February 25, 2024

Embedded Syntax Translator Help Guide

About:

Embedded Syntax Translator Version 1.0.0

Embedded Syntax Translator is a translator of text files to Visual Basic syntax as temporary text files or as readable VB6 bas modules, compatible with VB.Net. These bas modules can be read by current Visual Studio & then manually translated into a script module.vb or as a subroutine(s) as a simple copy and paste.

Description:

Embedded Syntax Translator creates a visual basic text version of any script syntax such as Powershell, Batch, plain text and rich text format languages only. Visual basic embeds the subroutine in a bas module or as individual text files. The translated script is encrypted and is only readable by visual basic. Visual basic cannot however, for its own security reason, directly read & run the translated script itself. It is translated as embedded text, allowing visual basic to easily run its original file format through a command shell. The runtime embedded script be can be loaded in runtime free RAM virtual memory upholding O.S. & information access security from malicious otherwise scripts.

A VB.net shell can redistribute a runtime version of the embedded script to a temporary repository. The temporary repository can then discard, the visual basic shell distributing it can stop its own process leaving the embedded runtime script running solely in free RAM virtual memory or continue to run as file system binding service until the intended application ends its own process.

Application Summary:

Embedded Syntax Translator embeds the script as a subroutine in a VB.net importable bas module or as an individual text file into the development project. The embedded script is encrypted as VB.net script and is only readable by visual basic. Visual basic cannot however, for its own security reasons, directly read & run the embedded script itself. It is translated as a runtime command shell when deployed. The runtime embedded script can be loaded as deployed script or text, allowing visual basic to easily run its original command line shell command format.

All embedded administration access temporary repository appellations must be placed in the ProgramData Folder as a 3 to 8 character non-conventional application name, naming convention.

Example:

App Acme Name - > C:ProgramData\aame

Note:

Special fonts are not supported. Translation error may result. Common special reverse double quote fonts like “Your Text” are not supported. VB.Net may treat them as real quotes causing module import error.

Controls:

See Image1, Embedded Syntax Translator GUI Interface.

About:

Gives an application summary about description.

Help:

There are three toolbar pull downs; Website, Local & License.

Website:

Selecting Website will load https://softglue.net into the browser to contact softglue for help, view this help document and FAQ's on line. There are no FAQ's for Constitution & Ethics application. It's self-explanatory and all FAQ's are answered in this help document.

Local:

Selecting Local will load this help file into the PC's default pdf reader for reading.

License:

Selecting License will load the softglue user license into wordpad for reading.

Select Tmp Folder:

Selecting this check box allows the user to select their own folder and temporary file name for the translated script file as a visual basic text script file with the name, FileName.tmp. Visual basic script, file naming, FileName.vbs is not supported. All Visual basic script programs have to be translated to visual basic per the operating system company as a modern compliant desktop or TCP/IP application such as VB.Net to meet the current operating system custom script applications licensing.

Translate File:

Translate file switch will convert any text readable script to a visual basic application (VBA) sub routine format as described below. This file format can be imported into any visual basic application either as copy & paste syntax as its own sub routine or imported as a bas module. The script below is converted as write lines to a script text file format such as filename.ps1, Filename.js, Filename.py etc.). The developer can easily define & call all the FileSystemObject & all its methods & properties in their own VBA script programs as described in available VBA library resources. See Image 1.

Example 1:

```
Public Sub FileName()
f.writeline "Statement 1"
f.writeline "Statement 2"
.
.
f.writeline "Statement N"
End Sub
```

Translate File (Continued):

Note: Visual basic does not support a dash (-) character as part of sub or function header names. All dash characters are removed from the source File-Name during the file translation process. However, the entire original file name FileName.ext is preserved for generating translated text files as follows: FileName.ext.tmp. All text files are saved to the C:\users\logonId\ AppData\Local\Temp\EBETrans folder.

Translate All Folder Files:

Translate All Folder Files switch translates all text files within a folder to an embedded VBA script formatted file as descried Example 1 in Translate File. See Image 1.

Compatibility Mode:

Previous versions of visual basic application (VBA) and VB.Net are supported with Compatibility Mode. Text files are translated & embedded into previous versions of visual basic such as VB6, VB2010, VB2015 & supported the previous versions of VB.Net. This further allows the support of visual basic script, FileName.vbs & the wscript compiler that has necessary server side support administration functions.

Create Bas Module:

Create Bas Module switch will create a user defined bas module (Drive:\MyRoot\MyFolder\MyModule.bas) from the translated file(s) as described in Translate File & Translate All Folders. Bas modules are supported for VB.Net & can be imported. See Image 1.

Append Bas Module:

Append Bas Module switch will append an existing user defined bas module (Drive:\MyRoot\MyFolder\MyModule.bas) from the translated file(s) as described in Translate File & Translate All Folders. See Image 1.

Image 1:

Embedded Syntax Translator: