

pSplashScreen Version 2.43 Free - Beta User Guide

pSplashScreen is intended to be used as a visual messaging splashscreen for use with COM capable systems such as vbscript , powershell, COM applications etc on any modern Windows OS. COM stands for “Common Object Model” and allows for the splashscreen to receive updates via the com interface.

For example, here is some vbscript which sends a message for pSplashScreen to display to the user

```
Dim poCOM
Set poCOM = CreateObject("pSplashScreen2.Application")
poCOM.SetMessageText("this is a test message to the COM interface!")
```

The freely available version of pSplashScreen is licensed as a non-expiring evaluation edition. This means that all the default basic functionality is available for use for an indefinite period.

The basic functionality allows the following functions

1. Run pSplashscreen from COM systems such as vbscript
2. Send messages via COM for pSplashScreen to display
3. Load the default bmp background bitmap upon startup
4. Set the SplashScreen Windows Foreground/background state in the window order on the desktop
5. Change the font of the text displayed in the Message Status box in the pSplashscreen window

pSplashScreen can also be registered for business or corporate use and allows the following additional options.

1. Set the Window Title text
2. Move/Resize the messaging status window
3. Resize the pSplashScreen window
4. Load custom bitmaps via the COM interface

Using pSplashScreen

pSplashScreen is a COM enabled application.

IMPORTANT See “Explanation of pSplashScreen non-COM functions” further below for important non-COM functions such as [registering the pSplashScreen Com Interface](#) to make it available for COM functions.

IMPORTANT An example script is included with this deployment called “pSplashScreen.vbs”. It would be helpful to view that script file while referencing this guide. If you open the script with vbsedit or some other such full function vbscript editor such as Admin Script Editor, PrimalScript, etc, it will be more clear as the keyword highlighting makes the script easier to read.

BETA specific information: Please send any comments, problems or errors to pauljorgensen5@gmail.com. This is a tested complete fully capable free version, but is being released as a BETA free edition for the near term. A registered version will be available shortly after the short BETA period ends.

1. pSplashScreen loads the default.bmp file as a background when it first starts. The default.bmp included with the application is shown below. If the default.bmp is not available the background will be the default grey background. Backgrounds can be loaded dynamically as needed with the registered version of pSplashScreen.



2. The initial splashscreen startup opened via COM call from vbscript



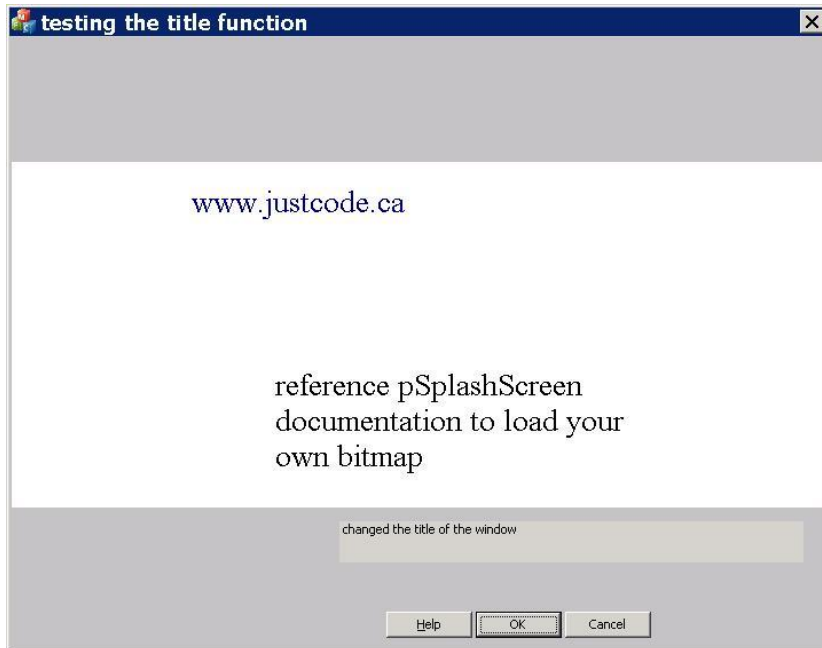
3. The splashscreen checks for registration during the first 5 seconds of execution (it will display messages during this time). Once registration has been received it will display the status on the screen.



