WeSay Walkthrough

WeSay was made by the Payap Language Software group
© 2007 – 2009
Available under the open source MIT license.

This documentation written by Nolan Adams
© SIL International 2009.

WeSay Walkthrough

Table of Contents

About This Document....1
Who Should Read This....1
What You Will Learn....1
Document Conventions....1
About WeSay....2
Who WeSay is For....2
Task-Based Approach to Dictionary Making....2
Gloss vs. Definition....4
Introduction....5
HowTo Install WeSay on Windows....6
Downloading WeSay....6
Running the Installer....6
HowTo Install WeSay on Linux....8
Getting the APT Lines....8
Installing the Repositories....8
HowTo Setup For a New Project....12
Creating A New Project....12
Writing System Setup....13
HowTo Interact with FLEx....17
Importing LIFT Into WeSay....17
Exporting LIFT From WeSay....17
HowTo Add an Additional Language....19
Steps to Add Additional Languages....19
Setting Displayed Writing Systems....19
HowTo Control Fields in WeSay....25
HowTo Record Audio....27
Setting up an Audio Writing System....27
Recording with WeSay....28
HowTo Change the Interface Language....30
Accessing User Interface Languages....30
HowTo Change WeSay Tasks....32
Tasks Menu....32
Actions Menu....33
Uses for Tasks....34
Reference....36
Tasks....36
Dashboard....36
Dictionary Browse & Edit....37
Add Meanings....38
Add Parts of Speech....38
Add Example Sentences....39
Add Base Forms....40

Gather Words Using the SIL Comparative African Wordlist....40
Gather Words by Semantic Domain....41
Writing Systems....42
Basic....42
Font....43
Sorting....43
Fields....43
Actions....45
Email This Dictionary....46
Backup To Device....46
Export to SFM....46
Make PDF Dictionary....46
Open in Lexique Pro....47

About This Document

Who Should Read This

This document is intended for people beginning to use WeSay, or for those looking for a reference pertaining to WeSay's features. No technical knowledge is required.

What You Will Learn

The first section is a series of steps showing how to get started with WeSay, as well as explaining the features of WeSay and the WeSay Configuration Tool. The second section is a reference so you can easily look up any features or tools in WeSay.

Document Conventions

In this document you will find three basic signs that denote importance. There are:

Info signs, which are for clarification or expanding on the topic;

Lightbulbs, which indicate tips and extra information;

and Warning signs, which are for areas easy to mistake.

There are also two character styles that indicate important areas:

A set of words in italics, separated by a series of | marks, shows a file path you need to use. For instance, *My Documents* | *My Music* means you need to open My Documents and then open the My Music folder.

Any words that are **bold** show specific actions you need to do. They will help you easily find the steps in a process. For instance, in some cases you might need to **open the Start menu**, found on the menu bar. Then **go to Control Panel** and **find the Regional Settings**.

About WeSay

Who WeSay is For

WeSay is intended as a program for a team working on a dictionary project. It is built so that one person with computer experience, the Adviser, can set up WeSay and allow people with less computer experience, the Vernacular Language Experts (VLEs), to add to the dictionary.

WeSay is not intended as a program for a single linguist to build an entire dictionary. More powerful programs, like FLEx, are better for linguists to construct lexicons.

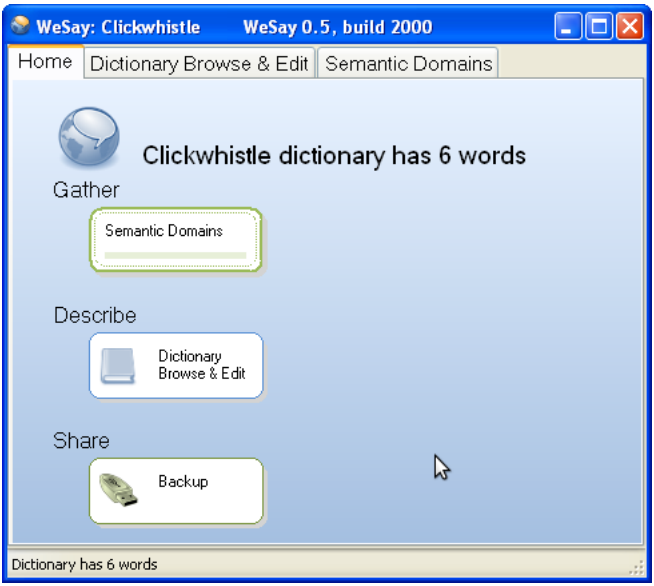
The WeSay Configuration Tool is the program that the Adviser should use in order to start the dictionary and set up the fields. With the Configuration Tool, the Adviser can also focus WeSay to only do one specific task at a time.

A VLE working with the Adviser would use the actual WeSay program that the Adviser set up in order to add to the dictionary.

Task-Based Approach to Dictionary Making

The WeSay Configuration Tool controls everything that appears in WeSay. By changing settings in the Configuration Tool, you can change the entire functionality of WeSay.

For example, when WeSay first opens, it has been set for the functionality of adding words with 'Semantic Domain', editing the words in 'Dictionary Browse & Edit', and backing up the dictionary with 'Backup'.



Later on you can change this functionality by adding or removing buttons on this page. This is useful because, by using different buttons and functionality, the Adviser can customize WeSay for the unique skills of each VLE. WeSay can be set up to allow each VLE to individually complete separate tasks without being overwhelmed by dozens of options and buttons.

At first the Adviser might want to add words to his dictionary. He could use the Configuration Tool to set up WeSay for Semantic Domain input. He could then have a VLE who only speaks the vernacular come and enter words for a few months.

Once he has a large number of words in his dictionary, the Adviser would probably want to start entering definitions for all these words. He could set up WeSay, then, to only allow definitions to be entered, and have a VLE who knows both the vernacular and English to go through and translate all the words.

Later on he could set up WeSay to allow only example sentences to be entered, and have a VLE who is good at example sentences add example sentences to all of the dictionary entries.

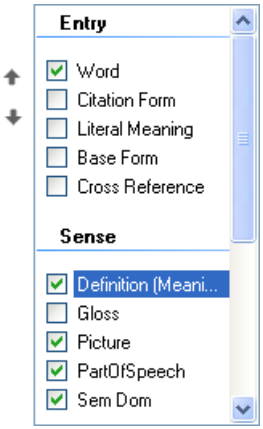
WeSay can be set up to focus individually on Meanings, Parts of Speech, Example Sentences or Base Forms.

By working through these steps, you could complete a dictionary with a group of

people who are only capable of one step each.

Gloss vs. Definition

In WeSay we prefer to use definition over gloss, which is intended for use with inter-linear texts. You have the option to enable glosses, but by default the program is set up to only use definitions.



The word descriptions entered in definitions can be many words, much longer than a typical gloss, but they don't necessarily have to be. For a dictionary that is translating from one language to another, one or two word definitions are often sufficient.

Introduction

This manual will show you all the setup necessary to be able to work with WeSay as discussed in the 'About WeSay' section. For this walkthrough, we will be using an example project of a team making a dictionary for the imaginary language Clickwhistle.

For each step in the process of setting up WeSay, we will relate the configuration to the goal of translating from Clickwhistle to English.

This manual is not strictly a tutorial, however. Each section is a stand-alone HowTo guide, so if you only need to know how to complete one specific task, you can find that task in the table of contents and read that section specifically.

Our first step in building a Clickwhistle dictionary is to install WeSay, which will be covered in the next section. There are separate sections for installation in either Linux or Windows.

HowTo Install WeSay on Windows

Our first step in starting our Clickwhistle dictionary is to get the tool we will be using to make our dictionary, WeSay.

Downloading WeSay

In order to install WeSay, we need the installer. You can find this installer on the WeSay web page by **clicking on this link** or copying it into your browser: www.wesay.org/downloads. In this index, **click on the most recent WeSay update**.

There are two versions of WeSay for you to choose from – WeSay.0.4, which is the stable version, and WeSay.0.5, which is the development version. This version has more features but there may also be some issues.

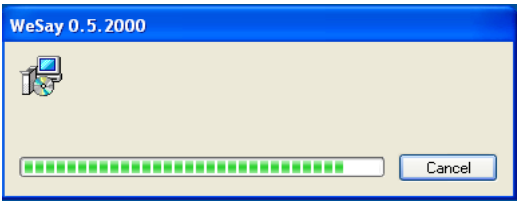
The Windows version of WeSay currently runs on Windows XP, Vista, and Windows 2000. It requires Microsoft .Net Framework 2.0, and Windows Installer 3.1 or later.

To find the right file, look for a name like 'WeSayInstaller.0.4.1767.msi' that has a size of roughly 6.0 – 7.0 M and the description 'WeSay Installer'.

Check the WeSay website, www.wesay.org, for more information about this product. The FAQ (http://www.wesay.org/wiki/FAQ) contains information about system requirements and WeSay details.

Running the Installer

After you've clicked on the link, **select download** and **wait for the file**. Once the file has finished downloading, **double-click on the icon** to start the setup. You should see a pop-up appear like the one below.



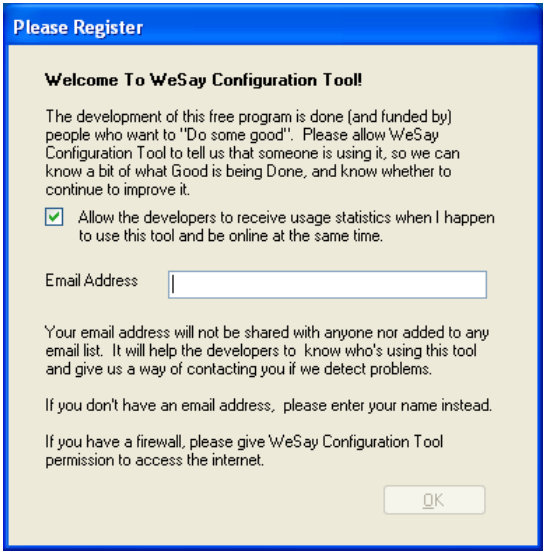
Note that there will be no setup menu asking you to confirm the installation. When the welcome text appears, WeSay will have already finished installing.

Now that WeSay is installed on your computer, you can access it through *Start Menu | Programs | WeSay*.

You can also find WeSay in *My Computer | C:/ | Program Files | WeSay*.

In the Start Menu folder, you will find four icons. The first two, 'Configure Biatah Sample' and 'Run Biatah Sample' are example dictionaries included with WeSay that you can look at to see how a dictionary should be configured. The other two, 'WeSay' and 'WeSay Configuration Tool' are for your use in building a dictionary.

