# deltax组件结构文档

## 简述

1)Style为gui组件的控制类型属性

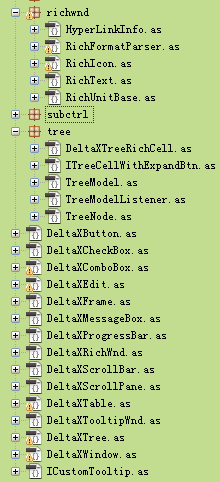
2)ItemType是gui组件包含的部件种类

3)ItemState是每一个部件的状态

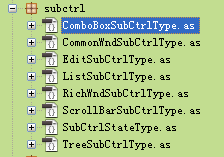
4)所有组件的第一个部件均是CommonWndSubCtrlType. BACKGROUND// 值为1

5)所有公用状态在SubCtrlStateType类当中

6) “->” 表示gui组件类包含的子gui组件



组件类



组件皮肤类型

## 组件公共状态

SubCtrlStateType {

**public** **static** **const** HITTEST\_AREA:uint = 0;//

**public** **static** **const** TOOLTIP\_BACKGROUND:uint = 1;

**public** **static** **const** ENABLE:uint = 2;

**public** **static** **const** DISABLE:uint = 3;

**public** **static** **const** MOUSEOVER:uint = 4;

**public** **static** **const** CLICKDOWN:uint = 5;//按下

**public** **static** **const** UNCHECK\_ENABLE:uint = 6;

**public** **static** **const** UNCHECK\_DISABLE:uint = 7;

**public** **static** **const** UNCHECK\_MOUSEOVER:uint = 8;

**public** **static** **const** UNCHECK\_CLICKDOWN:uint = 9;

**public** **static** **const** PROGRESSBAR\_FILL:uint = 10;

**public** **static** **const** TREE\_ENABLE:uint = 11;

**public** **static** **const** TREE\_DISABLE:uint = 12;

**public** **static** **const** LISTITEM\_SELECTED:uint = 13;

**public** **static** **const** LISTITEM\_NORMAL:uint = 14;

**public** **static** **const** COUNT:uint = 15;

}

## 基础组件类

### DeltaXWindow (组件基类)

Style: WindowStyle

Itemtype: CommonWndSubCtrlType.BACKGROUND

ItemState: SubCtrlStateType .HITTEST\_AREA

SubCtrlStateType.DISABLE

SubCtrlStateType.ENABLE

### DeltaXComboBox

Extends DeltaXWindow

Style: ListStyle, ButtonStyle

-> DeltaXButton

ItemType： ComboBoxSubCtrlType.DROP\_BUTTON

ItemState:

-> DeltaXTable

ItemType： ComboBoxSubCtrlType.LISTBOX\_BACKGROUND

ComboBoxSubCtrlType.LISTBOX\_SCROLLBAR

ComboBoxSubCtrlType.LISTBOX\_SCROLLBAR\_DOWN\_BTN

ComboBoxSubCtrlType.LISTBOX\_SCROLLBAR\_UP\_BTN

ComboBoxSubCtrlType.LISTBOX\_SCROLLBAR\_THUMB

ComboBoxSubCtrlType.LISTBOX\_SCROLLBAR

ComboBoxSubCtrlType.LISTBOX\_SCROLLBAR\_DOWN\_BTN

ComboBoxSubCtrlType.LISTBOX\_SCROLLBAR\_UP\_BTN

ComboBoxSubCtrlType.LISTBOX\_SCROLLBAR\_THUMB

ItemState:

### DeltaXCheckBox

Extends **DeltaXButton**

Style: CheckButtonStyle.SINGLE\_CHECK\_IN\_GROUP

ItemType: CommonWndSubCtrlType.BACKGROUND

ItemState: ~~SubCtrlStateType.MOUSEOVER~~

~~SubCtrlStateType.CLICKDOWN~~

SubCtrlStateType.UNCHECK\_CLICKDOWN

SubCtrlStateType.UNCHECK\_MOUSEOVER

SubCtrlStateType.UNCHECK\_ENABLE

SubCtrlStateType.UNCHECK\_DISABLE

### DeltaXButton

Extends DeltaXWindow

Style: ButtonStyle类

ItemType: CommonWndSubCtrlType.BACKGROUND

ItemState: SubCtrlStateType.MOUSEOVER(代码设置的)

