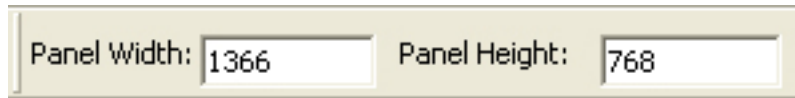
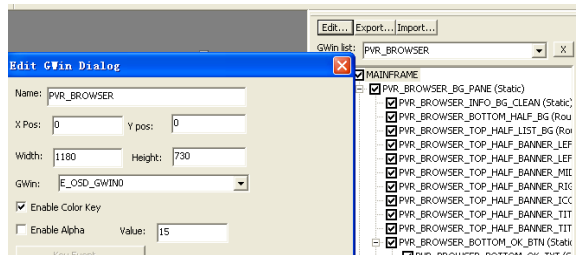


## ZUI 基本操作流程

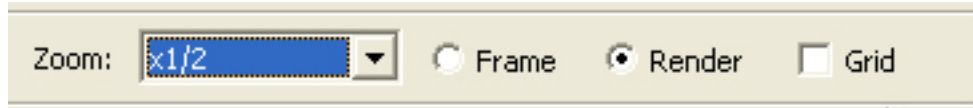
1. 对于一套新的 UI，我们首先要设置我们所用 Panel 的 Size，如 1366X768。



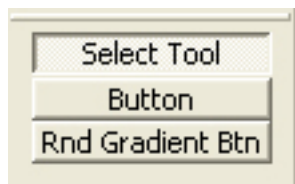
2. 如果要画一个新的菜单，如 PVR\_BROWSER Menu，我们要设置它的名字以及在 Panel 上的位置和大小，其他的可以用默认值。



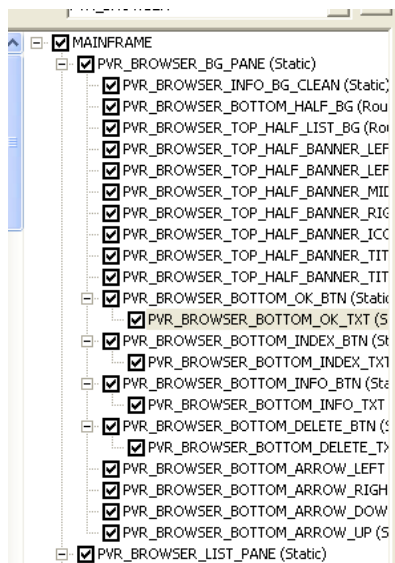
3. 然后选择合适的 Zoom 值(如 x1/2)和钩上 Render，以便我们能预览到我们画的整个菜单。



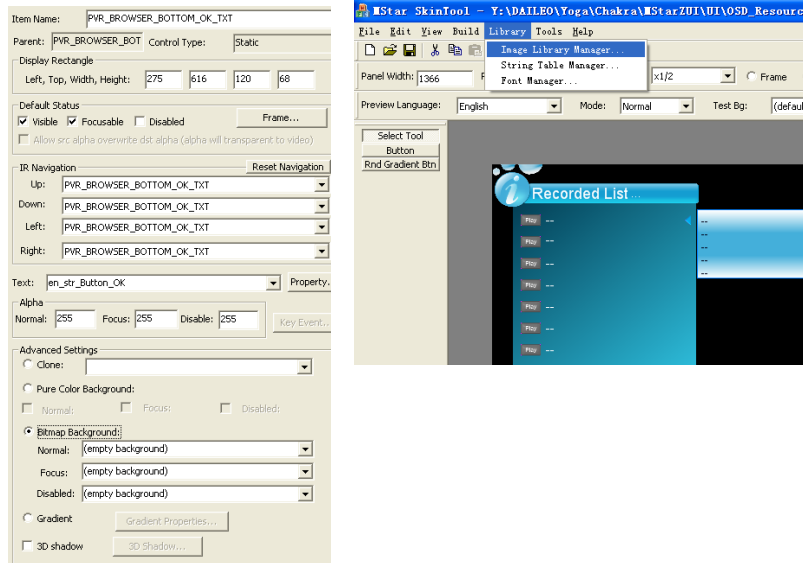
4. 接下来我们就可以用这 2 种 Button 来开始画 UI 了，Button 只能画单色的矩形，Rnd Gradient Btn 既能画单色也能用来画带渐变效果的和弧形角的。



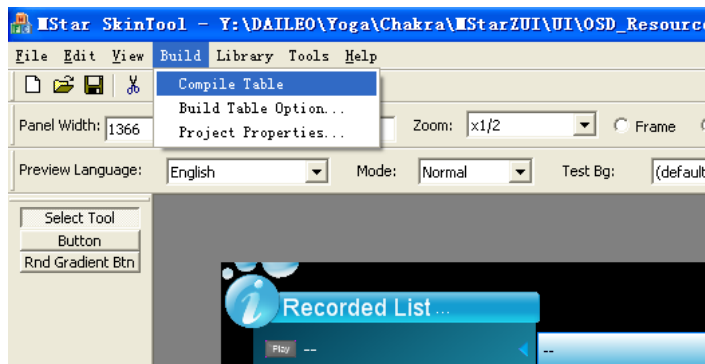
5. ZUI 最基本的元素是 Button，它既能放图片，也能用来显示文字，或者单纯显示某种颜色，一个 Button 下可以挂很多个 Button(即父-子-孙的关系)。一个菜单就是由大量的 Button 堆积形成的。



