPaper document (e.g. the XLingPaper form of the Grammar Sketch). XLingPaper is a way to write linguistic papers using XML technologies.

7 Grammar sketch

The Grammar area provides a Grammar Sketch tool that can generate a grammar sketch of your morphology at any time. This grammar sketch is normally shown inside Flex, but can be exported by clicking Save in the special toolbar. This requests a file for a standard HTML export and then asks for a file for XLingPaper XML format. The XML file can be opened in XML Mind for editing using the XLingPaper styles.

8 Phonology Assistant

The new Phonology Assistant program reads data directly from the FieldWorks database, so an actual export is not necessary. It automatically updates as you make changes in Flex.

9 Lexique Pro

When you export your lexicon using the MDF format, you can then import this into Lexique Pro to produce various types of output including Word documents and Web pages. This process requires configuring Lexique Pro to handle the languages and SFM markers exported from Fieldworks.

A second approach is to export LIFT format. Lexique Pro is able to load the LIFT format and extract writing system information from FieldWorks, so that basic configuration is done automatically.

10 Python

Python programs can be written to export any data from the database in any format you want. See Python database access.doc for information on doing this.