SubCtrlStateType.CLICKDOWN(代码设置的)

### DeltaXEdit

Extends DeltaXScrollPane

Style: EditBoxStyle

-> m\_verticalScrollbar:DeltaXScrollBar;

ItemType: EditSubCtrlType. VERTICAL\_SCROLLBAR

EditSubCtrlType.VERTICAL\_SCROLLBAR\_DOWN\_BTN

EditSubCtrlType.VERTICAL\_SCROLLBAR\_UP\_BTN

EditSubCtrlType.VERTICAL\_SCROLLBAR\_THUMB

-> m\_horizonScrollbar:DeltaXScrollBar

ItemType: EditSubCtrlType.HORIZON\_SCROLLBAR

EditSubCtrlType.HORIZON\_SCROLLBAR\_DOWN\_BTN

EditSubCtrlType.HORIZON\_SCROLLBAR\_UP\_BTN

EditSubCtrlType.HORIZON\_SCROLLBAR\_THUMB

### DeltaXFrame

Extends DeltaXWindow

Style: WindowStyle

### DeltaXProgressBar

Extends DeltaXWindow

Style: ProgressStyle

ItemType: CommonWndSubCtrlType.BACKGROUND

ItemState: SubCtrlStateType.PROGRESSBAR\_FILL

### DeltaXRichWnd

Extends DeltaXScrollPane

Style: LockFlag, RichWndStyle

ItemType: EditSubCtrlType. VERTICAL\_SCROLLBAR

EditSubCtrlType.VERTICAL\_SCROLLBAR\_DOWN\_BTN

EditSubCtrlType.VERTICAL\_SCROLLBAR\_UP\_BTN

EditSubCtrlType.VERTICAL\_SCROLLBAR\_THUMB

ItemType: EditSubCtrlType.HORIZON\_SCROLLBAR

EditSubCtrlType.HORIZON\_SCROLLBAR\_DOWN\_BTN

EditSubCtrlType.HORIZON\_SCROLLBAR\_UP\_BTN

EditSubCtrlType.HORIZON\_SCROLLBAR\_THUMB

### DeltaXScrollBar

Extends DeltaXWindow

Style: ScrollStyle

-> m\_thumbBtn:DeltaXButton

ItemType: ScrollBarSubCtrlType.THUMB

-> m\_incrementBtn:DeltaXButton

ItemType: ScrollBarSubCtrlType.UP\_BUTTON

-> m\_decrementBtn:DeltaXButton

ItemType: ScrollBarSubCtrlType.DOWN\_BUTTON

### DeltaXScrollPane

Extends DeltaXWindow

Style:

-> m\_verticalScrollbar:DeltaXScrollBar

ItemType:

-> m\_horizonScrollbar:DeltaXScrollBar

ItemType:

### DeltaXTable

Extends DeltaXScrollPane

ItemType: CommonWndSubCtrlType.BACKGROUND

ItemState: SubCtrlStateType.LISTITEM\_NORMAL

SubCtrlStateType.DISABLE

SubCtrlStateType.MOUSEOVER

SubCtrlStateType.LISTITEM\_SELECTED

### DeltaXTooltipWnd

Extends DeltaXRichWnd

### DeltaXTree

Extends DeltaXScrollPane

Style:

ItemType: TreeSubCtrlType. VERTICAL\_SCROLLBAR

TreeSubCtrlType. VERTICAL\_SCROLLBAR\_DOWN\_BTN

TreeSubCtrlType. VERTICAL\_SCROLLBAR\_UP\_BTN

TreeSubCtrlType. VERTICAL\_SCROLLBAR\_THUMB

ItemState:

ItemType: CommonWndSubCtrlType.BACKGROUND

ItemState: SubCtrlStateType.ENABLE

SubCtrlStateType.DISABLE

SubCtrlStateType.MOUSEOVER

SubCtrlStateType.CLICKDOWN

SubCtrlStateType.UNCHECK\_ENABLE

SubCtrlStateType.UNCHECK\_DISABLE

SubCtrlStateType.UNCHECK\_MOUSEOVER

SubCtrlStateType.UNCHECK\_CLICKDOWN

## GUI组件文件结构

\*.GUI文件：

**CommonFileHeader**:

Endian: LITTLE\_ENDIAN

Uint32 文件类型

Uint32 版本

Uint32 关联文件家属数

{//for

Uint32 关联文件类型

Uint32 关联文件数

{//for

String 关联文件名

}

}

Uint32 m\_extraDataSize

**WindowCreateParam**:

String 组件类名

Stirng 组件标题

Uint32 组件类型style

Int32 x

int32 y

int32 width

int32 height

int32 xBorder

int32 yBorder

int32 groupID

string fontName

int fontSize

float textHorzDistance

float textVertDistance

uint32 lockFlag//拉伸

string tooltip

string superClassName

string superInfo//?