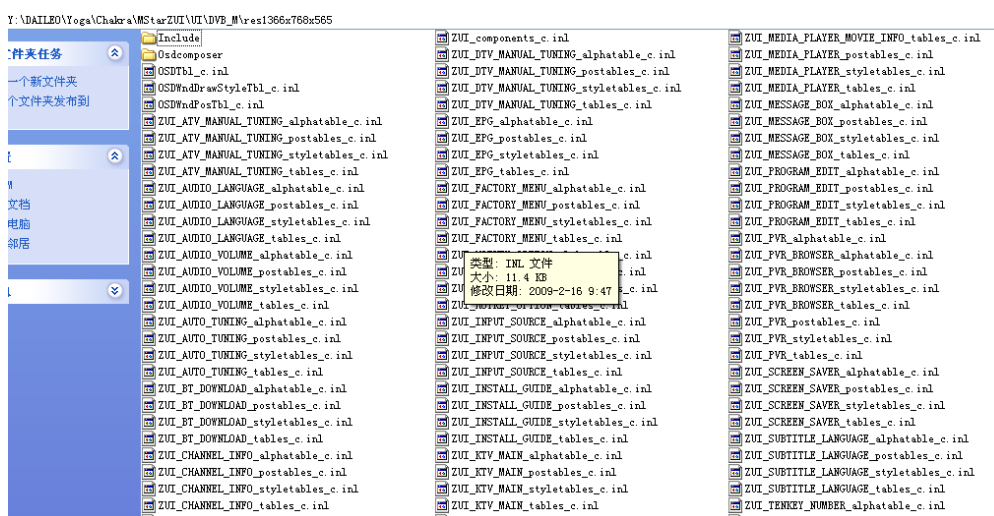
6. 我们画 UI，主要是编辑 Button 的位置/背景色/所用的图片/所显示的文字等属性来达到我们所想要整体效果。但文字和图片要事先导入 ZUI 才行



7. 画完 UI，只是样子画出来了，我们还要 Build ZUI，生成相应的文件并导入 Code 中。



8. 由 ZUI 生成的一百多个文件中，我们只能修改后缀为\_tables\_c.inl 的文件，如 ZUI\_AUDIO\_LANGUAGE\_tables\_c.inl，



9. 后缀为\_tables.c.inl 的文件, 一般我们只能修改每个 Button 五个属性中的 1, 2, 5 位, Bit 1 表示此 Button 是文字可以随时刷新还是图片可随时刷新还是进度条, 等等;  
 Bit 2 表示 Bit 1 所选的属性的参数, 一般为 NULL;  
 Bit 5 表示此 Button 在某些 Key 按下时要转化成什么事件, 以便在 Code 里去做相应的动作, 这个按键→事件的映射要我们自己写, 一般放在后缀为\_tables.c.inl 的文件的前面;

```
// 407 = HWND_MENU_PIC_ADJ_CONTRAST_BAR
{
    EN_ZUI_BALLPROGRESSBAR_WINPROC, NULL, WS_VISIBLE|WS_FOCUSABLE,
    MENU_PIC_ADJ_CONTRAST_BAR_Navigation,
    _MENU_PIC_ADJUST_CONTRAST_KEY,
},
// 408 = HWND_MENU_PIC_ADJ_CONTRAST_OPTION
```

10. 按键→事件的映射加完了, 现在剩下的就是 UI 如何响应并执行我们的按键操作了, 这一部分 Code 都是放在 MApp\_ZUI\_xx.c 中的 MApp\_ZUI\_ACT\_xxAction() 函数里, 如 MApp\_ZUI\_ACTaudiovolume.c 的 MApp\_ZUI\_ACT\_ExecuteAudioVolumeAction() 函数。

```
00122: }
00123:
00124:
00125:
00126: BOOLEAN MApp_ZUI_ACT_ExecuteAudioVolumeAction(U16 act)
00127: {
00128:     switch(act)
00129:     {
00130:
00131:     case EN_EXE_CLOSE_CURRENT_OSD:
00132:     case EN_EXE_POWEROFF:
00133:         MApp_ZUI_ACT_TransitionEffectBegin(EN_EFFMODE_PAGE_FADE_OUT, E_ZUI_STATE_TERMINATE);
00134:         return TRUE;
00135:
00136:     case EN_EXE_RESET_AUTO_CLOSE_TIMER:
00137:         //reset timer if any key
00138:         MApp_ZUI_API_ResetTimer(HWND_VOLUME_CONFIG_PANE, 0);
00139:         return FALSE;
00140:     /*
00141:     case EN_EXE_MUTE_AUDIO_VOLUME:
00142:     {
00143:         BOOLEAN bisAudioMuted;
00144:
00145:         bisAudioMuted = msAPI_AUD_IsAudioMutedByUser();
00146:
00147:
```

11. 还有每个 Button 被刷新时还要重新调用新的图片和文字等, 这些都是  
 MApp\_ZUI\_ACT\_GetxxDynamicText()  
 MApp\_ZUI\_ACT\_GetxxDynamicValue()  
 MApp\_ZUI\_ACT\_GetxxDynamicBitmap() 里修改.

12. 总之, 一个菜单每次显示都是先执行 MApp\_ZUI\_ACT\_AppShowxx(), 去做一些必要的初始化, 然后会调用 MApp\_ZUI\_ACT\_GetxxDynamicText() / MApp\_ZUI\_ACT\_GetxxDynamicValue() / MApp\_ZUI\_ACT\_GetxxDynamicBitmap() / MApp\_ZUI\_ACT\_QueryxxItemStatus() 去取每个 Button 上要显示的文字/图片/状态等信息, 最后根据这些信息把整个菜单全部显示出来。

THE END