



MagicSpeller for REALbasic

MagicSpeller for REALbasic is a collection of a REALbasic objects for use within the REALbasic development environment. Using MagicSpeller, a developer can easily add a full-featured spell checker into their project in just a few minutes. The system is customizable and cross-platform; running on Macintosh, Windows, and LINUX builds of REALbasic projects.*

MagicSpeller for REALbasic consists of three separate components:

- *MagicSpeller.rbo*: A REALbasic module that contains the essential methods and properties of the spell check system.
- *MagicSpellerInterfaceWnd.rbw*: A pre-constructed REALbasic window object that provides a default interface for the spell check system.
- *MagicSpellerTA.rbo*: An optional REALbasic class that extends the basic TextArea class to allow for inline (as you type) spell checking.

System Requirements:

- Macintosh OS 10.x; Windows 2000, XP, or Vista; x86-based Linux distributions with GTK+ 2.8
- REALbasic version 2008r2 or higher

* Technical support for LINUX builds is not available at this time, but many users have reported successful use of MagicSpeller in their LINUX projects.

Using MagicSpeller – Basic Steps

1. Drag the *MagicSpeller* module into the REALbasic project you are developing.
2. Drag the *MagicSpellerInterfaceWnd* object into the REALbasic project you are developing.
3. Install one of the *MagicSpeller* dictionaries onto the computer you are using and create a folderitem property that will point to the installed dictionary.
4. Before using the spell checker, two methods must be called. (See the method section later in this manual for complete description of the method and parameters)
 - First, if you are a registered user, register the application by calling the *MagicSpeller_register()* method. If this method is not called with the correct parameters, the MagicSpeller system will run in demo mode.
 - Second, prepare the spell check system by calling the *MagicSpeller_initialize()* method.
5. Add any TextArea that you want to spell check into the array by calling *MagicSpeller_fieldsToCheck.append(TextArea)*.

6. Call the method *MagicSpeller_checkSpelling()*. This will open the spell check interface and start the spell check for the TextAreas added to the *MagicSpeller_fieldsToCheck* array.

Adding Basic Inline Spell Checking (spell check as you type)

1. Follow steps 1, 3, and 4 as above.
2. Drag the *MagicSpellerTA* object into the REALbasic project you are developing.
3. Select the TextArea that you would like to have inline spell checking and set its *super* property to *MagicSpellerTA* in the REALbasic properties window.
4. Make sure the *styled* property of the TextArea is *true* in the REALbasic properties window.
5. Get outside and treat yourself, you're done!

MagicSpeller Methods

Global Methods:

These methods form the basis of the MagicSpeller system. They may be called by any class or object within the REALbasic document.

MagicSpeller_checkSpelling()

Called when you want to invoke the MagicSpeller default spell check dialog window and start the spell checker. This method will spell check any TextArea control that has been added to the *MagicSpeller_fieldsToCheck()* array as well as any strings that have been added to the *MagicSpeller_stringsToCheck()* array. TextArea controls will automatically be updated while strings will be placed into the *MagicSpeller_checkedStrings()* array in the same order as they were in the *MagicSpeller_stringsToCheck()* array.

MagicSpeller_hiliteErrors()

Hilites any misspelled words found in the TextArea controls it searches; which may be useful should a user paste some text into an TextArea. *MagicSpeller_hiliteErrors* searches each TextArea control in the *MagicSpeller_fieldsToCheck()* array and sets the color of any misspelled word that it finds to the hilite color. TextArea controls must have their *styled* property set to true to make use of this method.

This method searches through the entire text of each TextArea control without breaking so a large amount of text may take some time. As such, it should be used at one's discretion and thread as needed.

MagicSpeller_initialize(dataBaseLocation as folderItem)

Should be called before using MagicSpeller for the first time in an application. *dataBaseLocation* is a folderitem whose path points to the *MagicSpeller* dictionary that you want to use on the system. If the property *MagicSpellerDB* is equal to *nil*, then this method has failed and the spell check system is not available for use.

MagicSpeller_register(userName As string, userSerial As string)

Allows MagicSpeller to run in registered mode, thereby eliminating the restrictions of the demo mode. *userName* is the full name used when purchasing MagicSpeller and *userSerial* is the serial number that was provided.

Public Methods:

These methods are optional routines that may be of use to developers who want to provide their own interface for the spell check system. The methods may be called by any class or object within the REALbasic document by referencing them in the *object.method* manner (e.g., *MagicSpeller.getThisWord*)

getThisWord()

Selects the word that the insertion point of the TextArea control is currently in.

wordSpelledWrong(wordToCheck as string, provideSuggestions as boolean) as boolean

Checks to see if a single word is misspelled. It will return *true* if the supplied word is not found in the current dictionary. *wordToCheck* is the word to be searched for in the current dictionary. If *provideSuggestions* is set to true then MagicSpeller will populate the *MagicSpeller_wordSuggestions()* array with suggestions for the misspelled word.

MagicSpeller Properties

Global Properties:

These properties may be called by any class or object within the REALbasic document.

MagicSpeller_checkedStrings() as string

A string array that is populated by the *MagicSpeller_checkSpelling()* method. Strings from this array can be selected by the developer and used to replace the original strings after they have been spell checked.

