

千千静听皮肤制作教程【新增 5.2.0 版本音乐窗脚本详解】

一、了解千千静听皮肤制作是怎么一回事？	1
二、开始制作啦！	1
(A) 切图：	1
(B) 修改配置文件	6
(C) 打包成皮肤文件	13
(D) 应用皮肤	13
三、几个重要的属性解释	14
四、皮肤制作注意事项及技巧：	15

注：此教程不介绍 Photoshop 或 Fireworks 等图像图形处理软件设计皮肤过程

一、了解千千静听皮肤制作是怎么一回事？

1、如果您是位从未接触过皮肤制作的人，那么请先仔细看下面的这段话：

如何把设计好的皮肤效果图应用到千千静听软件上去，有两个主导思想您应该了解：一个是需要把效果图上面的控件（或称按钮）单独切出来，另一个是需要把上一点说的控件（或称按钮）的坐标找到，为它精准定位，是不是听的有点眉目了，那么我们接着往下讲。

2、千千的皮肤位于安装目录下的 Skin 文件夹内，扩展名可以为 .skn 或 .zip，实际上二者是一样的。对于前者，可以先将 .skn 的扩展名改为 .zip（要在系统中显示文件的扩展名，依次点击“工具”-“文件夹选项”-“查看”，再把“隐藏已知文件类型的扩展名”前的小勾去除即可），然后将其解压到单独的文件夹，进入该文件夹，可以发现里面包含了许多 bmp 格式的图片 and 若干个 xml 文件，他这些文件便是皮肤的组成部分了，bmp 图片是各个窗口的背景及按钮图片，Skin.xml 则是配置文件，定义了皮肤的基本信息、窗口及按钮的位置、大小等，它是基于 XML 格式的文件，可直接用系统自带的记事本或者其它文本编辑工具打开的。

好，了解以上的基础知识后，我们就开始学习实际的制作过程

二、开始制作啦！

按照四个步骤进行：(A) 切图 (B) 修改配置文件 (C) 打包成皮肤文件 (D) 应用皮肤

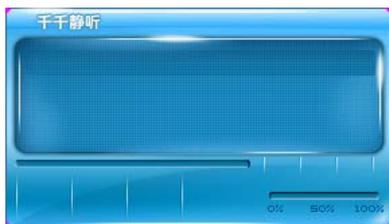
(A) 切图：

用 PS 或 FW 打开设计效果图，整体观察一下，下面讲一下哪些图片是要单独切出来的以及图片的命名。



一、主窗口的控件

(1) 主窗口背景 命名: player_skin.bmp



重点: 边缘小圆角的镂空处理, 把镂空填充成 (#ff00ff) 这个颜色, 要细心处理这部分哦, 边缘要 1 像素 1 像素的填充 (如上图, 镂空部分的颜色处理)

(2) 最小化按钮 (4 个状态)

命名: minimize.bmp



重点: 凡是功能按钮都要做 4 种状态, 并把这四种状态做在一张图里, 存储成.BMP 格式, 注意每种状态按钮他的宽度和高度要一致, 说一下每种状态代表的含义

第一个状态: 自然状态

第二个状态: 鼠标划过时的状态

第三个状态: 鼠标按下去时的状态

第四个状态: 按钮失效时的状态 (举个例子, 比如播放列表只有一首歌曲, 那么“下一首”按钮就是无法点击的, 那么此时这个按钮状态就是失效时的状态)

(3) 迷你模式按钮

命名: minimode.bmp



同上, 这里不再赘述了。

(4) 关闭按钮

命名: close.bmp



(5) 播放进度滑块 (本例中播放进度条上的小圆按钮)

命名: progress_thumb.bmp



重点: 这个小按钮是需要脱离背景部分单独扣出来的, 那么镂空部分需要如何处理才能最终在界面上显示出透明的效果呢, 解决的办法和上面的大背景镂空处理一样就是把镂空填充成 (#ff00ff) 这个颜色, 如上图

(6) 播放进度填充背景图

命名: progress_fill.bmp



(7) 播放列表窗口打开关闭按钮

命名: playlist.bmp



(8) 均衡器窗口打开关闭按钮

命名: equalizer.bmp



(9) 歌词窗口打开关闭按钮

命名: lyric.bmp



(10) “上一首”按钮

命名: prev.bmp



(11) “播放”按钮

命名: play.bmp



(12) “暂停”按钮

命名: pause.bmp



(13) “下一首”按钮

命名: next.bmp



(14)音量小喇叭按钮

命名: mute.bmp



(15)音量进度背景填充图

命名: progress2.bmp



(16) 音量滑块

同 (5)

