

This file contains information regarding changing the Flash GUI of "The Secret World" (TSW).

Contents

01. FILE LOCATIONS	2
02. INTRODUCTION	2
02a. SOURCES FOLDER	3
03. USEFUL SHORTCUTS/COMMANDS	3
04. HOW TO ADD A NEW UI	4
05. ADDITIONAL INFO.....	5
05a. depth_layer.....	5
05b. sub_depth.....	6
05c. flags.....	6
05d. criteria	6
05e. Scaleform CLIK	6
05f. Example File	6

01. FILE LOCATIONS

- Client folder (default for Windows 7):
 - C:\Program Files (x86)\Funcom\The Secret World
- Default Flash GUI files:
 - <Client Folder>\Data\Gui\Default\Flash
- Folder for customized Flash GUI files:
 - <Client Folder>\Data\Gui\Customized\Flash
- Source code for Flash GUI:
 - <Client Folder>\Data\Gui\Customized\Flash\Sources

02. INTRODUCTION

The original TSW Flash GUI is located inside your game client installation folder.

Typically, in Windows 7, it would be:

- C:\Program Files (x86)\Funcom\The Secret World\Data\Gui\Default\Flash

This folder contains all the original SWF files published by Funcom.

Those files will be loaded in case no customized UI is found.

To override the original files with your own version, you will need to publish your own SWF files and place them in:

- C:\Program Files (x86)\Funcom\The Secret World\Data\Gui\Customized\Flash

The new file needs to have the same name as the original file from Funcom.

For example, to override Crafting UI, your file should be:

- C:\Program Files (x86)\Funcom\The Secret World\Data\Gui\Customized\Flash\Crafting.swf

You can find all the source code for the TSW Flash UI in:

- C:\Program Files (x86)\Funcom\The Secret World\Data\Gui\Customized\Flash\Sources

Most UIs have one .fla (Flash) file and one or more .as (ActionScript) files.

02a. SOURCES FOLDER

Inside the Sources folder you will find the following:

- PrecompiledClasses.as and PrecompiledClasses fla
 - These files include any class that is used by more than one UI.
- GUI folder
 - Inside this folder you will find subfolders for all the Flash UIs in the game. Inside the subfolders will be the .fla and .as files.
- GUIFramework folder
 - This folder contains the code for the framework developed by Funcom. It includes low level functionality, such as mouse handling and loading clips. This is the root movie clip for all our UIs.
- Components folder
 - This folder contains the Flash files (.fla) for different basic components used in other UIs. Examples of components are buttons and dropdowns.
- com folder
 - com\Components
 - This folder contains the ActionScript code used by the components in the folder mentioned above.
 - com\GameInterface
 - This folder contains the ActionScript code that can interface with C++ gamecode.
 - com\PatcherInterface
 - This folder contains the ActionScript code that can interface with the C++ code for the game Patcher.
 - com\Utils
 - This folder contains the ActionScript code simple classes with useful utilities.

03. USEFUL SHORTCUTS/COMMANDS

- CTRL+SHIFT+F1 or /reloadui
 - Reloads all the GUI framework
- /loadclip <swf-file>
 - Load a swf file into the game. Game will look first in the Customized folder for the specified name.
- /unloadclip <swf-file>
 - Unload a previously loaded flash file.
- /loadclip DebugWindow.swf
 - Flash Debug window
 - Use this debug window to inspect all the variables in the UI. Please note that you can scroll on this UI (even though no scrolling bar is shown!).

04. HOW TO ADD A NEW UI

Step 1

Create a Modules.xml file (or edit if one already exists) in the Customized folder.

Step 2

Add your own module by adding the info below (if Modules.xml already exists, just add the Module section to the existing file):

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<Root>
  <Module name = "Test01"
    movie      = "Test01.swf"
    flags      = "GMF_CLOSE_ON_ESCAPE"
    depth_layer = "Middle"
    sub_depth  = "0"
    variable   = "test01_module"
    criteria   = "test01_module &&& (guimode && GUIMODEFLAGS_INPLAY)"
  />
</Root>
```

Step 3

Register your module in the preferences.

- For character based UI, use CharPrefs.xml
- For account based UI, use LoginPrefs.xml
- For machine based UI, use MainPrefs.xml

Example:

- Create a file with the name you selected from above in the Customized folder (or edit if one already exists)
- Add the following lines (for example in CharPrefs.xml):

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<Root>
  <Value name="test01_module" value="true"/>
</Root>
```

- “true” will have it open by default

Step 4

Load and see you UI in the game client by copying your Test01.swf into the Customized/Flash folder.

You can use the following commands to see your UI in game:

- /setoption test01_module <true|false>
- /setoption test01_module !test01_module
 - Use this one to be able to easily toggle.

NOTE: Please note that your movie clips MUST implement a onRelease() function for it to receive any mouse events. (It can be an empty function.)

NOTE: Please note that you will need to embed the fonts you use into the flash file.

05. ADDITIONAL INFO

05a. depth_layer

“depth_layer” values map to an enum in CPP code. Here is the mapping:

```
enum ViewLayer_e
{
    e_ViewLayerDebugBack,        // "DebugBack"
    e_ViewLayerBack,             // "Back"
    e_ViewLayerMiddle,           // "Middle"
    e_ViewLayerTop,              // "Top"
    e_ViewLayerOptions,          // "Options"
    e_ViewLayerDialogs,          // "Dialogs"
    e_ViewLayerTooltip,          // "Tooltip"
    e_ViewLayerDragAndDrop,       // "DragAndDrop"
    e_ViewLayerOnScreenMessage,   // "OnScreenMessage"
    e_ViewLayerSplashScreen,      // "SplashScreen"
    e_ViewLayerSplashScreenTop,   // "SplashScreenTop"
    e_ViewLayerDebugTop           // "DebugTop"
};
```

05b. sub_depth

“sub_depth” sorts the UIs within a depth_layer.

If you have all UIs in a layer to have a sub_depth of 0, then the framework will sort by the last clicked.

05c. flags

Module flags define certain behaviours. See the list below.

```
GMF_DONT_UNLOAD      = 0x0020, // Loaded at first use and kept forever
GMF_PRELOAD          = 0x0040, // Loaded at client startup and it is kept forever
GMF_CLOSE_ON_ESCAPE = 0x0080, // Closes when pressen ESC key
```

05d. criteria

Defines an expression that will determine when your UI can be opened.

The start of your expression should always be the variable you defined. In our example: test01_module.

Then you should usually only allow the UI to open while your character is in play. This prevents the UI from showing while on the loading screen or cinematics, etc.

This is achieved by adding the section: `&& (guimode && GUIMODEFLAGS_INPLAY)`

05e. Scaleform CLIK

Funcom uses the Scaleform middleware in our GUI framework. Scaleform has their own set of basic components called CLIK.

Please note that you will not be able to reuse/edit/improve Funcom's flash files that use CLIK components unless you own a license yourself.

You may, however, build your own components (buttons, lists, etc) from scratch.

05f. Example File

You can find a very simple example UI made following this guide in the Customized folder. It is called Test01.swf