4. This first test message via the test vbscript included in the pSplashScreen package.



5. Changing the title of the window via vbscript COM update (requires registered version)



6. Resizing the window via vbscript test script COM call (requires registered version)



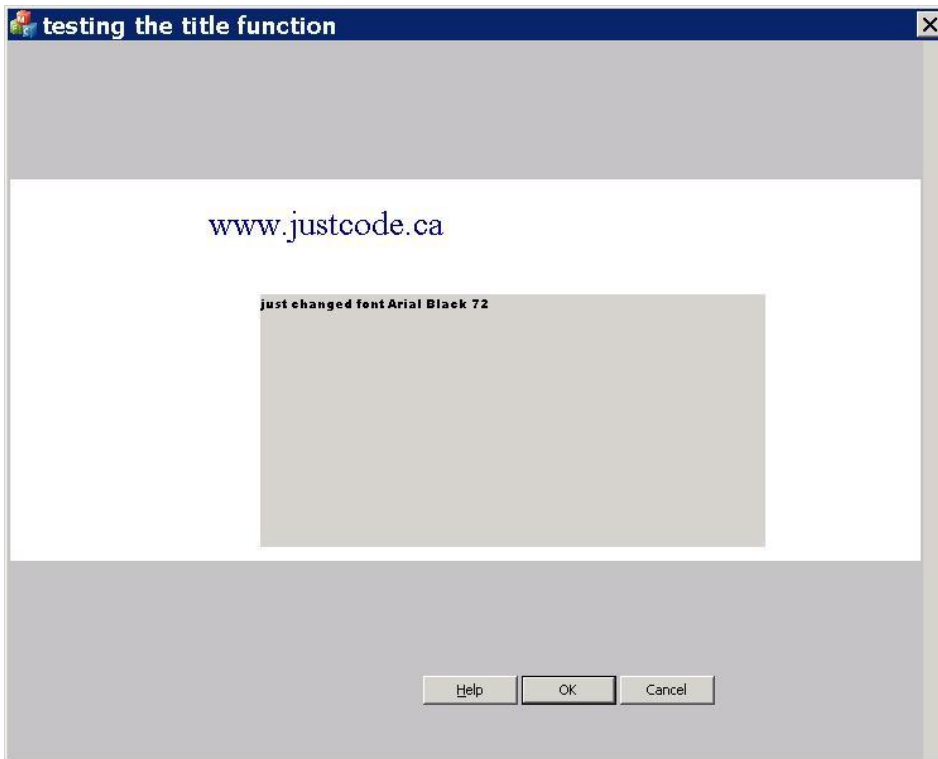
7. Sending a long line of text to the messagebox. The text will be wrapped when it gets to the limits of the message box