二、均衡器窗口的控件

(17) 开启按钮

命名: eq_enabled.bmp



(18) 重设按钮

命名: reset.bmp



(19) 配置按钮

命名: eq_profile.bmp



(20) 关闭按钮, 切图同主窗口上的关闭按钮

(21) 平衡器环绕声所有滑动的小按钮 同 (5)

(22) 平衡, 环绕填充背景

命名: eqfactor_full2.bmp



(23) 均衡填充背景

命名: eqfactor_full.bmp



三、播放列表窗口的控件

(24) 关闭按钮, 同主窗口关闭按钮

(25) 工具条按钮,

命名: playlist_toolbar.bmp



热点状态命名: playlist_toolbar_hot.bmp



(26) 滚动条上下按钮

命名: scrollbar_button.bmp

 重点: 将上下按钮拼在一张图上制作

(27) 滚动条滑动按钮

命名: scrollbar_thumb.bmp



(28) 滚动条背景

命名: scrollbar_bar.bmp



四、歌词秀窗口的控件

(29) 关闭按钮, 同主窗口

(30) 总在最前按钮

命名: ontop.bmp



五、音乐窗窗口的控件



此处为html网页部分，不属于皮肤范围

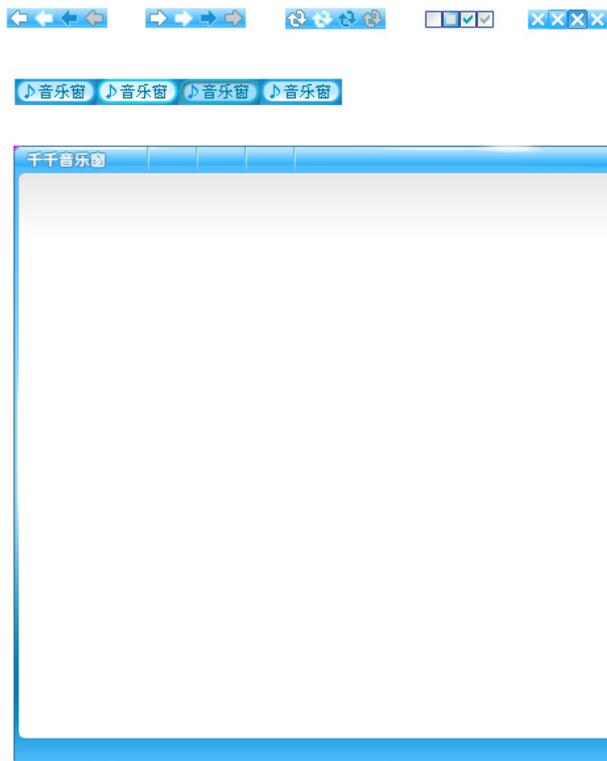
上图为千千音乐窗界面，其中用绿色边框套住的部分为显示部分，和皮肤设计无关，就是说我们要做的是绿色边框外的内容。

好，明确了制作部分，开始讲制作过程，首先，要制作一个窗口背景，就是图中显示的最外面的蓝色风格的窗口，像主窗口一样，不难理解，格式同样为.bmp，需要设置透明色背景（#FF00ff），注意圆角像素的处理。

然后制作窗口里的控件，上图中用红色框标出了所有控件，包括后退、前进、刷新、关闭、多选框、连接文字区。其中后退、前进的功能是像网页一样的控制当前页面，并不是歌曲的后退、前进，不过也没影响，不多说了。

最后，还要制作一个按钮，用来打开音乐窗，按钮要做在主窗口上，做主窗口时别忘了留出地方哦...

具体切图示例如下，大家一看就会明白啦：



(B) 修改配置文件

一、首先看一下 skin.xml 这个配置文件

```
<skin version="2" name="静听·蔚蓝" author="千千静听" url="http://www.ttplayer.com" email="none" transparent_color="#ff00ff">
```