When you open the WeSay Configuration Tool for the first time, a pop-up like the one shown below will ask you to give usage statistics to the developers of WeSay. Read this and **decide whether or not to share your information**.



Once you've done this, the WeSay Configuration Tool will open and you will be ready to being building a dictionary.

HowTo Install WeSay on Linux

To install WeSay in Linux, you will need to work through the Synaptic Package Manager.

Getting the APT Lines

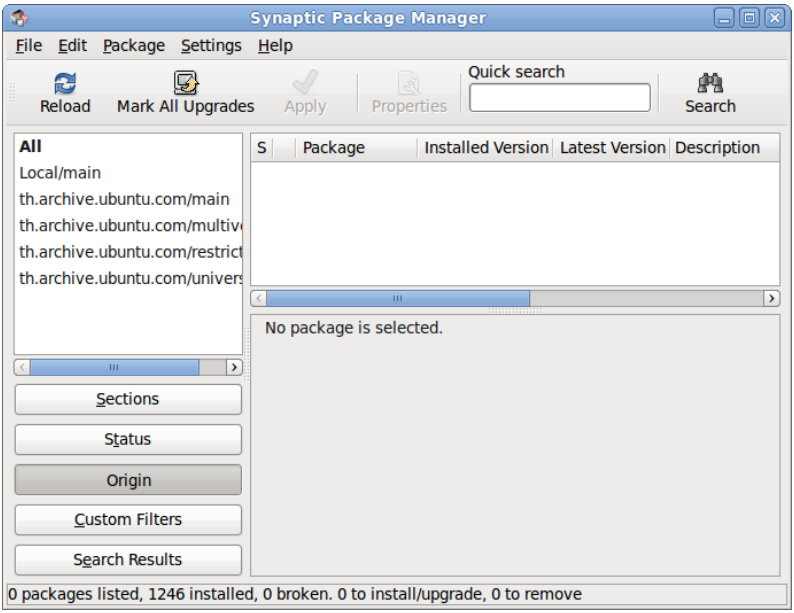
There are three APT lines you will need to add to the Synaptic Package Manager in Linux in order to have access to WeSay. The lines are different for each version of Linux – Hardy, Intrepid and Jaunty. You can **find the line specific to your version of Linux** at http://projects.palaso.org/wiki/wesay.

Or you can **use the APT lines here**, which are for Jaunty:

```
deb http://packages.sil.org/ubuntu jaunty main
deb http://packages.palaso.org/ubuntu jaunty main
deb-src http://packages.palaso.org/ubuntu jaunty main
```

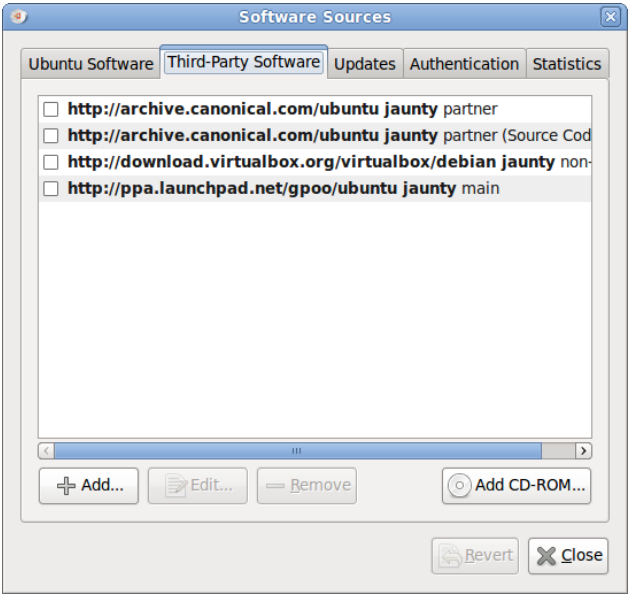
Installing the Repositories

The WeSay program is included in these repositories, so you will need to enable the repositories in order to install WeSay. To do this, start by **opening the Synaptic Package Manager**.

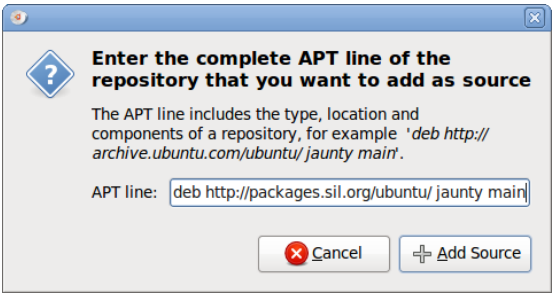


Click on 'Settings' on the menu bar and **go to 'Repositories'**, which will open a

Software Sources menu. In this menu, **click on the 'Third-Party Software' tab**, as shown below.



In this window, **click on the 'Add...' button**. This will open a dialogue asking you to input an APT line. **Type in one of the lines** listed above, as shown.

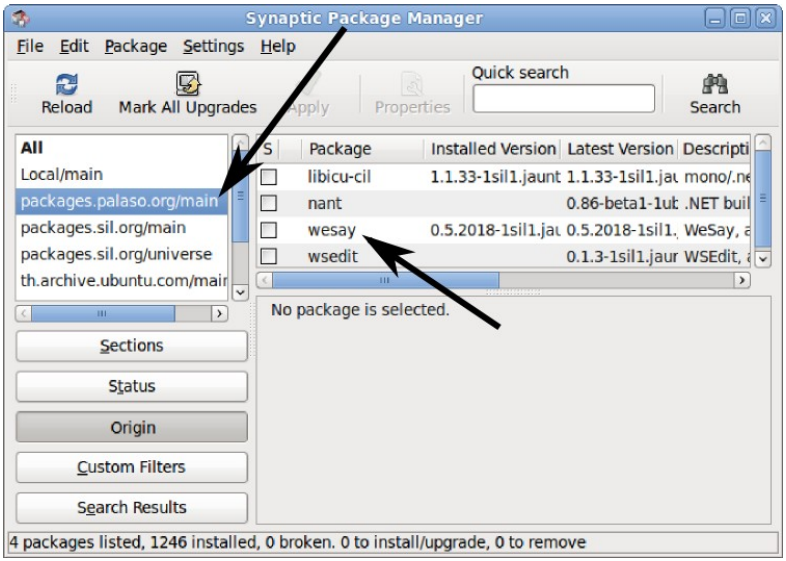


After you've typed in the APT line, **click 'Add Source'** to install the new repository Do this separately for each of the three lines listed.

Once you've added all three repositories, **press 'Close'** to exit the Software Sources menu and return to the Synaptic Package Manager. Your new repositories will still not be installed. You need to **click 'Reload'** in the top left of the window to update the new

repositories.

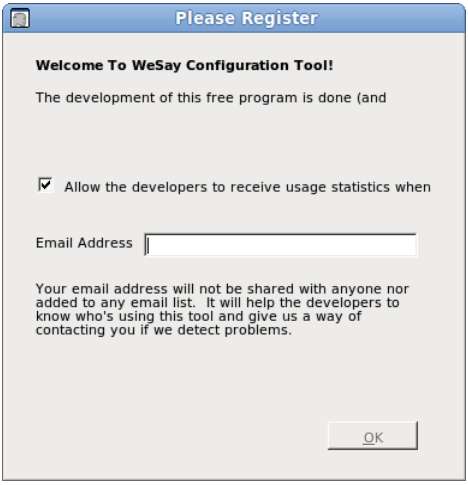
After you load the repositories, they will appear in the Synaptic Package Manager.



You can see the new repositories on the left, at the top of the list. Inside of the selected repository, you can also see the list of packages it contains. The important one is WeSay, indicated by an arrow. **Click on the check box** to choose to install WeSay, and then **click 'Apply**'. WeSay will automatically install itself and all of its necessary components.

After it's installed, you can find WeSay either in *Applications* | *Office* or in *File System* | *usr* | *bin*.

When you open the WeSay Configuration Tool for the first time, a pop-up like the one shown below will ask you to give usage statistics to the developers of WeSay. Read this and **decide whether or not to share your information**.



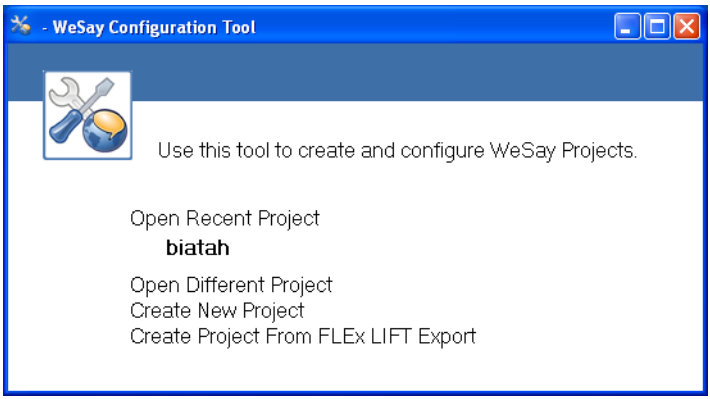
Once you've done this, the WeSay Configuration Tool will open and you will be ready to being building a dictionary.

HowTo Setup For a New Project

In order to begin building the Clickwhistle dictionary, we need to start a WeSay project. This section explains how to start a dictionary and get its writing systems initialized.

Creating A New Project

When the WeSay Configuration Tool first opens, you will see a window like the following:



The recent project displayed is the example dictionary 'biatah' included with WeSay. You can choose to either look at the 'biatah' dictionary for reference, work from a FLEx LIFT export, or create a new project. To set up a new dictionary, **choose 'Create New Project'**.

If you already have a dictionary made in FLEx, you should create your project from that. For more information about working between FLEx and WeSay, see the section 'HowTo Interact with FLEx'.

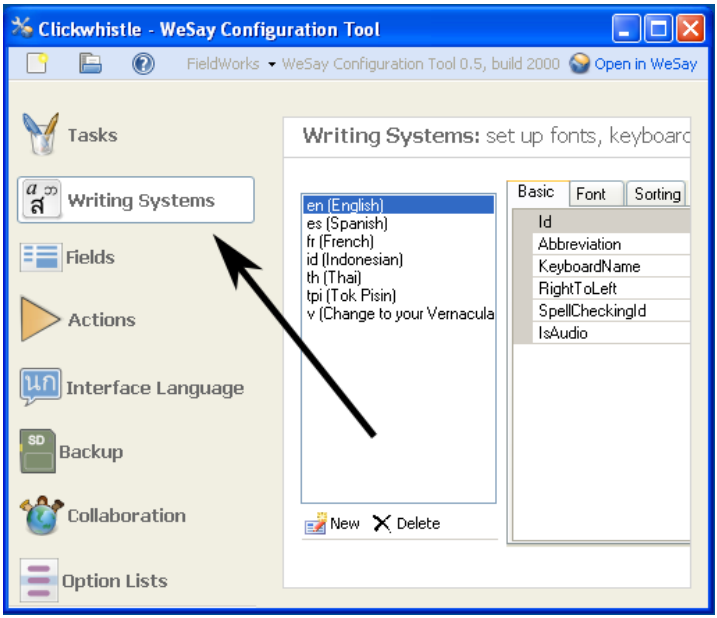
WeSay will then ask you for the name of your project. Typically this will be the name of the language you are making the dictionary for.