8. Moving and Resizing the message window via vbscript COM call (requires registered version)



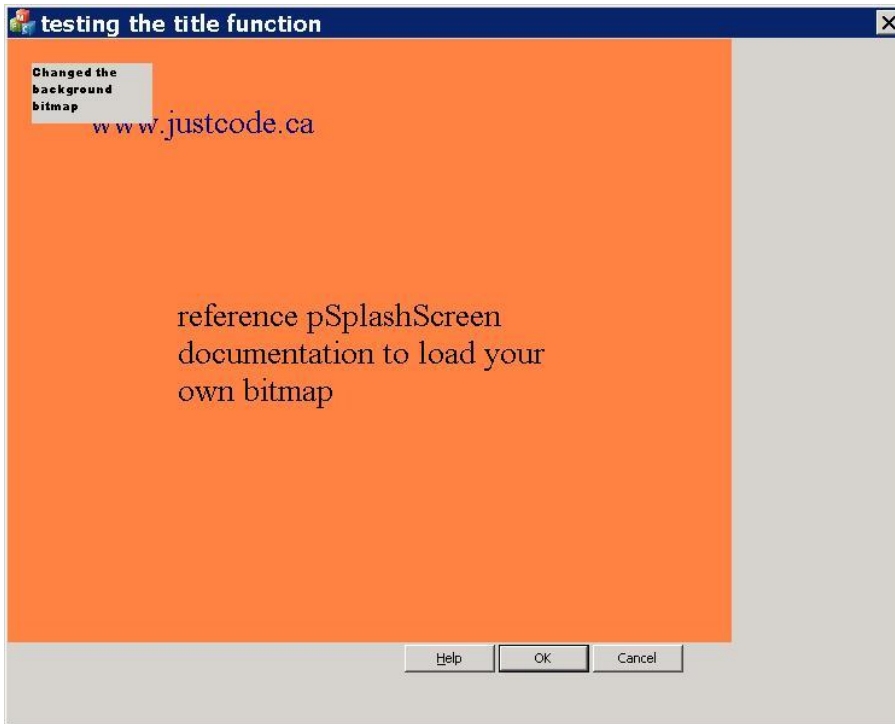
9. Changing the message window font



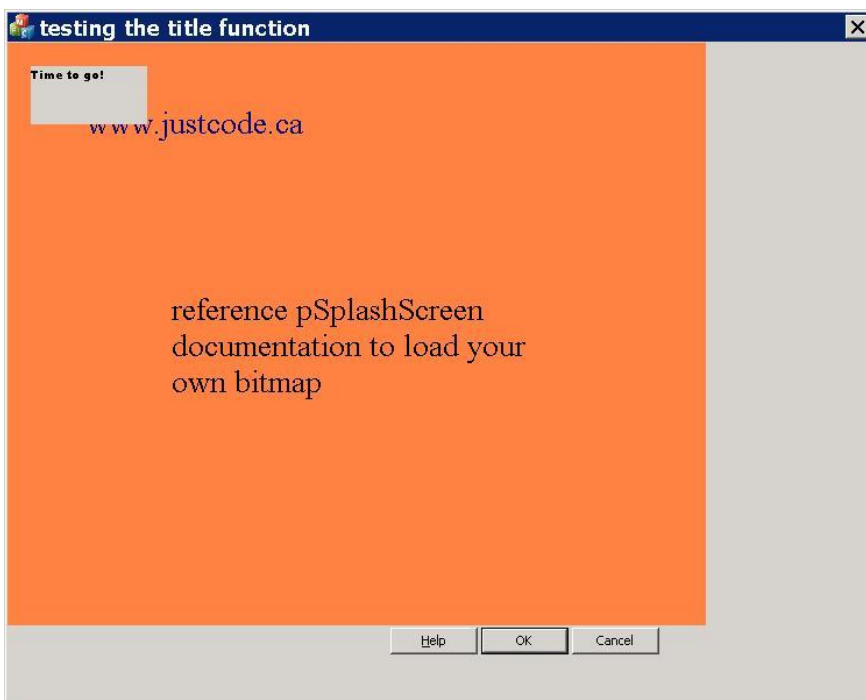
10. Moving and Resizing the Message window again



11. Loading a custom window background via vbscript COM call (requires registered version)
The window should be resized to the size of the bitmap in production deployments.



12. The last message before the script terminates. pSplashScreen auto terminates once script completes.



13. The license message for the Free edition.



14. An error message is displayed when a registered version function is attempted with the evaluation edition. In this case loading a bitmap was attempted with the Evaluation edition.



Explanation of pSplashScreen COM enabled functions

pSplashScreen can be used by any COM enabled system. In this case the examples are provided for vbscript, but for example, powershell could also be used.

1. **IMPORTANT** See "Explanation of pSplashScreen non-COM functions" below for important non-COM functions such as **registering the pSplashScreen COM Interface** to make it available for COM functions.

2. The first thing to do after COM interface registration is to **create the pSplashScreen object** as below

```
Set poCOM = CreateObject("pSplashScreen2.Application")
```

3. Then you need to set the license with the **SetLicense** function, even if it is the Free edition of pSplashScreen. (the vbscript contains the free license key by default)If you do not set the license pSplashScreen will exit after a 5 second grace period.

