

一、控件布局

1.1 为什么需要布局参数

如果界面上元素是预先知道的，而且窗口的大小也是固定的，通过可视化的工具，以所见即所得的方式，去创建界面是最轻松的方式。但是在下列情况下，使用布局参数却是更好的选择。

- 窗口的大小可以动态调整的。
- 需要适应不同大小的屏幕。
- 界面上的元素是动态的，需要用程序创建界面。

[AWTK](#)提供了简单而又强大的布局参数

1.2 控件自身的布局参数

1.2.1 像素

直接指定控件的x/y/w/h的像素值，这是缺省的方式，也是最缺乏灵活性的方式。

示例：

- 在XML界面描述文件中：

```
<button x="10" y="5" w="80" h="30" text="ok"/>
```

- 在代码中：

```
widget_move_resize(btn, 10, 5, 80, 30);
```

1.2.2 百分比

- x/w的值如果包含"%", 则自动换算成相对其父控件宽度的百分比。
- y/h的值如果包含"%", 则自动换算成相对其父控件高度的百分比。

示例：

- 在XML界面描述文件中：

```
<button x="10%" y="10" w="50%" h="30" text="ok"/>
```

- 在代码中(看起来要麻烦一点)：

```
widget_set_self_layout_params(btn, "10%", "10", "50%", "30");  
widget_layout(btn);
```

在代码中设置控件的布局参数，方法类似，后面就不再举例子了。

1.2.3 水平居中

让控件在水平方向上居中，只需要将x的值设置成"c"或者"center"即可。

示例：

```
<button x="center" y="10" w="50%" h="30" text="ok"/>
```

1.2.4 垂直居中

让控件在垂直方向上居中，只需要将y的值设置成"m"或者"middle"即可。

示例：

```
<button x="center" y="middle" w="50%" h="30" text="ok"/>
```

1.2.5 位于右边

让控件位于父控件的右侧，只需要将x的值设置成"right"即可。

示例：

```
<button x="right" y="10" w="50%" h="30" text="ok"/>
```

如果还想离右侧有一定距离，可以在right后指定距离的像素。

示例：

```
<button x="right:20" y="10" w="50%" h="30" text="ok"/>
```

1.2.6 位于底部

让控件位于父控件的底部，只需要将y的值设置成"bottom"即可。

示例：

```
<button x="10" y="bottom" w="50%" h="30" text="ok"/>
```

如果还想离底部有一定距离，可以在bottom后指定距离的像素。

示例：

```
<button x="10" y="bottom:20" w="50%" h="30" text="ok"/>
```

1.2.7 宽度和高度为负数

无论是像素模式还是百分比模式，宽度和高度均可为负数。

- 宽度为负数。其值为父控件的宽度+该负值。
- 高度为负数。其值为父控件的高度+该负值。

1.3 子控件的布局参数

为了方便父控件布局子控件，AWTK提供了下面几个参数：

参数	说明
rows	行数，UI描述文件中可简写为r
cols	列数，UI描述文件中可简写为c
width	子控件的宽度(可以用来计算列数，与cols互斥)，UI描述文件中可简写为w
height	子控件的高度(可以用来计算行数，与rows互斥)，UI描述文件中可简写为h
x_margin	水平方向的边距，UI描述文件中可简写为x
y_margin	垂直方向的边距，UI描述文件中可简写为y
margin	边距(相当于同时设置x_maring/y_margin)，UI描述文件中可简写为m
spacing	子控件之间的间距。UI描述文件中可简写为s

在代码中，可以通过下面的函数设置这几个参数：

```
ret_t widget_set_children_layout_params(widget_t* widget, const char* params);
```

在XML中，可以通过layout设置这几个参数(参数的顺序无关，重复以后者为准):

```
<group_box x="0" y="140" w="100%" h="90" layout="rows:2 cols:2 spacing:10 maring:5">
  <button x="0" y="0" w="0" h="0" text="button"/>
  <button x="0" y="0" w="0" h="0" text="button"/>
  <button x="0" y="0" w="0" h="0" text="button"/>
  <button x="0" y="0" w="0" h="0" text="button"/>
</group_box>
```

上面的layout值也可以简写为：

```
layout="r2 c2 s10 m5"
```

下面我们看看，如何调整rows/cols两个参数，来实现不同的布局方式。

1.3.1 缺省

在没有设置子控件布局参数或者rows/cols都为0时，采用缺省的布局方式，父控件啥事也不做，完全由子控件自己的布局参数决定。

1.3.2 hbox

当rows=1,cols=0时，所有子控件在水平方向排成一行，可以实现其它GUI中hbox的功能。子控件的参数：

- x 从左到右排列，由布局参数计算而出。
- y 为y_margin
- w 由子控件自己决定。
- h 为父控件的高度-2*y_margin

1.3.3 vbox

当cols=1,rows=0时，所有子控件在垂直方向排成一行，可以实现其它GUI中vbox的功能。子控件的参数：

- x 为x_margin
- y 从上到下排列，由布局参数计算而出。
- w 为父控件的宽度-2*x_margin
- h 由子控件自己决定。

1.3.4 listbox

当cols=1,rows=N时，所有子控件在垂直方向排成一行，可以实现其它GUI中listbox的功能。子控件的参数：

- x 为x_margin
- y 从上到下排列，由布局参数计算而出。
- w 为父控件的宽度-2*x_margin
- h 为父控件的高度(减去边距和间距)分成N等分。

1.3.5 grid

当cols=N,rows=N时，所有子控件放在NxN的网格中，可以实现其它GUI中grid的功能。

优先使用父控件指定的layout参数，父控件有layout参数时，除特殊情况(vbox中的高度和hbox中的宽度)外，忽略子控件的xywh参数。