Once you've input the name of your project, the WeSay Configuration Tool will open for configuration and there will be a pop-up, explaining how to start using WeSay. This information will be covered later in this guide.

Your created dictionary will be stored in *My Documents | WeSay* after you make it.

Writing System Setup

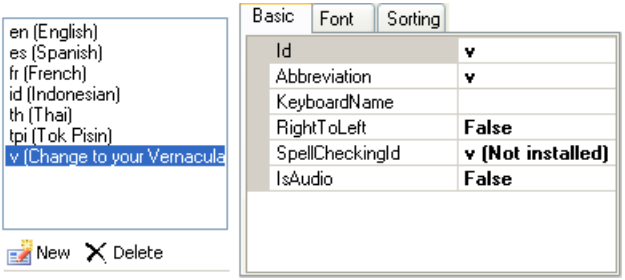
The first step to setting up the Configuration Tool is to choose the writing system of the language you are going to be working with. For this, **click Writing Systems** on the bar on the left of the window.



Since we are translating the Clickwhistle language, that is the name of our dictionary. It's also the writing system we need to set up.

In this menu, you can see seven possible writing systems, starting with the selected – English. At the bottom of this list is the vernacular writing system intended for you to use to make your own writing system to use in the dictionary. It's the one abbreviated 'v'. **Click on the 'v'** to start making your writing system.

The menu should then look like this:



The Id is the first line you need to change. **Type in the iso-code** (also known as the ethnologue code) for the language you will be making your dictionary for. Each language has a specific iso-code, known internationally, and it is important that you input the correct iso-code, not the name of the language or a non-standard abbreviation.

You can find a language's iso-code at http://www.ethnologue.com/country_index.asp. Click on the region of the world your language is in, and select your country to bring up a list of all writing systems in your region, along with their iso-codes.

Note that you won't be able to add the language name in parentheses like the writing systems already in WeSay. You can only add the iso-code.

Next you can **change the Abbreviation line**, which will modify how your writing system is displayed in WeSay. For instance, the iso-code for Zulu is 'zu', but you could set its abbreviation as 'zulu' which may be more meaningful.

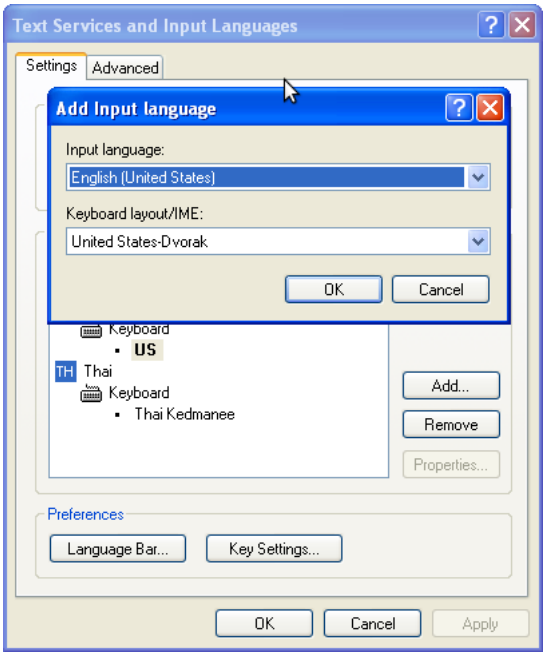
The SpellCheckingId line, a dropdown menu, is how you enter a spell checker for your new writing system. If you have an Open Office spell checker installed on your computer for the writing system you are entering, you can **activate that spell checker by selecting it in the dropdown menu**.

Once that is finished, **click on the line next to KeyboardName** and open the dropdown menu. If you have different languages installed on your computer, they will appear as selections in this menu. **Choose which language you are going to be using** for this writing system.

On Windows you can tell what languages you have on your computer by the language bar on the bottom right side of your screen or by clicking on the Keyman icon.

If the keyboard you need isn't already installed, there are a few different ways you can try to get it. The easiest way is to search through the Windows set of keyboards by

going to *Control Panel* | *Regional and Language Options* | *Languages* | *Details*. In this window, look for the keyboard you need. If it is not already present, click 'Add...' to open a dialogue like the following.



Now select the language and keyboard you would like to install, then press OK to install the keyboard. It will then be added to your language bar and list of keyboards. Select your new keyboard and press OK to close the menu.

Some languages, however, are not included with Windows and you will need to install them through other means. A good program to do this is Tavultesoft's Keyman. You can find a link to install this program, and a list of many Keyman keyboards, at http://scripts.sil.org/KeymanKeyboardLinks.

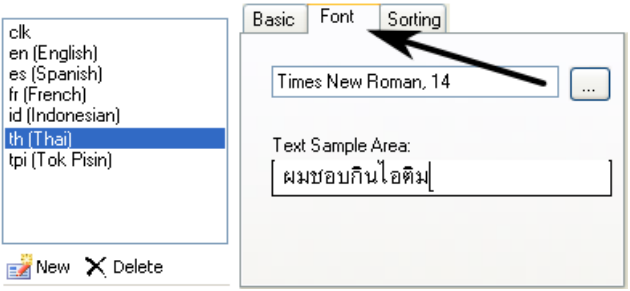
However, if your language is not there either, you can make a custom keyboard in the language you want by using Microsoft's Keyboard Layout Creator (MSKLC). More information about MSKLC, as well as the download for this program, is available at http://microsoft.com/downloads.

Once you have installed a new keyboard, you may need to restart the WeSay Configuration Tool.

After your keyboard is selected, **click on the RightToLeft line** and use the dropdown menu to choose which direction the language is read from. For now, as you first set up

a writing system, don't worry about the IsAudio line.

Now that the basics are set up, **click on the Font tab** to choose what font you want your writing system to display in.



Click on the button with the ellipses to choose the font parameters, and use the sample area to make sure the font displays clearly. The sample can display in any language that your computer supports.

Once you have a working font chosen, you can **click 'Open in WeSay'** in the top right corner of the WeSay Configuration Tool to begin working in your dictionary.

HowTo Interact with FLEx

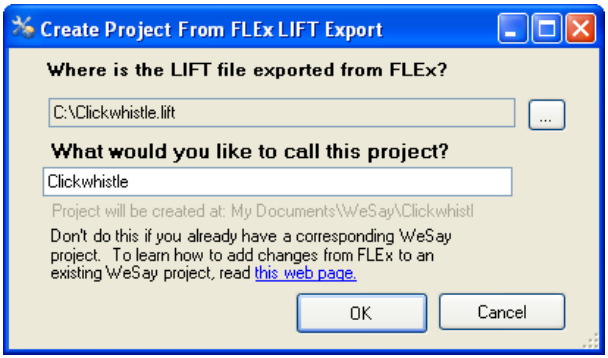
Importing LIFT Into WeSay

WeSay works with the LIFT file format, which is one of the export formats that the Fieldworks Language Explorer (FLEx) uses. This allows you to transfer your dictionary between FLEx and WeSay.

You can get detailed instructions for how to work between WeSay and FLEx at http://www.wesay.org/wiki/ShareWithFLEx.

FLEx's major export is LIFT, so if you already have a dictionary made in FLEx, you can easily **go to the export menu** and **save your dictionary as LIFT**.

Then, when you open WeSay, **select 'Create Project From FLEx LIFT Export**'. WeSay will open a window like the following.



Click on the button with the ellipses to find the LIFT file you want to bring into WeSay. You can name the dictionary something other than what it was called in FLEx, and then **press OK** to bring the dictionary into WeSay to work on.

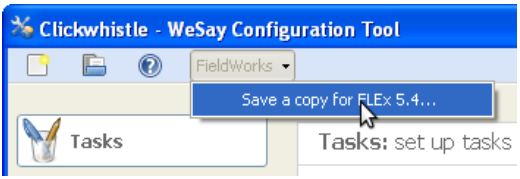
When you open a FLEx dictionary, WeSay will already have words entered, and it will already have the writing systems configured.

Don't change any information about the writing systems, or FLEx will create duplicates if it reads your data again.

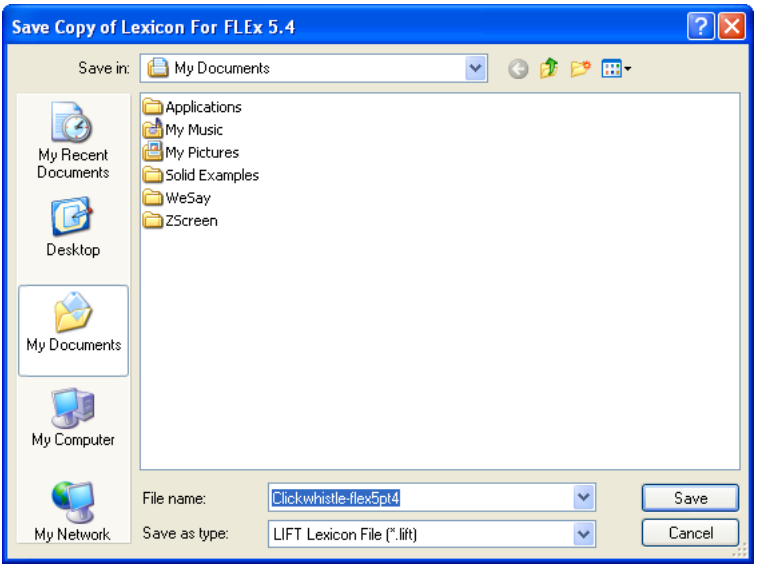
Exporting LIFT From WeSay

You can also use LIFT format to export a dictionary you've worked on with WeSay.

This can then be sent back to FLEx. To do this, open the WeSay Configuration Tool and **click on the FieldWorks button** in the upper left corner. Then **click the 'Save a copy for FLEx 5.4' button**, shown below.



This will open a dialogue which will ask you where to put your dictionary, and what to call it.



Input this information and press Save to export your dictionary. This will create a LIFT file that you can later use in FLEx.

HowTo Add an Additional Language

After we made a writing system for Clickwhistle, we were able to start a dictionary translating from Clickwhistle to English. However, if another VLE, Luigi, joined the team and wanted to translate into Italian as well, we would need to add another writing system.

