October 21, 2026

Application Extension

Extended Application Developers Help Guide

About:

Application Extension Version 1.0.0

Copyright (c) 2019-2024 by Mark L. Alberi. All rights reserved.

Application Extension can be used a GUI window or command line interface to the original startup application. Thereby, giving the original program's runtime, a modern windows look and feel to the user with an additional security socket layer (SSL) runtime memory file system buffer.

Application Extension runtime window is fully configurable by the user as text file object programming of the form settings, labels, and buttons: color, text, text properties, position, and size in a Config.dat file placed in the root of the users Application Extension Extended Application directory. Below is the format of the Config.dat file. To setup the Config.dat file, create a copy of the Config.org file, or use the runtime definition table below as the guide and set the appropriate runtime properties to suite the developer's needs. The Config.dat file will be hidden by the runtime binary if it is not hidden by the developer after creating it.

Application Extension supports executables (FileName.exe), powershell scripts (FileNames.ps1), batch file scripts (FileName.bat and FileName.cmd), and screen saver applications, (FileName.scr). It does not support general purpose script & web applications such as java script (FileName.js), & Hyper Text Markup Languages (HTML, HTM, Hypertext Preprocessor (FileName.php) etc), or visual basic scripts, (FileName.vbs).

A Config.dat configuration file is optional to run Application Extension binary, but is required if the user wants to run their application from it. The user can use Application Extension program "as is" to evaluate. To retain any or all programmed default values, place "default" without quotes in the entries below.
See below the Config.dat format definitions.

About Application Summary:

Application Extension can extend the security socket layer (SSL) security administration. Application Extension extends runtime memory of any application it starts, including command line legacy programs such as JAVA, enabling them to startup as modern windows programs with a basic start up runtime security socket layer (SSL) until the program(s) are migrated to current 64-Bit windows security authorized platforms. Application Extension does not meet all forms of necessary SSL windows security for C syntax platforms such as Java, and should be evaluated by both the software company and the OS Company before authorizing the use of Application Extension as the new windows interface.

Controls:

See Image 1, Custom Form Example Output GUI Interface for details.

Help:

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

About:

Gives an application summary about Application Extension description.

Website:

Selecting Website will load https://softglue.net Technology page 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.

Multiple Application Configuration:

To create multiple application instances, create or copy an existing AppConfig.dat to an existing applications folder called MyFolder Name and create a short cut with argument as follows:

"C:\Program Files\SoftGlue\App Extension\AppExtsnStart.exe" MyFolder Name.

Command Line Support:

While, Application Extension supports any executable, there are executables that are traditional command line applications that require they are started from their root directory / folder on the command line. The .net shell command does not support these types of executables because these executables use their root as the call reference to other root dependency applications & the shell command starts all applications form its own root directory pointer. The shell command will start these applications but these applications will fail to start because the startup reference directory is not their own root directory. In command line terms, the command line pointer is not pointing to their root at startup, so when they start, they try to reference their dependency applications from this directory and fail to start.

The disadvantage, is some development limitations & automation inconvenience. Below is a simple command line shareware script that solves this problem for the shell command and Application Extension's support of these family types of command line applications. For applications like these, command line macro language known as known BATCH or Command is the best solution so it's supported by SoftGlue & Application Extension.

Sample Startup Command script:

::Copyright by Mark L. Alberi 2019-2024

::Root directory command line executable, VB.net command line startup script

@ECHO off

cls

Set AppPath="MyFullPathRoot\"

::Setup startup processes

Set AppProg= "MyApp.[exe, bat, cmd]"

cd %AppPath%

%AppProg%

exit

AppConfig.dat - Definition Table 1:

Number	Property	Example Value All values are in Twips
0	Unused	Enter “default” or C:\Program Files\SoftGlue\AppExtension as a reference. Do not leave black.
1	Try & Taskbar Icon Setting this property will set a taskbar try icon when Application Extension starts.	Application icon full file path. Select Startup Positon & Form Controls 4 to 7 Example: C:\App Path\File.ico
2	External Application Full Startup Path Executables (exe), command line (cmd), batch (bat), powershell (ps1) & screen savers (scr) files are supported with a single argument string option.	Run My App Button Application Full path. Number 29, Run Button Caption Target. Example: %System32%\Notepad.exe<Space><Argument> <Space> & <Argument> = Real space and argument input

AppConfig.dat - Definition Table 1 (Continued):