{//循环3 次

String “”

String “”

String “”

}

Uint32 fadeDuration

Int32 DisplayItem数量

{//循环

//if(Background)

{

displayStateInfoNum

{//循环

Int16 0

Int16 stateInfoName

stateInfoData

}

}else{

Int32 displayItemRect.x

Int32 displayItemRect.y

Int32 displayItemRect.width

Int32 displayItemRect.height

displayStateInfoNum

{//循环

Int16 0

Int16 stateInfoName

stateInfoData

}

}//end\_else

}//end\_循环

StateInfoData

Uint32 fontColor

Uint32 fontEdgeColor

ImageListData

Uint32 imageListNum

{//循环

Uint32 贴图索引

Float 贴图x

Float 贴图y

Float 贴图right

Float 贴图bottom

Float 窗体x

Float 窗体y

Float 窗体right

Float 窗体bottom

Uint32 color

Int16 lockFlag

Int16 drawFlag

}

**ChildWindowParam**:

Uint32 childWindowParamsNum

{//循环

String childName//子组件的name属性

windowParamData -> **WindowCreateParam**

}

## 属性探讨

### ImageDrawFlag

ImageDrawFlag. ZOOM\_WHILE\_SCALE

ImageDrawFlag. ADD\_TEXTURE\_COLOR

ImageDrawFlag. DISABLE\_ALPHA\_TEST

ImageDrawFlag. TILE\_HORIZON

ImageDrawFlag. TILE\_VERTICAL

ZOOM\_WHILE\_SCALE🡨{

If(delWidth!=0){

If(TILE\_HORIZON){

textureRect.left -= delWidth;

}else{

textureRect.right += delWidth

}

}

If(delHeight!=0){

If(TILE\_HORIZON){

textureRect.top -= delHeight;

}else{

textureRect.bottom += delHeight

}

}

}

### LockFlag

RIGHT, LEFT, TOP, BOTTOM

globalRect与wndRect的宽高差决定是左偏移还是左右拉伸或者是上偏移还是上下拉伸。

补充：改属性目前工程用在DisplayImageInfo里面的textureRect和wndRect,还有DeltaXWindow的m\_bounds:Rectangle属性中。(bounds这个就是globalRect)

### Style

MODAL //形态对象，不参与鼠标点击检测列表中

MSG\_TRANSPARENT//控制是否接受鼠标事件

CHILD //是否为字组件，控制是否接收鼠标拖动

# 组件分类

1. deltaxButton

默认按钮

交叉按钮

空白按钮?

1. deltaxCheckButton

勾选按钮

触发按钮

1. deltaxWindow

默认窗体

空白窗体

# 个别属性介绍

## GoupID

## XBorder, Yborder

边距，自身rect相对于父层注册点的距离

## textHorzDistance, textVertDistance

文本横间距，竖间距textHorzDistance是可以实时看到效果，textVertDistance要保存重新载入才可以看到效果

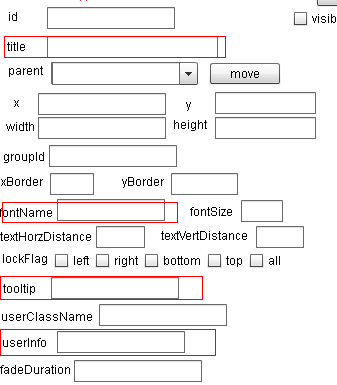
## Userinfo

暂无特别作用

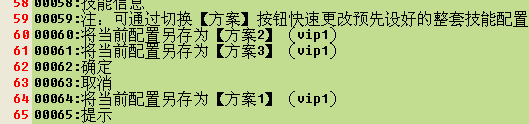
## fadeDuration

渐变时间，主要用于控制组件的渐隐渐现时间(ms)