Steps to Add Additional Languages

When we decide to add an additional writing system – in our case, from Clickwhistle into Italian – the second writing system of Italian will not automatically appear in WeSay. We will need to manually set it to appear.

There are two steps to entering a new writing system and getting it to show up in WeSay:

1.**Add the new writing system** in the Writing Systems menu in the WeSay Configuration Tool.

2.Get the new writing system to display by **configuring where it is shown** in the Fields menu.

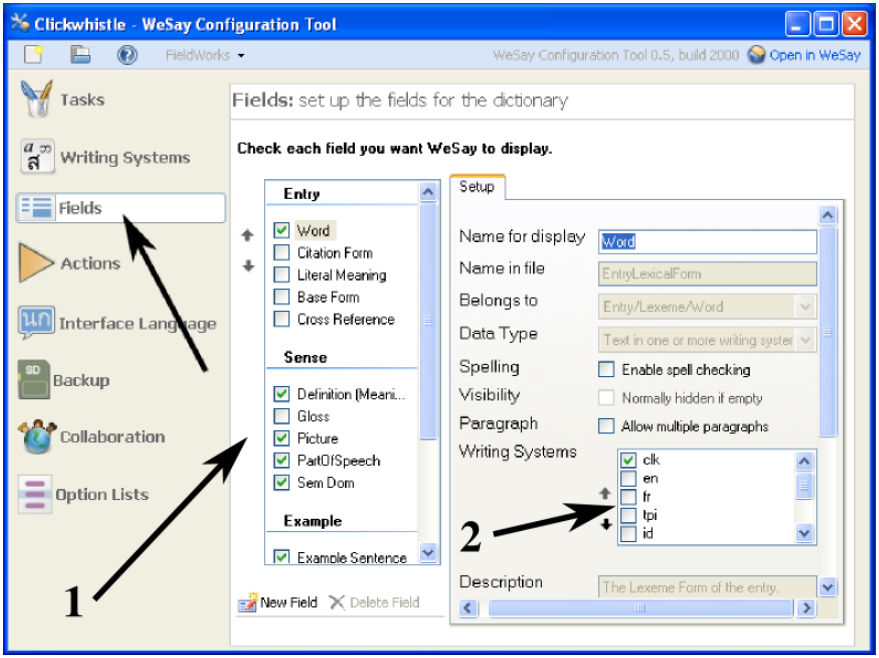
Start by **going to the Writing Systems menu** in the WeSay Configuration Tool. Then **click the 'New' button**, shown above. Now **fix the settings you want** for this new writing system (covered in Initial Setup).

Even though you just added this writing system, it still won't appear in WeSay in any of the text boxes. Now you have to configure the fields to display this writing system.

Setting Displayed Writing Systems

In order to set where your new language is displayed, you need to **open the Fields menu**, shown below.

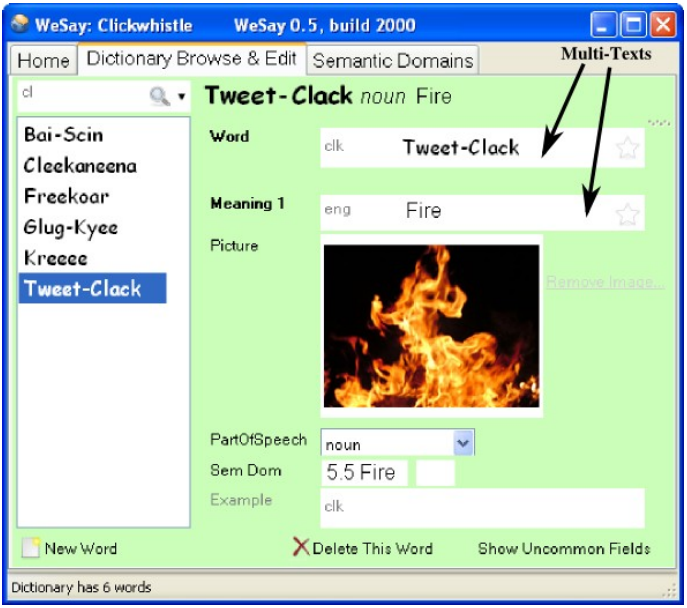
The two important features of this menu are the Checkbox Panel, labeled '1'; and the Writing Systems box, labeled '2'. You can click on any of the items in the Checkbox



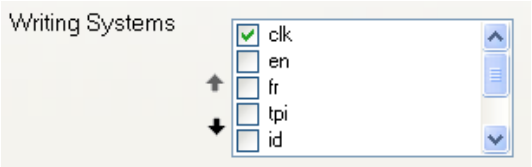
Panel to open its set of Writing Systems. Then you can **check off the writing systems you need** in the Writing Systems box.

You can do more with the Checkbox Panel. See the section 'Controlling Fields in WeSay' for a guide on this.

Whatever changes you make in the Fields menu will show up when you open WeSay and go to 'Dictionary Browse & Edit', shown below. Notice how in the multi-texts*, the languages we chose under Writing Systems appear.

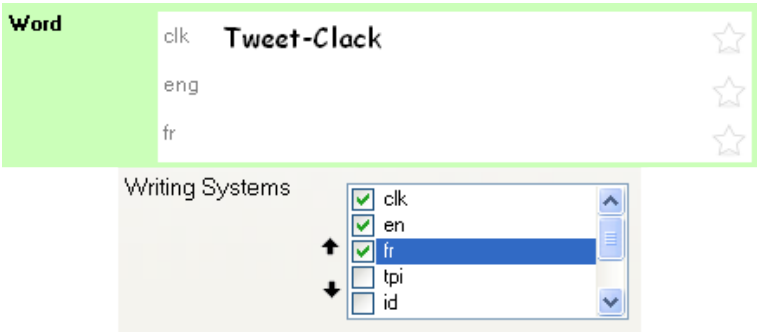


*A multi-text is WeSay's term for the text fields used to input the different information in WeSay. They are indicated in the screenshot above.



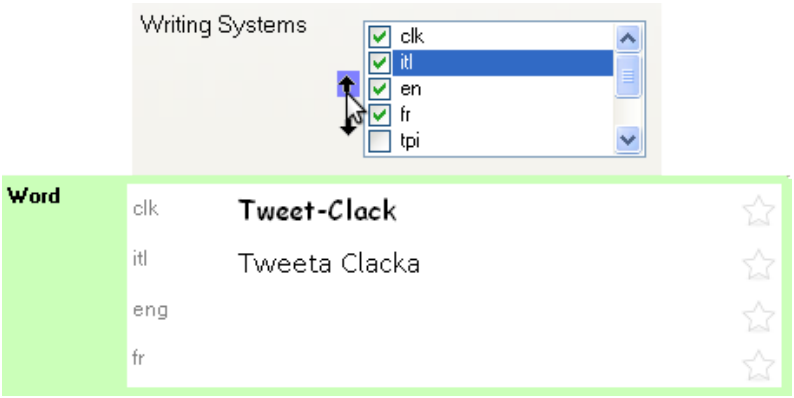
Whatever writing systems you **check or uncheck in the 'Writing Systems' box** will be displayed in the multi-texts in WeSay. This is the crucial part of the Fields menu. The writing systems checkboxes shown above have total control over what languages WeSay can use.

In the 'Word' field of our WeSay at the moment there is only Clickwhistle ('clk') being displayed. However, if we were to **check other boxes**, like 'en' and 'fr', these languages would become available in the multi-text and the multi-text would look like the picture below.

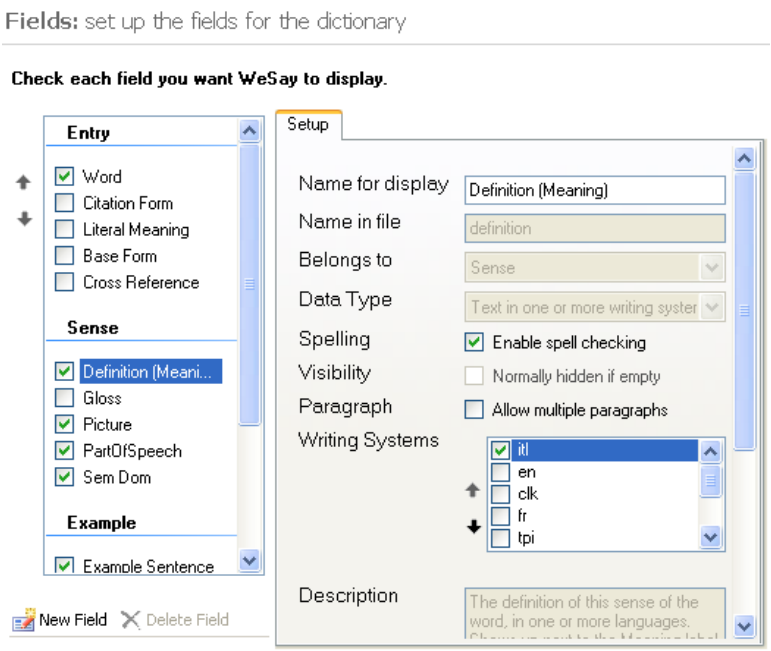


You can add these extra languages to the multi-texts of any category in your dictionary – Words, Notes, Definitions, etc. Just select different fields in the Checkbox Panel.

The new writing system we made (in our case, Italian) will always appear at the bottom of the list of Writing Systems. You can scroll down to find it, and **use the black arrows beside the list to move it up the list**.



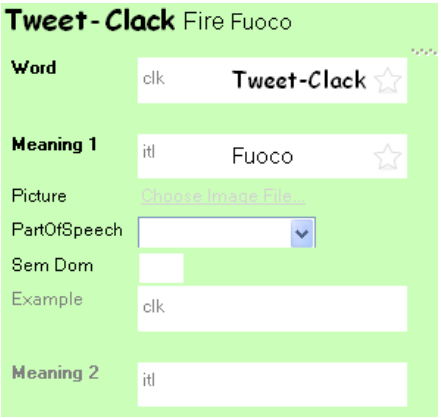
The different writing systems displayed in a multi-text will appear in the order they are shown in the selection box. Use the black arrows to put the writing systems you want at the top above the other systems.



Coming back to our original intention of translating from Clickwhistle to Italian, shown above is a way we could set up the Fields menu to give us the right results.

In this form, the only fields we've changed are the Word and its Definition. The Word we've made only in Clickwhistle (clk) and the Definition we've made only in Italian (itl).

Shown below are the resulting WeSay dictionary fields. Because we made the word in Clickwhistle and the definition in Italian, those are our only options for what to write under Word and Meaning.