'Set Evaluation license

```
poCOM.SetLicense("JT58-QEGQ-699R-ETCK-CUCY-DDHF-JWPS")
```

```
WScript.Sleep(1000)
```

4. The **SetMessageText** function is used to send messages to be displayed

```
poCOM.SetMessageText("this is a test message to the COM interface!")
```

```
WScript.Sleep(5000)
```

5. You can send **lines of text longer then the width** of the message window, they will automatically be wrapped.

```
poCOM.SetMessageText("Sending a fairly long line of text to test out the line wrapping of the text display box, adding more text to wrap the line")
```

```
WScript.Sleep(5000)
```

The **SetWindowSize** function sets the size of the full splashscreen window. The parameters are [top x, top y, x extent, y extent]

```
poCOM.SetWindowSize 75,75,600,600
```

```
poCOM.SetMessageText("just changed Window size to 600 x 600")
```

```
WScript.Sleep(5000)
```

7. The **MoveTextMessageBox** function moves and resizes the status message window as required. The parameters are [top x, top y, x extent, y extent, repaint flag(should always be 1)]

```
poCOM.MoveTextMessageBox 200,200,400,200,1
```

```
poCOM.SetMessageText("just moved window to 200,200,400,200")
```

```
Wscript.sleep(5000)
```

The **SetMessageBoxFont** sets the font for the status message window

The parameters are [font name, size]

```
poCOM.SetMessageBoxFont "Arial Black",72
```

```
poCOM.SetMessageText("just changed font Arial Black 72")
```

```
Wscript.sleep(5000)
```

9. The `LoadSplashBitmap` function loads a background bitmap for the splashscreen window.

You will likely want to resize the splashscreen window as required, or make sure all your background bitmaps are the same size.

```
poCOM.LoadSplashBitmap("www-justcode-ca5.bmp")
poCOM.SetMessageText("Changed the background bitmap")
WScript.Sleep(5000)
```

10. The `SetSplashWindowTitle` can be used to change the default splashscreen window title to a custom title.

```
poCOM.SetSplashWindowTitle("testing the title function")
poCOM.SetMessageText(" changed the title of the window")
WScript.Sleep(5000)
```

11. The `SetSplashWndTopMost` can be used to put the pSplashScreen window in front of all other windows. It remains the topmost window until termination or another window state is set via the COM interface.

```
Call poCOM.SetSplashWndTopMost()
poCOM.SetMessageText("set the window to be TopMost window")
WScript.Sleep(5000)
```

12. The `SetSplashWndTopOnly` can be used to make the pSplashScreen Window the top window for all non-topmost windows.

```
Call poCOM.SetSplashWndTopOnly()
poCOM.SetMessageText("set the window to only be the Top window, not the TopMost window")
WScript.Sleep(5000)
```

13. The `SetSplashWndBottom` can be used to make the window a background window.

```
Call poCOM.SetSplashWndBottom()
poCOM.SetMessageText("set the window to be bottom window in Z-Order")
WScript.Sleep(5000)
```

14. The `setSplashWndNoTopMost` can be used to make the pSplashScreen window to not be the topmost window.

```
Call poCOM.SetSpalshWndNoTopMost()
poCOM.SetMessageText("set the window to be WndNoTopMost in Z-Order")
WScript.Sleep(5000)
```

15. Finally you may want to send a final message before your script exits such as "Install complete". The pSplashScreen window will terminate when the script terminates.

```
poCOM.SetMessageText("Time to go!")
Wscript.sleep(5000)
```

Explanation of pSplashScreen non-COM functions

1. Registering the COM interface pSplashScreen. As long as your script is running with the appropriate privileges, usually local admin privileges, you can register the COM interface of pSplashScreen directly from the script as below. pSplashScreen needs to be registered (at least once) with the COM subsystem in order to be called from vbscript, powershell or any COM capable system.

```
Set WshShell = CreateObject("Wscript.Shell")
```

```
'Unregister pSplashScreen first
```

```
Set objWshScriptExec = WshShell.Exec("C:\pSplashScreen238\" & "pSplashScreen.exe /Unregister")  
WScript.Sleep(1000) ' This small wait time is required
```

```
' Then register pSplashScreen
```

```
Set objWshScriptExec = WshShell.Exec("C:\pSplashScreen238\" & "pSplashScreen.exe /Register")  
WScript.Sleep(1000) ' This small wait time is required
```

You should run the /Unregister function first to ensure the register function succeeds. This may depend on the situation, and if pSplashScreen is already registered, you do not need to register it again. Also the script does not specifically need elevated privileges if pSplashScreen is already registered unless your script needs elevated permissions for other functions, such as installations, etc.

