# 简单说明

TestLib是ATL用的例子，TestMFC是MFC用的例子，MakeSkin是专用的皮肤打包器，打包后可以是单个文件，也可以嵌入到项目资源中。例子最开始，都有UILib::LoadSkin先加载皮肤，一般Debug下指定路径，加载路径下的图片文件，便于开发调试，Release下指定打包后的整个皮肤文件或资源，提高效率，解包算法完爆用不压缩的.zip。

先从基本的介绍，CImagex是公用图像类，有2个成员std::shared\_ptr<CImage>、CRect，分别是图像的指针、和自己在图中占的区域。比如一个3态的按钮，每态宽高为40、20，则图像总宽高为120、20，加载时指定“按钮.png:3”，就加载了第1态，用SetFrameIndex(n)切换其它态，从源码可以看到，切换只是把自己的CRect改了一下，图像是共享的。

接下来是CUIBase、CUIView、CUIControl、CUIRootView，这4个类把整个UI库驱动起来了，其它控件只是继承CUIView或CUIControl，并特化自己的绘制、键鼠消息。主要看下这4个类的源码，看懂之后，其它控件就都非常简单了。

CUIBase是所有的祖先类，里面方法很少，一看就懂；CUIView继承自CUIBase，它是一个容器类，有成员vector<CUIBase \*> m\_vecChilds，方法更少更好懂；CUIControl继承自CUIView，看名字就知道它是控件的基类，像CUIButton、CUICheckBox等都是继承自它，这个类提供的方法同样很少；CUIRootView继承自CUIView，它是顶层视图，用于和window、windowless交接。整个界面是树形结构，顶层根是RootView，往下是各层各个View、Control，每个控件都能GetRootView使用它。

处理鼠标消息时，树形结构最下层Control先处理，也就是视觉的最上层控件，它会处理鼠标点击消息，即使它不处理，也不给其它控件机会（就像windows的窗口行为一样），如果想让鼠标点击时穿透它，比如CUIImage等，则设置m\_bClickable = false。

MouseEnter、MouseLeave这2个消息特殊，如果它的Parent是CUIControl，则会让Parent也同步MouseEnter、MouseLeave。比如浏览器标签：标签是按钮，右边的×号是它的子按钮，你试试鼠标放在×号上，是不是标签和×号同时MouseEnter了。

UI库最核心的部分，是如何布局所有的控件，算法在CUIView的RecalcLayout和CUIBase的CalcRect。对于每个View（Control也是View），包括RootView，当它的区域改变时，会先计算出自己的m\_rect，然后再布局所有子控件，每个子控件布局后，如果和之前相同则返回，否则就又会布局它的子控件，这样依次递归。

RecalcLayout算法：初始rect为自己的m\_rect，依次把这个rect传给子控件，让它们按照约束，CalcRect出自己需要的部分，并把剩下的部分返回。

CalcRect算法：传进来的rect，对于左右方向，有左、右、宽3个属性，都不设置的话：left=rect.left、right=rect.right。只设置1个属性的情况：只设置左则left=rect.left+左、right=rect.right，只设置右则left=rect.left、right=rect.right-右，只设置宽则left、right在rect居中。同时设置2个属性的情况：设置左、右就定下来了，设置左/右、宽就能算右/左。如果同时设置3个属性则只用左、右。上下方向同理。

指定了左、宽时，可以同时指定左剪裁，就是算出自己的rect2后，剩余rect.left=rect2.right并返回，是否剪裁看SetLeft第2个参数，xml中对应的是数字后加\*号，右、宽，上/下、高剪裁都同理。

如果控件使用了图片，比如Button、Image等，可以不用设置宽高，因为初始宽高就是图片的宽高，也可以根据需要SetWidth、SetHeight。

例如构建一个简单窗口，先在RootView上用top、height划出标题栏并剪裁，剩下的就是客户区。然后再在标题栏View上用right划出关闭按钮并剪裁（不用指定width），再用right划出最小化按钮，再用left划出窗口名等等。

关于RecalcLayout性能，使用了延迟布局算法，任何控件添加、删除、改变上下左右宽高，都会InvalidateLayout只记录一次，在最后需要WM\_PAINT前一次性布局所有需要的，再DoPaint。

构建界面支持纯代码、xml配置、或两者结合，例子里都有，xml中每个控件可配置的属性，可以参考每个类cpp最后的OnLoaded，注意还有\_\_super::OnLoaded。

# 各类介绍

每个类源码都非常短，可以自己看看，也可以只看头文件，会用就行，以后有时间再写。