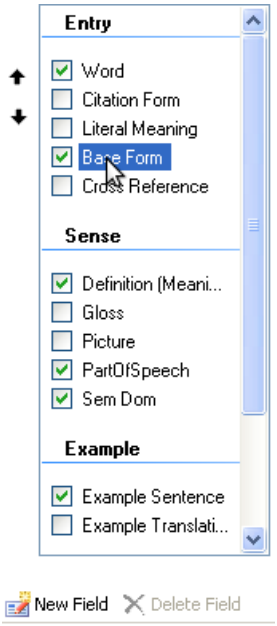


By using the Fields menu in the WeSay Configuration Tool, you can add multiple languages to WeSay and configure the dictionary to use exactly the languages you want to use.

HowTo Control Fields in WeSay

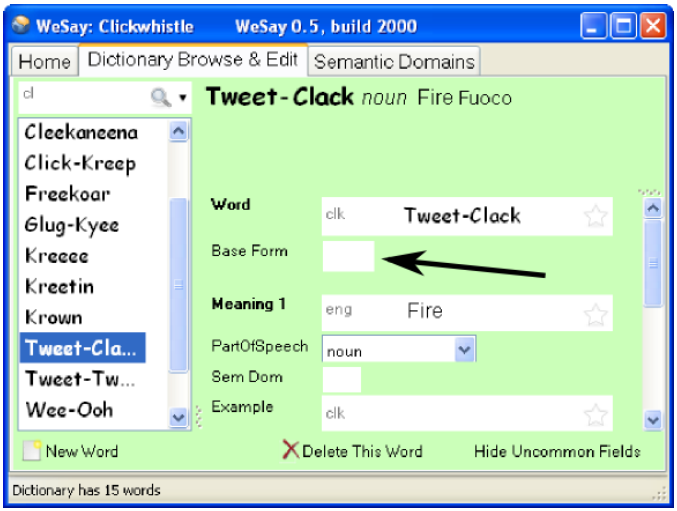
As you work on your project, you might find that the fields available in 'Dictionary Browse & Edit' aren't sufficient for your job. You might not want some fields, like Picture, and want other fields like Base Form.

You can control what entries will appear in the 'Dictionary Browse & Edit' menu in



WeSay by **using the Fields menu** in the Configuration Tool. Once in the Fields menu, **look at the Checkbox Panel** on the left of the screen.

Any of the checked boxes will show up in WeSay when you open it. For example, if you unchecked 'Picture' and checked 'Base Form', then when you opened WeSay again, there would be no line to add a picture, but there would be a new line for the base form.



You can see what fields you are showing by opening WeSay and going to 'Dictionary Browse & Edit', as shown above. Any changes you make in the Configuration Tool will be displayed in this dictionary window. Notice how, in our example, there is the Base Form field.

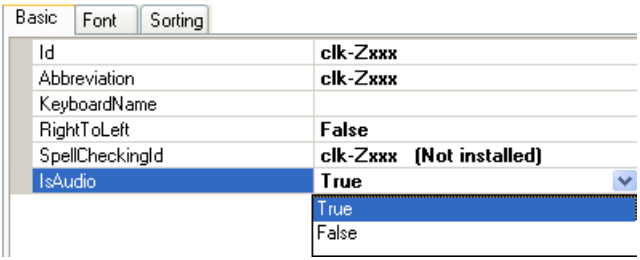
You can further change the lines in WeSay using the Writing Systems panel. See the section 'Adding an Additional Language' for information on this.

HowTo Record Audio

When you build your dictionary, you might find that it would be useful to have a VLE saying some of the words you are adding. In our case we would like a Clickwhistle native to say each of the Clickwhistle words we are adding. WeSay is already prepared to allow this.

Setting up an Audio Writing System

In order to add audio to your entries, the only thing you have to do is add a new audio writing system to your dictionary.

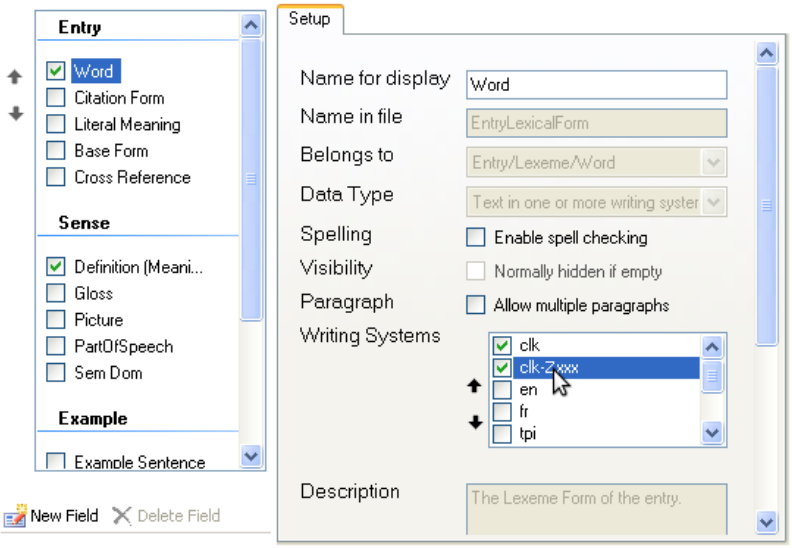


To make an audio writing system, **go to the 'Writing Systems' menu** in the WeSay Configuration Tool and go through the same process to **make a writing system** as you did before (see Initial Setup). This time, however, **set IsAudio to true**.

The iso-code for unwritten systems is -Zxxx. For example, if you were doing an audio English system, it would be en-Zxxx.

Once you have this audio writing system, **go to the Fields menu to set up WeSay for recording**. Like with the other writing systems, you can **put your audio writing system into any of the fields** in the checkbox panel, whether it's Word or Definition or Example Sentence.

This can be useful if you would like to record someone saying the word, as well as giving an example sentence. Later on you can also record the translation of the word, or even a translation of the example sentence.



In our case we've put the audio system of Clickwhistle alongside the regular writing system, and both of these are in the Word field. That means that in WeSay we can now transcribe a word, record someone saying it, and then write its definition.

Recording with WeSay

When we go into WeSay we'll see an extra field in the Word multi-text with a button.

This button is used to record sounds. If you have a microphone hooked up to your computer, when you **press and hold down on the button**, WeSay will record the sounds from the microphone.

The record function works like a walkie-talkie. As long as you hold down on the button, the microphone records. Once you finish talking, release the button. If you just tap the button without holding down, WeSay won't record.

After you've finished recording there will be two buttons, as shown below – a play button (the triangle) and a cancel button (the 'x').

Pressing the play button will play your recording back to you. **Pressing the cancel button** will remove the recording. If you do this, the record button will re-appear and you can make another recording.

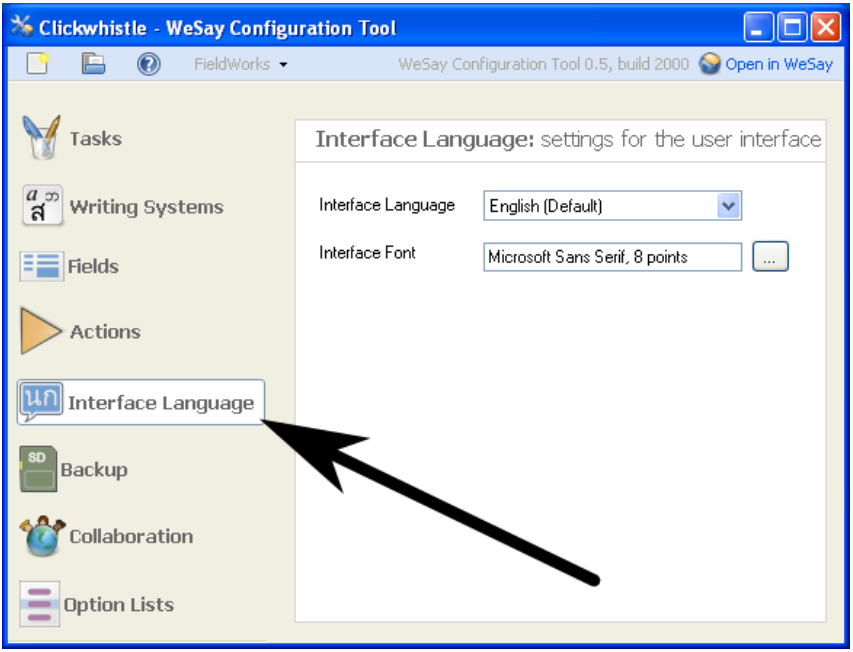
You can put as many audio writing systems as you need on every multi-text. And you can put each audio writing system on as many different multi-texts as you desire. All you need to do is make the audio writing systems and use the Fields menu to attach them to specific fields.

HowTo Change the Interface Language

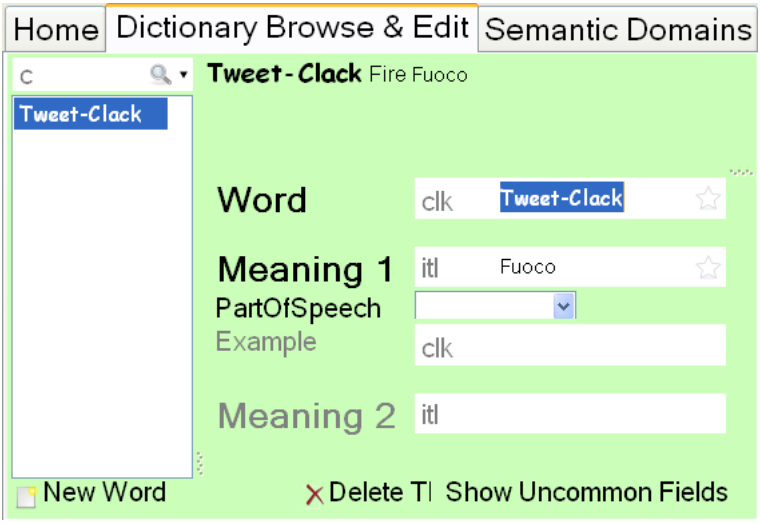
A Clickwhistle VLE in our project wanted to work with WeSay, but he wasn't able to read English. With the Configuration Tool, we can allow him to still use WeSay by changing the interface language.

Accessing User Interface Languages

To change the user interface language or font, **go to the Interface Language menu** in the WeSay Configuration Tool.



In this menu you can **choose what language WeSay will be in** using the 'Interface Language' dropdown list. You can also **select different fonts** in order to maximize the clarity of the languages you use, including choosing larger fonts to be easier to read as shown below.



However, notice that the changes in font size only apply to the menu words in WeSay, not the words entered into the dictionary.

The changes you make to the language will only affect WeSay and not the WeSay Configuration Tool.