MagicSpeller_fieldsToCheck() as TextArea

A string array that should be populated with any TextArea controls you want spell checked before calling the *MagicSpeller_checkSpelling()* method. If this array is empty, MagicSpeller will only check strings provided by *MagicSpeller_stringsToCheck()*.

MagicSpeller_hiliteColor as color

The color that MagicSpeller uses to display misspelled words in a styled TextArea control. This property is optional and defaults to red (RGB(255,0,0)).

MagicSpeller_maxSuggestions as integer

The maximum number of suggestions that MagicSpeller will return for a misspelled word. This property is optional and defaults to 5.

MagicSpeller_stringsToCheck() as string

A string array that should be populated with any strings you want spell checked before calling the *MagicSpeller_checkSpelling()* method. If this array is empty, MagicSpeller will only check TextArea controls provided by *MagicSpeller_fieldsToCheck()*.

MagicSpeller_textColor as color

The default text color in a styled TextArea control. Used to disable visual hilighting after a word has been corrected. This property is optional and defaults to black (RGB(0,0,0)).

MagicSpeller_wordSuggestions() as string

A string array that contains the words that MagicSpeller suggests as replacements for a misspelled word. This array is constrained by the *MagicSpeller_maxSuggestions* property and is populated by the *wordSpelledWrong()* method.

MagicSpellerDB

Can be checked to see if MagicSpeller has been successfully initialized. Should *MagicSpellerDB = nil* then MagicSpeller is not ready for use.

MagicSpellerUserDictLoc As folderItem

The location of the MagicSpeller user dictionary. This file is used to store the learned words of the user. This property is optional and defaults to the folder that contains the MagicSpeller dictionary currently in use.

Public Properties:

These properties may be called by any class or object within the REALbasic document by referencing them in the *object.method* manner.

gAudibleFeedback As boolean

Plays the current system sound when a misspelled word is typed in a *MagicSpellerTA*. This property defaults to false.

gDisableInlineChecking as boolean

Disables the display of misspelled words in a *MagicSpellerTA*. Typed words will still be checked and contextual menu clicks will still provide suggestions. This can be combined with the *gAudibleFeedback* property should you want to have only the system sound as an indicator of a misspelled word. This property defaults to false.

windowX as integer

An integer variable that contains the horizontal, left-side position of the default spell check window. This property can be set before the spell check window is opened to allow for consistent positioning. The default spell check window automatically sets this property in its *close* event.

windowY as integer

An integer variable that contains the vertical, top position of the default spell check window. This property can be set before the spell check window is opened to allow for consistent positioning. The default spell check window automatically sets this property in its *close* event.

MagicSpeller Dictionaries

All dictionaries for MagicSpeller are based on word list projects and are freely distributable. The

two dictionaries based on the SCOWL project are copyright 2000-2004 by Kevin Atkinson and are based on the version 6 release. The *MagicSpeller SCOWL-6 base dict* is intended to be the default dictionary and consists of 103,421 words. The *MagicSpeller SCOWL-6 lite dict* is intended for faster searches and consists of 77,676 words. The largest dictionary, *MagicSpeller YAWL-3 dict*, is based on the YAWL project and is in the public domain. This dictionary is based on the 0.3 release and contains 264,058 words.

Shareware License Agreement

Before using this software on your computer, please carefully read the following terms and conditions.

By using this software you admit agreement with this license agreement and are bound by the terms and conditions within it.

Shareware

infiniteline (“Author”) developed and licenses this software program: *MagicSpeller for REALbasic* (“Software”). This Software is distributed as Shareware. It is NOT free. You may use the Software for a trial period of thirty (30) days, at no cost to you, to determine if it fits your needs. The initial trial use may be restricted to a demo mode in which certain features are limited or excluded. If you decide to use the Software beyond the trial period, you agree to register it and pay the applicable registration fee.

By registering the software, you not only adhere to the license, you support continued development of this software and other products from the Author.

License Grant

This is a License between you (“Licensee”) and the Author. The Author grants to you a non-exclusive license to use the enclosed copy of software in accord with the terms set forth in this License Agreement. The software is owned by the Author or its licensors and protected by copyright and trademark laws.

License Permits

Licensee may use the software on one or more computers. However, this is a single-user license and you are encouraged to buy a license for each developer at your location. Site licenses may be negotiated as well. The Software, once incorporated into a built binary application, is freely distributable within your compiled product.

License Does Not Permit

Licensee may not publicize or distribute any registration code algorithms, information, or registration codes used by this software without permission of the Author. Registration codes found within the public domain (i.e., internet message boards, serial number databases) shall be disabled without warning. Licensee may not charge fees for distribution or delivery of the Software without expressed written consent of the Author.

Distribution

This software may be freely distributed, so long as no fees are charged, and original packaging and documentation are retained. Please do distribute this application to friends and associates. In most cases, linking to <http://www.infiniteline.com/software> is all that would be necessary.

No Warranty

This software has no warranty of fitness, suitability for a particular purpose or satisfactory quality. This Software is provided to licensee “as is”.

Disclaimer

Licensee hereby disclaims and indemnifies the Author from any claims or complaints about the Software. Under no circumstances shall the Author be liable for direct, indirect, special, incidental, or consequential damages resulting from the use, misuse, or inability to use the Software.

MagicSpeller for REALbasic is copyright 2003-2008 by Eric Andrews, DBA infiniteline studios. Technical support for this product can be obtained by email: support@infiniteline.com