# 自动翻译



对于图中的四个属性，UI编辑器会在导出gui文件的时候，自动翻译一次到config\dictionary\uilanguage.txt。



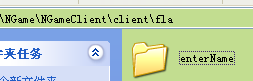
注意：目前对于这个自动翻译表提交冲突还木有较好的方案。

暂时的做法是，需要翻译的时候先手动更新一下这个uilanguage.txt文件。

# 使用流程

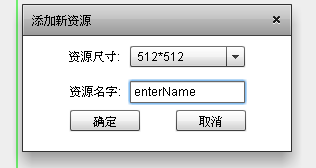
## 基本功能

1. 图片资源管理

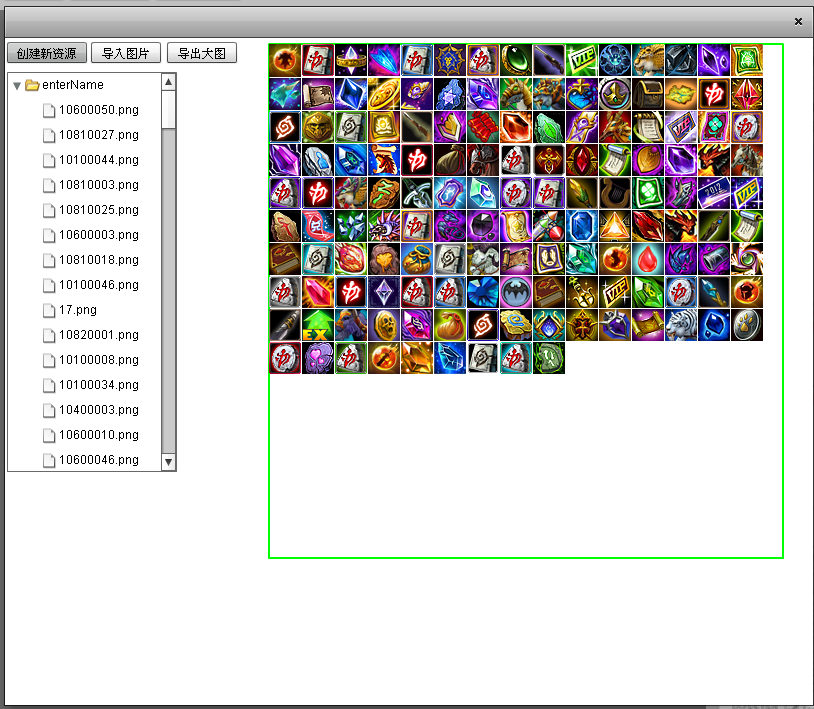
首先把要生成的一张大图的所有图片放入到，自定义大图的文件夹为enterName，必须和大图的名字一样。如大图为enterName.png,那么小图的文件夹为enterName文件夹。

其次：

点击入口，

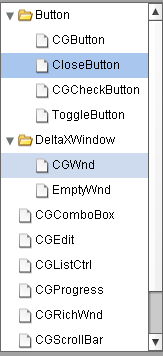
点击创建大图，，输入大图名字，点击确认按钮创建。

下一步，点击导入图片，打开在NGame\NGameCLient\client\fla下上面定义的文件夹enterName下的所有图片载入来。导入后稍等一会，出现生成的大图：