HowTo Change WeSay Tasks

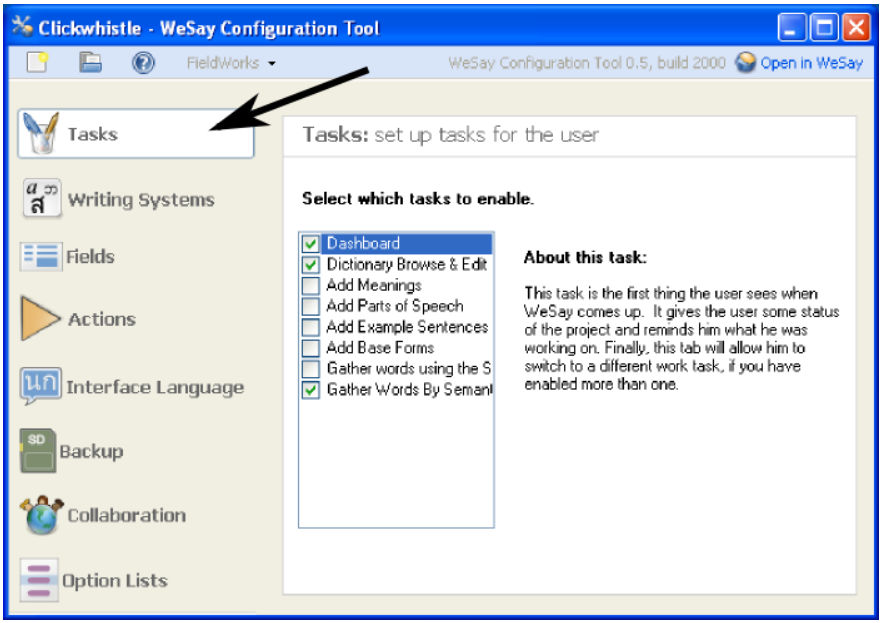
Now that you've gone through all of the previous steps in this documentation, your WeSay will be able to change tasks easily, as discussed in the 'About WeSay' section. For our Clickwhistle dictionary, we are now at the stage where we can change WeSay to only add words, or only add meanings, or any specific task.

This section will explain how you can change these tasks.

Tasks Menu

There are three tasks WeSay can do when you first open it – 'Semantic Domains', 'Dictionary Browse & Edit', and 'Backup'. However, there are seven other tasks that WeSay can undertake which you can enable in the WeSay Configuration Tool.

Five of these other tasks can be **found in the Tasks menu**, as shown below.



The first in the list, 'Dashboard', is the default menu you see when you open WeSay. The other two check-marked boxes are two of the buttons on the main screen in WeSay. The third option usually seen in WeSay, 'Backup', is found in the Actions menu and will be discussed shortly.

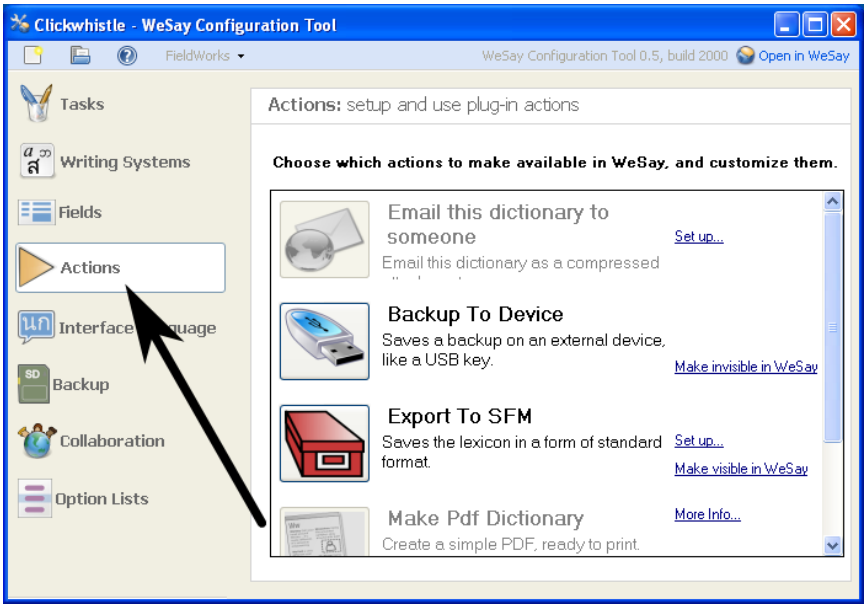
The other five unmarked boxes are possible options you can put in WeSay. If you **check any of the boxes**, then when you open your dictionary again they will be available on the opening screen. Likewise, if you **uncheck any of the current boxes**,

they will no longer appear in WeSay.

You can look through the individual tasks in the WeSay Configuration Tool to see what they are each for, or you can check the reference in this documentation for descriptions.

Actions Menu

The Tasks menu has five out of the seven possible options you can add to WeSay. The other two, and the already enabled 'Backup', can be found **in the Actions menu**, as seen below.



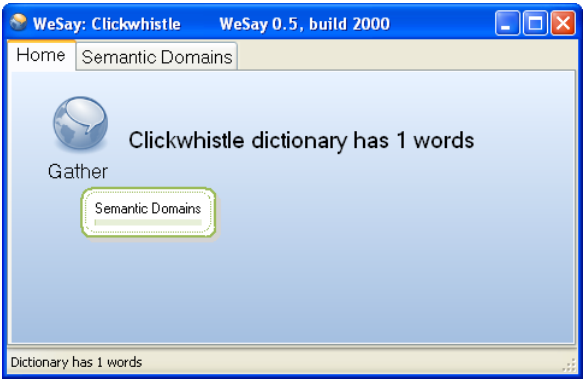
The Actions menu has several options. You can **configure three of these to appear** in WeSay – the Emailing, Backup, and Export options. Two other options are also available: Making a PDF and using Lexique Pro. All of these options in the Actions menu will require time to set up, whereas the Tasks menu has options that work as soon as you select them.

Explanations for the selections in the Actions Menu are also available in the reference. You can also get information by clicking next to each option on either the 'Set up...' or 'More Info...' buttons.

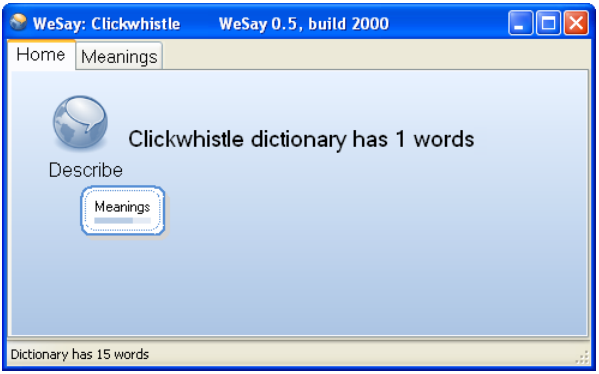
Uses for Tasks

This is the most important part for working in a group project. As discussed in the section 'About WeSay', you might at some point want to build a dictionary by having different people contributing different, discrete parts. By focusing what WeSay accomplishes, you can avoid overwhelming these people with too many buttons and options.

For instance, in a project we could start by only adding in words. We might want to use the Semantic Domain to give us prompts, so we will disable everything except the Semantic Domain. WeSay would then only have one button to press.



We could then give a computer with this WeSay to someone and have them go through and add in words through the Semantic Domain. Once we had enough words, we might want to put in the meanings of these words and so we could disable the Semantic Domain and turn on the Add Meanings function.

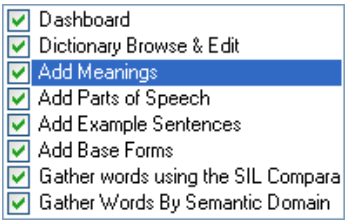


The next person to use WeSay would then only have the ability to add in meanings to the words in the dictionary. We could continue focusing the abilities of WeSay in this manner until we had made a dictionary and filled in all the entries.

Reference

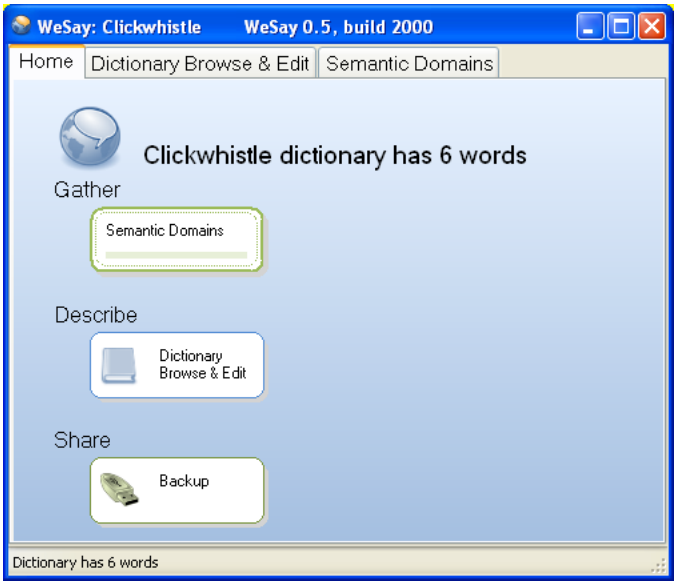
Tasks

There are eight total tasks that can be selected in WeSay, shown below. They are explained in the Configuration Tool, but you can read explanations here as well.



Dashboard

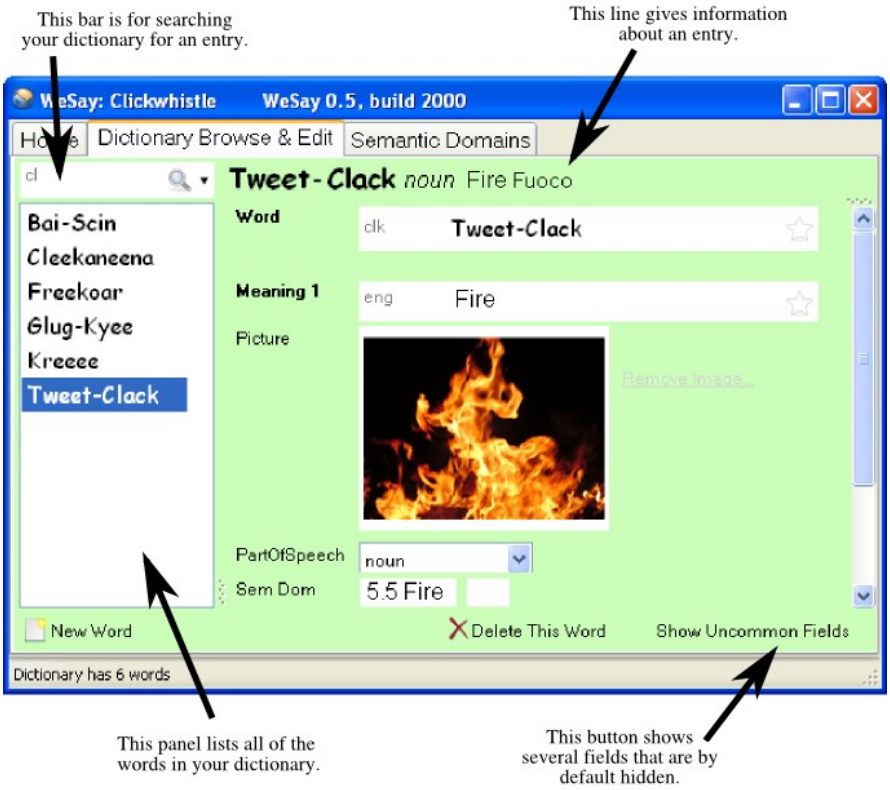
The dashboard is the first menu that appears when WeSay is opened, which means you can't turn it off.