以上是皮肤的基本描述信息，请根据您的情况填写，分别是版本号，皮肤名称，皮肤作者，地址，电子邮箱，透明色的设置

1、<player_window>和</player_window>之间的代码

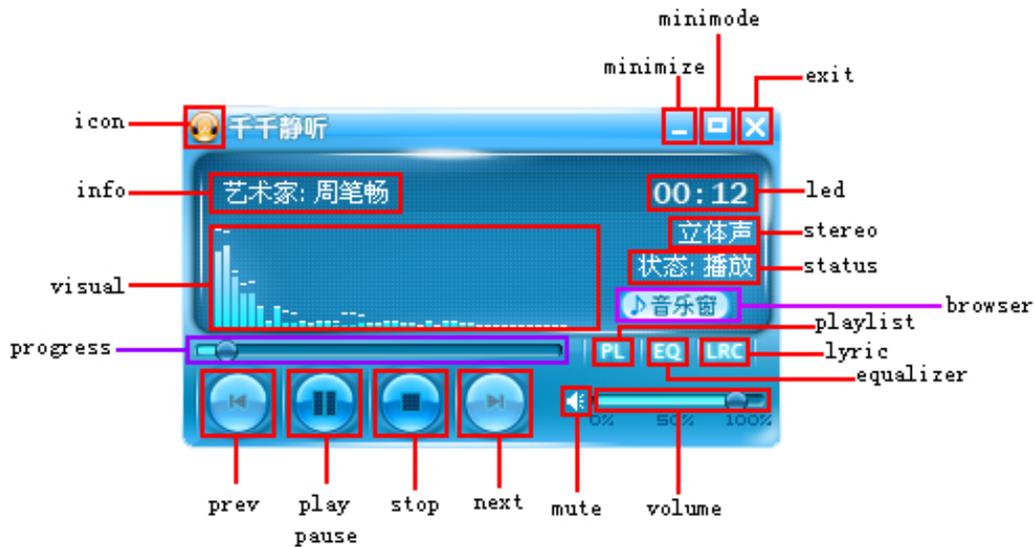
它是描述主窗口的参数设置的

Position 是坐标定位，image 是图片名称，就是我刚才讲述的每个图片的命名

坐标由 4 个数字组成，中间用逗号隔开，前两个数字是图片左上角的 x 坐标和 y 坐标，后两个数字是图片右下角的 x 坐标和 y 坐标，请注意，这里的右下角 x 坐标和 y 坐标都要多算一个点，否则播放器会少显示两条边；

这里需要注意的是：每个窗口的位置是组合窗口后抓的坐标，而每个窗口上面的按钮控件是单独定位的，比如，我们要获得歌词秀窗口上面的关闭按钮的坐标，是要把歌词秀窗口的左上角定位在切图软件 (0, 0) 坐标上，然后再抓关闭按钮的坐标。

请参考下图来理解代码



```

<player_window image="player_skin.bmp">
  <play position="50, 123, 90, 163" image="play.bmp"/>
  <pause position="50, 123, 90, 163" image="pause.bmp"/>
  <prev position="9, 123, 49, 163" image="prev.bmp"/>
  <next position="132, 123, 172, 163" image="next.bmp"/>
  <stop position="91, 123, 131, 163" image="stop.bmp"/>
  <mute position="183, 137, 194, 148" image="mute.bmp" />

  <lyric position="248, 112, 272, 126" image="lyric.bmp"/>
  <equalizer position="223, 112, 242, 126" image="equalizer.bmp"/>
  <playlist position="197, 112, 216, 126" image="playlist.bmp"/>
  <browser position="214, 86, 276, 106" image="browser.bmp" />

  <minimize position="230, 3, 247, 21" image="minimize.bmp"/>
  <minimode position="248, 3, 265, 21" image="minimode.bmp"/>
  <exit position="266, 3, 283, 21" image="close.bmp"/>

  <progress      position="7,      112,      184,      123"      bar_image=""
  thumb_image="progress_thumb.bmp" fill_image="progress_fill.bmp"/>
  <volume      position="197,      136,      277,      147"      vertical="false" bar_image=""
  thumb_image="progress_thumb.bmp" fill_image="progress2.bmp"/>

```

以下文字是播放器上面的一些显示文字的设置

Icon 是千千静听的 logo;

info 是音乐标题和专辑歌手的信息，轮显在播放器窗口上;

led 是时间数字，这里不是文字代码，是做好了一张图片，这张图片由大小相等的 12 个字符

组成， 0 1 2 3 4 5 6 7 8 9 : -

记住这 12 个字符缺一不可。

Stereo 是立体声的字体设置

Status 是状态的字体设置

Visual 是视觉效果的设置，这里面只是简单定义了位置，更详细的设置请看 Visual.xml 文件

```
<icon position="4, 3, 20, 19" image="TTPlayer.ico"/>
<info position="21, 34, 208, 49" color="#ffffff" font="Tahoma" font_size="13"
align="left"/>
<led position="160, 37, 270, 49" image="number.bmp" align="right"/>
<stereo position="224, 54, 273, 70" color="#ffffff" font="Tahoma" font_size="13"
align="right" />
<status position="200, 70, 273, 86" color="#ffffff" font="Tahoma" font_size="13"
align="right" />
<visual position="17, 56, 185, 106"/>
</player_window>
```