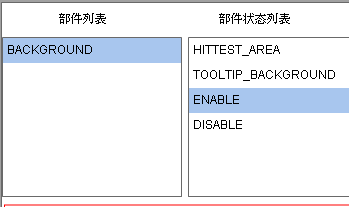
到这里就告一段落。

1. 利用基础ui组件开发新的ui。

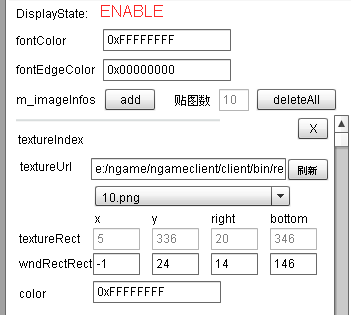
在里面拖拉想要的组件到中间黑色区域，然后再右侧属性面板处就编辑相应的属性。



然后可以通过shift+鼠标左键选择图中要编辑的ui。然后右侧属性栏就会显示ui的各种属性值。其实需要用到贴图的地方有下图：

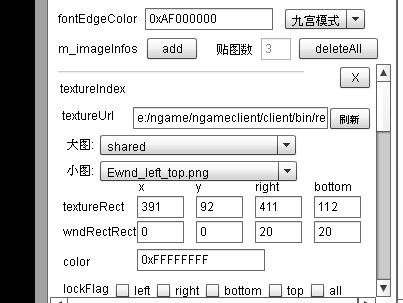


意思就是选中在背景图层的mouseEnable状态的显示效果：



图中的10.png下拉框就可以选择

## 九宫格模式自动配置



要先添加贴图

1首先添加带有对应命名规则的贴图。

如果是添加9张图，则命名规则是，9张图分别每张图要含有~~”left\_top”,”left\_mid”,”left\_bottom”,”mid\_top”,”mid\_mid”,”mid\_bottom”,”right\_top”,”right\_mid”,”right\_bottom”~~。就是九个格子每一列的格子标记。

对应的关键词规则改为：

“lt”, “lm”,“lb”,

“m”, “mm”,“mb”,

“rt”, “rm”,“rb”,

可以按大图，小图，来选择对应的图片。

例如：选择了有这样的图片列表

~~xxxx\_left\_top.png~~ xxxx\_lt.png

~~xxxx\_left\_mid.png~~ xxxx\_lm.png

~~xxxx\_left\_bottom.png~~ xxxx\_lb.png

~~xxxx\_mid\_top.png~~ xxxx\_mt.png

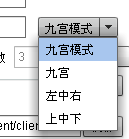
~~xxxx\_mid\_mid.png~~ xxxx\_mm.png

~~xxxx\_mid\_bottom.png~~ xxxx\_mb.png

~~xxxx\_right\_top.png~~ xxxx\_rt.png

~~xxxx\_ right \_mid.png~~ xxxx\_rm.png

~~xxxx\_ right \_bottom.png~~ xxxx\_rb.png

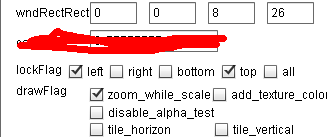


自动设置九宫模式列表

1.九宫就是九张图的九宫格配置

2.左中右就是按钮类型的九宫格配置

3.上中下就是竖直拉伸的图片九宫格配置

选中一个后，对应的图显示的属性就会自动更新

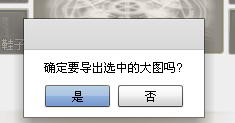
## 自动同步更新gui文件

步骤：

1.打开资源编辑器，并选中一张大图



2.只要对选中大图的所有小图有位置尺寸的修改，都可以通过“同步更新”按钮更新到所有的GUi文件中去。

1. ，点击确定
2. ，这个也要点击确定

就完成了。

实现思路：

通过资源编辑器修改大图，并把Gui文件对应的位置小图索引Rect以及缩放相关的属性进行同步更新。

**注意：**

1. 更新过程需要加载一份原大图配置表用作搜索小图，故在同步更新前，不能马上导出大图。
2. 影响到的文件夹：（若E:\NGame\NGameClient\client\bin\resources\为资源目录）
   1. E:\NGame\NGameClient\client\fla\uiconfig
   2. E:\NGame\NGameClient\client\bin\resources\data\gui\tex
   3. E:\NGame\NGameClient\client\bin\resources\data\gui\cfg

## 代码生成ui

create(\_title:String, \_ style:uint, \_x:int, \_y:int, \_width:int, \_height:int, \_parentWin:DeltaXWindow, \_fontName:String="", \_fontSize:uint=12, \_groupId:int=-1, \_arg11:Object=null, \_arg12:uint=0, \_arg13:uint=1, \_arg14:uint=4278190080, \_arg15:uint=4278190080, \_arg16:uint=0):Boolean{

createFromDispItemInfo(\_title:String, \_arg2:Vector.<ComponentDisplayItem>, \_style:uint, \_parentWin:DeltaXWindow, \_fontName:String="", \_fontSize:uint=12, \_groupId:int=-1):Boolean{

最常用：

createFromRes(fileUrl:String, parentWin:DeltaXWindow=null, callBack:Function=null):Boolean{

reassignChild(childName:String, news:DeltaXWindow):void{

createFromWindowParam(windowParam:WindowCreateParam,parentWindow:DeltaXWindow = null):Boolean

组件都要在加载完毕后才可以调用对应的属性。通过回调函数，和覆盖deltaxwindow里面的 onWndCreated函数，可以知道加载完成,然后进行相应的操作。

# DeltaXWindow功能介绍