It is the point of access from which you can select any of the other options you have configured WeSay to use. It also tells you the name of your dictionary and the number of words you have collected.

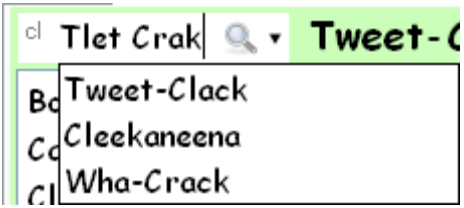
Dictionary Browse & Edit

This tool is used to search through all of your dictionary entries to find certain words and edit any incompletions.



There are the basic options 'New Word' and 'Delete This Word' for adding to or deleting part of your dictionary. The 'Show Uncommon Fields' button reveals several otherwise hidden fields, which don't appear so that the page doesn't get cluttered. You can change what fields are hidden in the Fields menu in the Configuration Tool.

The search function isn't used to find the exact word you are searching for. You only need to type in a word similar to the one you are looking for, and the search function will automatically bring up a list of close approximations, as seen below.



Add Meanings

The 'Add Meanings' tool allows you to scroll through your entire dictionary and add definitions to any words that lack them.

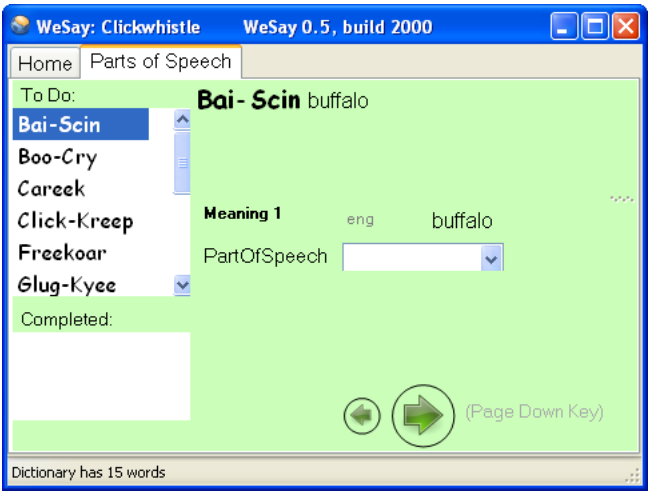
Type the meaning in the 'Meaning 1' box, and then either click the arrow or press Page Down to continue to the next entry. You can also use Page Up to go to the previous entry, and press Enter to add the meaning. As you finish adding definitions, they will appear in the Completed panel.

You can also use the WeSay Configuration Tool to set this function to only add definitions to words that lack definitions in specific languages, or already have definitions in certain languages. For example, you could have have WeSay only do meanings for words that lack definitions in Italian, even if they don't have English.

You can access these settings in the Tasks menu, by clicking on the Meaning line.

Add Parts of Speech

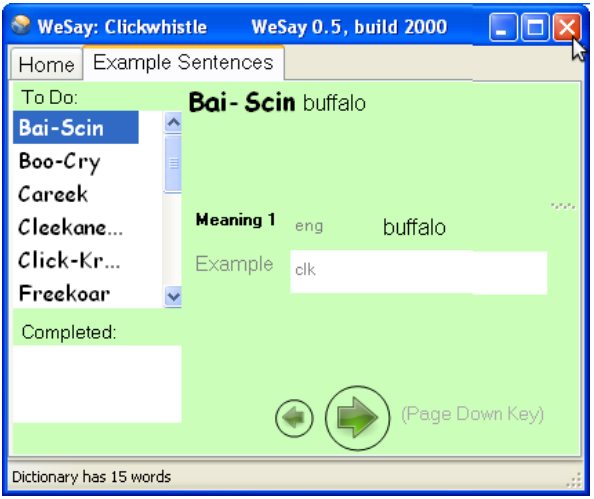
This tool enables you to go through a list of your dictionary words that lack identifying parts of speech and add them in.



Use either the green arrows, or Page Up and Page Down to scroll through the words and use the dropdown menu to select the corresponding part of speech. The words you have chosen Parts of Speech for will appear in the Completed box.

Add Example Sentences

The 'Add Example Sentences' function allows you to scroll through all of your dictionary entries that lack example sentences and add them in.

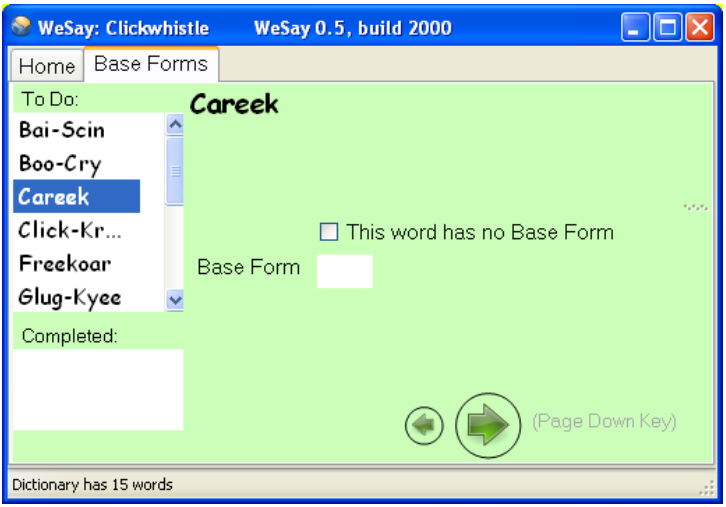


You can use the WeSay Configuration Tool to make the 'Add Example Sentences' menu sort through the example sentences by writing systems present or absent. For instance, you could have WeSay only do example sentences for entries written in Italian, rather than Clickwhistle. You can also use the Configuration Tool to set whether or not the example sentence will include a translation.

Use Page Up and Page Down, or the arrows, to scroll through the entries, and press Enter to add an example sentence to the dictionary. When you add the entries, they will appear in the Completed panel.

Add Base Forms

The 'Add Base Forms' tool is a means to go through all dictionary entries that don't have base forms entered.

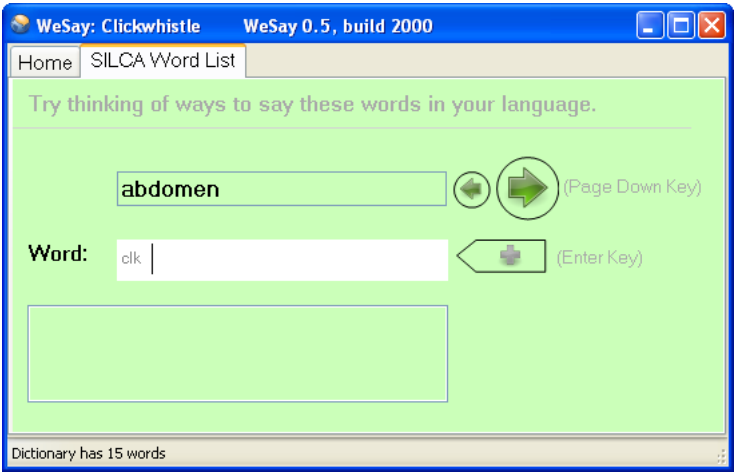


You can either add a base form or identify the word as lacking a base form. This tool uses either the green arrows, or Page Up and Page Down to scroll through the words and Enter to add the base form. Check the box if there is no base form for the word you are editing.

Gather Words Using the SIL Comparative African Wordlist

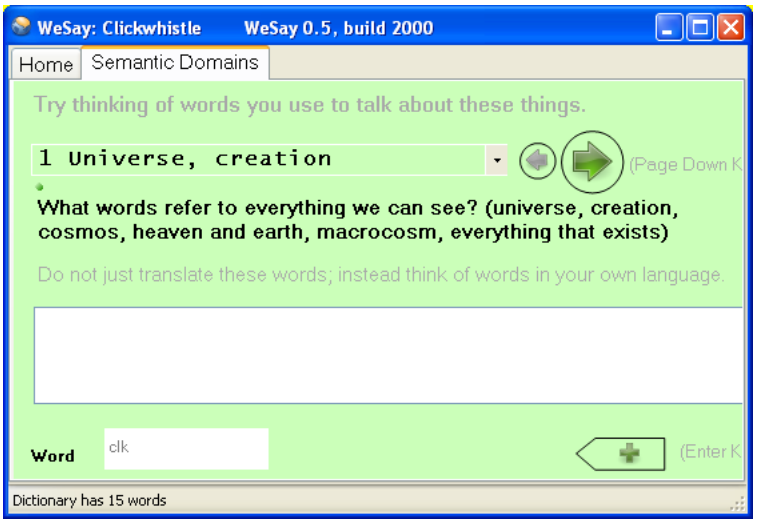
This tool is a collection of 1700 English words gathered by SIL that you can scan through. When you find a word you know in the language you are working on, you can add it.

Use Page Up and Page Down, or the two arrows, to scroll through the words; type the word and press Enter to add a word. It will appear in the empty box. You can add multiple entries for each word in the database. Later on you can go through and put in more information for these words separately.



Gather Words by Semantic Domain

This tool allows you to go through a list of Semantic Domain definitions and input words in another language.



The new words that you write will be tagged with the Semantic Domain number they were identified by. You can also go to the WeSay Configuration Tool and choose to tag the words with a short definition.

The definitions in the Semantic Domain include brief questions to help define the

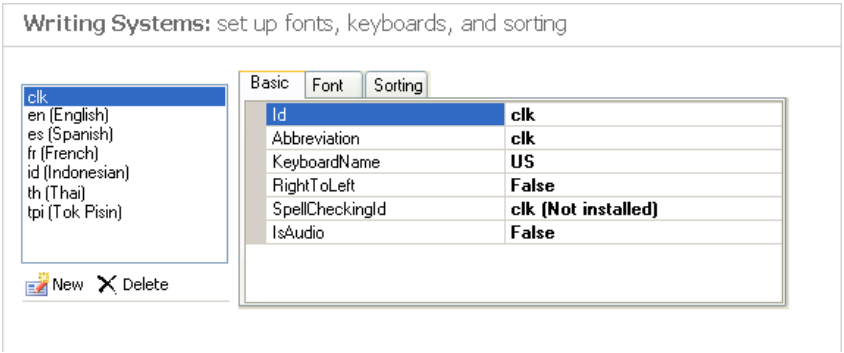
word. Use Page Up and Page Down, or press the arrows, to scroll through the questions in the Semantic Domain. Or, if you want to look through the options faster, use the dropdown menu and look for the number you want. Use Enter to input words to your dictionary, or press the button with the arrow.

Later on you can go to 'Dictionary Browse & Edit' and input Semantic Domain numbers separately. In that menu, you can search the Semantic Domains by keyword.

The Semantic Domain is capable of using different languages, but you'll need our help to do it. For more information on changing its language, see the help page http://www.wesay.org/wiki/How_to_customize_the_semantic_domains.

Writing Systems

In the Writing Systems menu of the WeSay Configuration Tool, you can either add a new writing system, or modify or delete an existing one. There are three tabs you can use to modify your writing systems, 'Basic', 'Font' and 'Sorting', as seen below.



Most of the Writing Systems menu information is explained in greater detail in the 'Initial Setup' section, under 'Writing System Setup'.

Basic

This tab allows you to enter the iso-code (also known as an ethnologue code) for your new writing system, as well as set an abbreviation of your writing system to use in WeSay. You can further select a keyboard to use with your new writing system, as well

as set a spell checker and choose the direction your writing system will read. The Basic tab also allows you to choose whether your writing system is audio or not.

Font

You can use this tab to change the font and size of your writing system. There is also a sample text area available for you to test the font you are choosing in order to make sure the font is clear.

Sorting

If the language you are working on has an alphabet with a peculiar ordering of letters, you can set WeSay to recognize this in the Sorting tab.

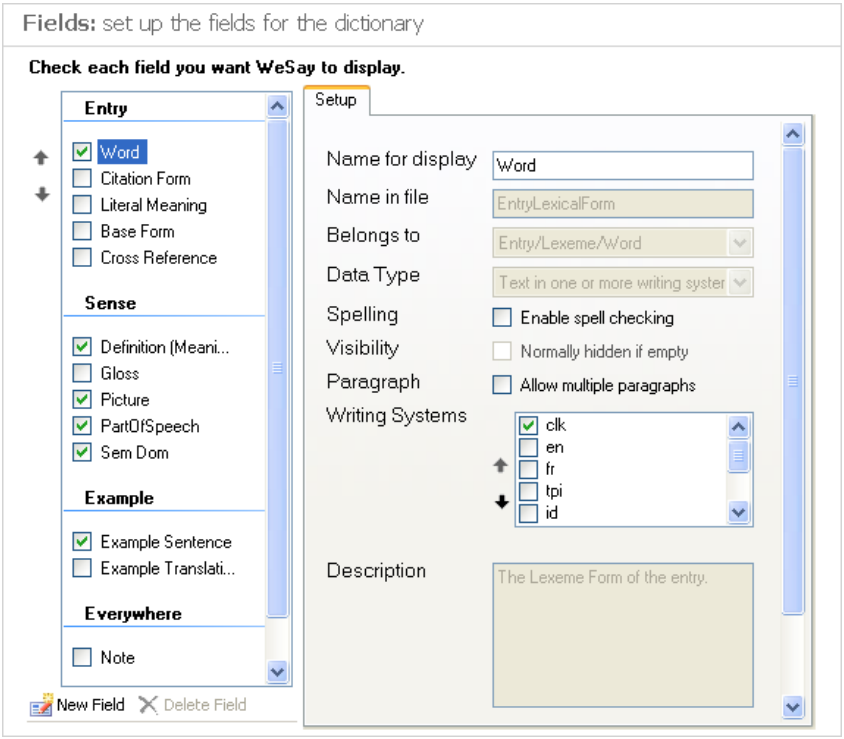
Use the dropdown menu to select a system to sort your language. If none of them fit, you can use either of the two custom sort methods at the top of the list to set up your letter order. Type different words into the box on the right to test your alphabet sorting order.

You can find more information about using the Sorting function at http://www.wesay.org/wiki/How_to_sort_using_a_custom_sort_sequence.

Fields

The Fields menu in the WeSay Configuration Tool, shown below, controls what information will be shown in WeSay. You can change what multi-text boxes will be shown (for example, the Meaning multi-text or the Example Sentence multi-text).

Or you can change what writing systems will be displayed within each individual multi-text (for example, add English to the Word multi-text or Spanish to the Meaning multi-text).



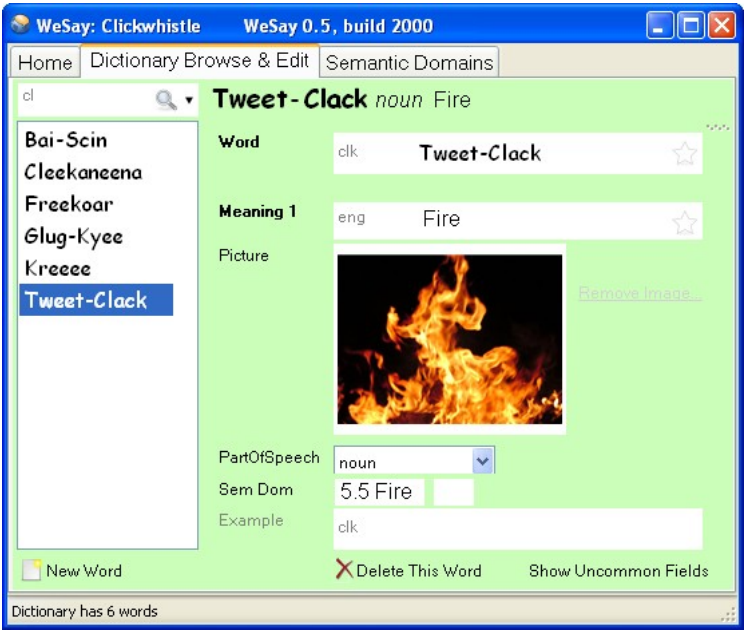
You can find more detailed information about the Fields menu in the 'Adding an Additional Language' section, under 'Setting Displayed Writing Systems'.

The Checkbox Panel on the left of the screen controls what multi-texts appear in WeSay. You can check or uncheck the boxes or you can press 'New Field' at the bottom to change which multi-texts are set to appear in WeSay.

The list of checkboxes on the right called Writing Systems controls what languages appear in each multi-text. Click on different fields in the Checkbox Panel and use the Writing Systems box to individually set what writing systems you want to appear in the multi-texts in WeSay.

You can go to WeSay and open 'Dictionary Browse & Edit' to see what your choices in Fields have changed.

In the Fields menu in the Configuration Tool, you can also use the other lines under the Setup tab to change information about different fields.

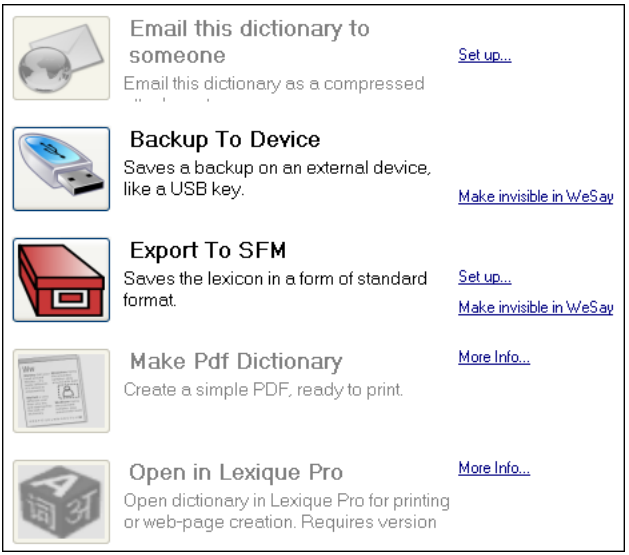


For the fields that came with WeSay, like Word, Definition or Part of Speech, you won't be able to edit most of their information. However, if you add your own field you will be able to change all of the information..

Actions

There are five possible actions to select in the WeSay Configuration Tool, shown below. All of these actions require some setup and are for use in storing, sharing or processing your dictionary, rather than building more entries. The Configuration Tool features some explanations of these functions, but you can read explanations here as well.

After you have set up the tools, you can use their functions either by opening WeSay and using the buttons there, or by clicking the icons in the WeSay Configuration Tool.



Email This Dictionary

You can use this tool to email your entire dictionary to someone else, provided you have a working email address. Click 'Set up...' to enter the recipient's email address, and then you will have the option to make the button visible in WeSay. Once the button is in WeSay, you can click it to send your dictionary. WeSay will ask you for your name, email address and then information about your email server. After you enter this information, you will be able to send your dictionary through email.

Backup To Device

This tool will back up your entire dictionary onto an external device, provided you have one plugged into your computer. You can make the button for this visible or invisible in WeSay. When it's visible, if you have a device like a thumbdrive plugged into your computer, you can press the button to back up your dictionary.

Export to SFM

This is a wayof saving your dictionary in a standard form that was previously used in Toolbox (for more info, see http://www.wesay.org/wiki/Notes_on_Export_To_SFM). If you click 'Set up...' you can change the languages that this export will use, as well as entering exceptions and rules. You can also toggle the visibility of this button in WeSay. When the button is in WeSay, if you click it a text document will pop up containing your entire dictionary in SFM format, with the appropriate tags in place.

Make PDF Dictionary

This option converts your dictionary to a PDF, which makes it easier to print. To do this you will need the PrinceXml engine (for more information visit

www.princexml.com or www.wesay.org/wiki/MakePdfDictionary). You can use PrinceXml free unless you are intending to use your dictionary for commercial use, in which case you will need to purchase a license. Note that this feature is not for final publication, but is intended as an easy way to print out your dictionary as you are building it.

Open in Lexique Pro

This tool is a way to set up your dictionary for the internet or for print. It is a completely free program but you will need to download version 3 or later from www.lexiquepro.com/test.htm. Lexique Pro is for Windows only, and its button will become available in WeSay after Lexique Pro is downloaded. For complete instructions, go to http://www.wesay.org/wiki/WeSayWithLexiquePro.