2、歌词秀窗口的代码

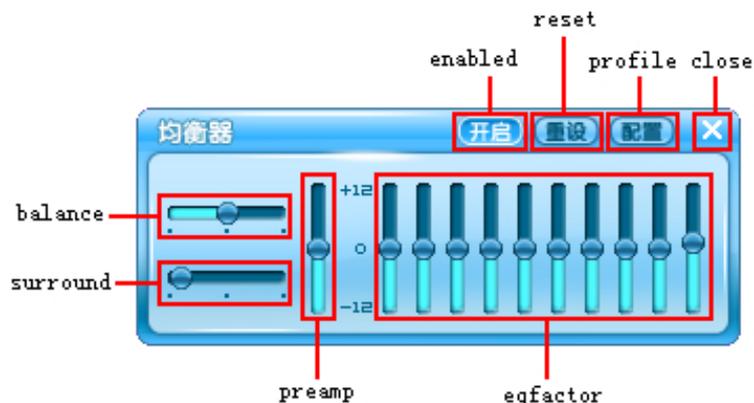
参考如下图一起看



注意：lyric是歌词的文字显示区域，re_size_rect是拉伸窗口时的填补区域，那么技巧就是re_size_rect区域要在lyric区域内部，把皮肤上“歌词秀”这样的文字圈在外面，避免拉伸窗口时，文字跟着变形

```
<lyric_window position="0, 393, 287, 475" resize_rect="49, 25, 245, 73"
image="lyric_skin.bmp">
  <lyric position="10, 25, 277, 73"/>
  <close position="221, 3, 283, 20" image="close.bmp" align="right"/>
  <ontop position="202, 3, 264, 20" image="ontop.bmp" align="right"/>
</lyric_window>
```

3、均衡器窗口的代码

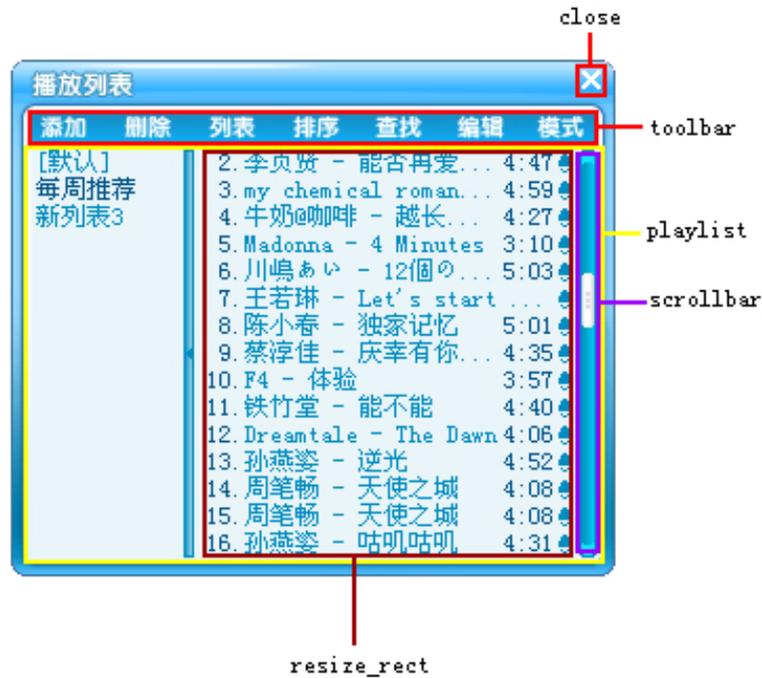


注意：eq_interval="5"这个是
eqfactor每一个之间的距离

```
<equalizer_window position="0, 165, 287, 279" image="eq_skin.bmp" eq_interval="5">
```

```
    <close position="266, 3, 283, 20" image="close.bmp" align="right"/>
    <enabled position="152, 3, 187, 21" image="eq_enabled.bmp"/>
    <profile position="224, 3, 259, 21" image="eq_profile.bmp"/>
    <reset position="188, 3, 223, 21" image="reset.bmp"/>
    <balance position="15, 45, 71, 56" thumb_image="progress_thumb.bmp" bar_image=""
fill_image="eqfactor_full2.bmp"/>
    <surround position="15, 76, 71, 87" thumb_image="progress_thumb.bmp" bar_image=""
fill_image="eqfactor_full2.bmp"/>
    <preamp position="81, 36, 92, 99" thumb_image="progress_thumb.bmp" bar_image=""
fill_image="eqfactor_full.bmp"/>
    <eqfactor position="115, 36, 126, 98" thumb_image="progress_thumb.bmp"
bar_image="" fill_image="eqfactor_full.bmp"/>
</equalizer_window>
```