You could also check in your script if pSplashScreen is already registered for the COM interface by checking the following registry values, and only run registration if the pSplashScreen COM interface is not registered.

Example pSplashScreen Registry settings

```
HKEY_CLASSES_ROOT\pSplashScreen2.Application = pSplashScreen Application  
HKEY_CLASSES_ROOT\pSplashScreen2.Application\CLSID = {403FF780-0D30-4F11-80EE-D61E5194FF5F}  
HKEY_CLASSES_ROOT\CLSID\{403FF780-0D30-4F11-80EE-D61E5194FF5F} = pSplashScreen2 Application  
HKEY_CLASSES_ROOT\CLSID\{403FF780-0D30-4F11-80EE-D61E5194FF5F}\ProgId = pSplashScreen2.Application  
HKEY_CLASSES_ROOT\CLSID\{403FF780-0D30-4F11-80EE-D61E5194FF5F}\LocalServer32 = pSplashScreen.EXE
```

pSplashScreen has been tested with Windows XP and Windows 7 and should work with any modern version of the Windows OS. **NOTE: If you have any problems running the registration on for example – Windows 7 – try running the script that registers pSplashScreen with Local Elevated permissions such as Local account Administrative permissions. You can also check the permissions of the account that is failing for lack of proper permissions.**

2. pSplashScreen may have the capability in the future to use environment variables and/or the registry to update messages etc. This was used for initial non-public versions of the application and worked well.

pSplashScreen 2.43 Copyright and license

Registered Version License - pSplashScreen 2.43

If you have obtained a registered license for pSplashScreen, the registered license allows you to use the program as specified in the license agreement received with the license. In all cases regarding the registered version, you are licensed to use the program on the number of systems specified in the license. All features available for your registered version will be enabled based on your license key. The Author and www.justcode.ca retain all licensing and ownership rights for the original program, but title and ownership for your registered version is transferred to you for your specific registered copy. This does not convey any rights for you to reverse engineer the software or to use it or distribute it outside of its licensed or intended use, or to use it in any way that was not intended to be conveyed to you via the license. The program may only be released on authorized websites or media that have specifically been authorized by the author. Uploading of the program by the original author to an authorized website for the purpose of distribution or for the BETA program can constitute authorization, but does not remove any of the authors rights in anyway, or the limitation of unauthorized use. Contact the author or www.justcode.ca for distribution permissions. The program is not warrantied to operate in a specific manner on any or all systems, and you are required to test the program and determine that it operates as required for you or your organization.

pSplashScreen 2.43 – COM enabled messaging / status application for use with scripting systems etc
Copyright 2011 Paul Jorgensen, www.justcode.ca, pauljorgensen5@gmail.com

see Free Edition licensing information below

Free Edition license:

The Free Edition program is released free of charge for your use and convenience with an evaluation license key. The author is not liable in any way for its use or any damages incurred by its use. There is no warranty for the program. You may not decompile or reverse engineer the program or alter the program itself in any way. The program may be used with any operating system with which it will function. The BETA Evaluation edition does not include the functionality of the full registered version. To obtain the registered applications functions, you will have to purchase a full registered version when the registered version is released. The program may only be released on authorized websites or media that have specifically been authorized by the author. Uploading of the program by the original author to an authorized website for the purpose of distribution or for the BETA program can constitute authorization, but does not remove any of the authors rights in anyway, or the limitation of unauthorized use. Contact the author or www.justcode.ca for distribution permissions. The program is not warrantied to operate in a specific manner on any or all systems, and you are required to test the program and determine that it operates as required for you or your organization.