Number	Property	Example Value All values are in Twips
3	Active Process Name:	Notepad.exe
4	Window Format	0 = Basic Window 1 = Window with border 2 = Window With Frame
5	Window Height	Example: 550
6	Window Width	Example: 650
7	Startup Positon & Form Controls The form upper right corner exit button “X” can be enabled with value 3 to 5	Set window in upper left had corner of screen. 0 = Windows Default Location 1 = Center Screen 2 = Manual 3, 4, 5 = 0, 1 ,2 with a window “X” button
8	Window Visible: True or False Set to false for using start up window and only hide the command console.	Visible = True Hide = False
9	Window Title	Word - Application Extension
10	Full file background graphic picture (*.bmp, jpg, png, tiff, gif, etc)	C:\Program Files\Application Extension\Images\Test.jpg
11	Window, Frame, Label Background Color - Use RBG Red,Green,Blue. Example: 0,0,0 is a black background	194,194,194 = Dark Grey
12	Label Fore Color – Use RBG Red,Green,Blue. Example: 255,255,255 for white labels	Example: 0,0,0 = Black
13	Top Label Window Title New multiple line feeds (10) and carriage returns (13) are supported text labels in a visual basic syntax format.	Example: Demo & Chr(10) & App & Chr(10) & Description Chr(10) = New Line Feed Chr(13)= Carriage Return
14	Top Label Font	Arial
15	Top Label Font Face:	0 = Normal 1 = Italic 2 = Bold

AppConfig.dat - Definition Table 1 (Continued):

Number	Property	Example Value All values are in Twips
16	Top Label Font Size	Example: 24
17	Top Label Left	Example: 240
18	Top Label Top	Example: 100
19	Title label Justification Sets top label justification.	0 = Left 1 = Right 2 = Center
20	Top Label Visible	True or False
21	Lower Label Window Title New multiple line feeds and carriage returns are supported text labels in a visual basic syntax format.	Example: Demo & Chr(10) & Description & Chr(10) & Description2 Chr(10) = New Line Feed Chr(13)= Carriage Return
22	Lower Label Font	Arial
23	Lower Label Font Face	0 = Normal 1 = Italic 2 = Bold
24	Lower Label Font Size	Example: 14
25	Lower Label Left	Example: 268
26	Lower Label Top	Example: 230
27	Lower Title label Justification	0 = Left, 1 = Right, 2 = Center
28	Lower Label Visible	True or False
29	Run Button Caption	&Run or Custom
30	Run Button Height	Example: 37
31	Run Button width	Example: 70
32	Run Button Left	Example: 190
33	Run Button Top	Example: 430
34	Aux Button Caption	&Aux or Custom
35	Aux Button Height	Example: 37
36	Aux Button width	Example: 70
37	Aux Button Left	Example: 280
38	Aux Button Top	Example: 430

AppConfig.dat - Definition Table 1 (Continued):

Number	Property	Example Value All values are in Twips
39	Aux Button Visible	True or False Aux button is not supported with hidden powershell consoles.
40	Exit Button	&Exit or Custom
41	Exit Button Height	Example: 37
42	Exit Button width	Example: 70
43	Exit Button Left	Example: 370
44	Exit Button Top	Example: 430
45	Exit Button Visible	True or False
46	Process Delay Timer extends the extended application startup time and the active progress bar runtime in seconds during startup of the extended application. Choose this value carefully for good startup video behavior.	Example: 1 Minimum allowed startup timer = 0 seconds. Maximum allowed startup timer = 5 seconds. Zero seconds delay is the default. Set to zero for most extended applications.
47	Show or Hide The About and Help Menus	Show = True Hide = False
48	Auto Run Extended Application and Close App Extension	True = Auto run extended app and close app extension after words. False = Only open App Extension Window
49	Active extended application tray process The application to be started runs either in the system tray or on the desktop & its icon shown in the taskbar.	True = Application runs in the tray. False = Application runs on desktop. If the extended application is in the tray, then True must be set to use the Aux Button (39) to show your application from the tray.
50	Exit extended application on exit of application extension	True or False Ture = Extended Application will exit your application upon exit.
51	Exit Application Extension on start of extended application	True or False True = Application Extension will exit upon the start of your application.

AppConfig.dat - Definition Table 1 (Continued):

Number	Property	Example Value All values are in Twips
52	Command Console Sets my application run time command console name Application Extension (AE) supports running powershell FileName.ps1 files when administrator access is given to AE in script, or in the Advanced->Run as administrator Short cut setting. The user must be log in as administrator to grant this authority, and the client must administration access to the powershell console and the script Interpreter.	cmd = Command Line ps = Powershell
53	Hide or Show Command Console Aux button cannot be used with hidden powershell consoles.	0 = Hide 1 = Show

Custom Form AppConfig.dat Table Example:

```
C:\Program Files\SoftGlue\AppExtension
C:\Program Files\SoftGlue\AppExtension\Images\shell32_16806.ico
C:\Program Files\Microsoft Office\root\Office16\WINWORD.EXE C:\Program Files\SoftGlue\AppExtension\Demo.rtf
WINWORD.EXE
1
400
500
1
True
Word
default
255,254,136
```

Custom Form AppConfig.dat Table Example (Continued):

```
0,0,0
Demo & Chr(10) & Application
Arial
2
24
155
100
2
True
Extension
Arial
2
14
193
190
2
True
&Run
37
70
120
290
&Show
37
70
```

Custom Form AppConfig.dat Table Example (Continued):

```
210
290
True
&Exit
37
70
300
290
True
1
True
False
False
True
False
cmd
1
```

Image 1:

Custom Form Example Output:

