DictionaryManager

Used for building new spell checking dictionaries.

The general strategy for building the dictionaries are to assemble collections of words from various dictionaries and lists.

These lists then are combined, dropping the duplicates.

Lists are loaded to Adsdi.DCT.Dct (dictionaries) as Adsdi.DCT.Entry (entry objects). These loaded dictionaries can be “output” to Adsdi dictionary files, which are simple pipe delimited files with five fields as follows:

auditions|4||T|T

Baguio|5|M|T|T

calorimetric|1||T|T

Corneille|5|M|T|T

electrodynamometer|1||T|T

excrescence|1||T|T

guidepost|4||T|T

instrumentalists|4||T|T

Loamy|S||T|T

penalize|1||T|T

1. ***Text*** – this is the word itself (String)
2. ***Source*** – indicates which numbered source list the word came from (String)
3. ***Flags*** – zero to many character flags indicating something about the word (from a particular source) (String)
4. ***IsSpellingChecked*** – indicates whether the word has been spell checked (Bool)
5. ***SpellingIsValid*** – indicates whether the word is spelled correctly based on an external spell checker (Word used) (Bool)

The example file above is “randomized” meaning that its “sort order” is random based on the use of random numbers (within a given random number between 1000 and 9999) the words will sort alphabetically. This optimizes the balancing of the TriNode used for the actual runtime structure used by the spell checker.

The general process to assemble dictionary is as follows:

1. Collect a set of word source lists into files that the program will read. Currently the lists used are as follows:
   1. 1-Unabr\_dict.txt
   2. 2-Family-Names.txt
   3. 3-Given-Names.txt
   4. 4-winedt.txt
   5. 5-en\_US.dic.txt
   6. 6-en-common.txt

It is “handy” to be able to refer to these files by their numbers. Using this naming convention, the number becomes the “Source” in the Entry object as described above.

1. Run successive steps of combining files by loading both “A” and “B” dictionaries. The resulting file from a combination is written to the default output directory as a time-stamped txt file. This file becomes the “base” file (normally loaded to dictionary “A” ) for combining the next file into. For example:
   1. Combine file 2 into file 1 by selecting the files in the “Dictionary B” and “Dictionary A” respectively and loading both dictionaries into memory. Then use the “Combine” button to combine the data and drop duplicates. The output file is written out with a timestamp such as “20130227-142936.txt”. This file becomes the “base” file to combine other files into.
   2. Load the combined file from the first step (for example “20130227-142936.txt”) as Dictionary A and then load the “3” file as Dictionary B. Combine these files which will result in a new time-stamp named file.
   3. Continue to combine the “4”, “5” and “6” files into successive time-stamp named files as output.
2. Run the final combined file through the Word Spell checker by loading it into Dictionary A and clicking the