4、播放列表窗口的代码

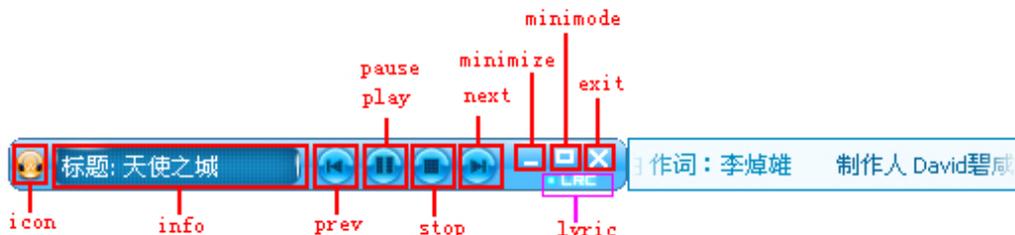


```

<playlist_window position="0, 279, 287, 393" resize_rect="61, 43, 265, 105"
image="playlist_skin.bmp">
  <close position="221, 2, 283, 19" image="close.bmp" align="right"/>
  <toolbar position="10, 24, 278, 41" image="playlist_toolbar.bmp"
hot_image="playlist_toolbar_hot.bmp" align="left"/>
  <scrollbar buttons_image="scrollbar_button.bmp" thumb_image="scrollbar_thumb.bmp"
bar_image="scrollbar_bar.bmp" align="center"/>
  <playlist position="10, 43, 278, 105" selected_image="playlist_selected.bmp"/>
</playlist_window>

```

1、迷你窗口的代码



```

<mini_window image="mini-player.bmp">
  <play position="177, 3, 271, 27" image="play_mini.bmp" />
  <pause position="177, 3, 271, 27" image="pause_mini.bmp" />

```

```

<prev position="153, 3, 247, 27" image="prev_mini.bmp" />
<next position="224, 3, 318, 27" image="next_mini.bmp" />
<stop position="201, 3, 295, 27" image="stop_mini.bmp" />
<lyric position="269, 17, 379, 27" image="lyric_mini.bmp" />
<minimode position="272, 3, 335, 20" image="minimode.bmp" />
<minimize position="255, 3, 317, 20" image="minimize.bmp" />
<exit position="289, 3, 351, 20" image="close.bmp" />
<icon position="3, 6, 19, 22" image="TTPlayer.ico"/>
<info position="27, 7, 142, 22" color="#ffffff" font="Tahoma" font_size="13"
align="left"/>
</mini_window>

```

2、音乐窗的代码（代码后面//部分为注释）



```

<browser_window position="540, 0, 810, 336" image="browser_skin.bmp"
transparent_color="#ff00ff">

```

```

<close position="452, 3, 469, 19" image="close.bmp" /> //关闭按钮位置

```

```

<backward position="112, 3, 133, 19" image="browser_backward.bmp" /> //后退按钮
位置

```

```

<forward position="148, 3, 169, 19" image="browser_forward.bmp" /> //前进按钮位置

```

```

<refresh position="186, 2, 207, 18" image="browser_refresh.bmp" /> //刷新按钮位置

```

```

<startup position="7, 458, 144, 473" ckbox_image="browser_startup.bmp" interval=4
color="#ffffff" font="SimSun" font_size="12" /> //多选框的位置，注意 X 坐标要包括后面的文字，interval 的值为多选框和文字之间的距离

```

```

<linktxt position="280, 457, 464, 472" color="#ffffff" font="SimSun" font_size="12" />
//连接文字区域，建议多留一些，以显示更多的文字内容

```

<browser position="9, 26, 464, 447" /> //html 网页位置，和“歌词秀”中的歌词显示范围性质一样
</browser_window>

二、下面看一下 Lyric.xml 这个配置文件

以下是歌词文字的设置，分别定义了字体类型，字体颜色，高亮颜色和背景颜色

```
<tplayer_lyric>  
  <Lyric  
    Font="-11,0,0,0,400,0,0,0,134,3,2,4,49,Tahoma"  
    TextColor="#008CC1"  
    HilightColor="#005489"  
    BkgndColor="#F4FBFE" />  
</tplayer_lyric>
```

三、下面看一下 Playlist.xml 这个配置文件

这个是播放列表窗口的文字设置，分别定义了字体类型，字体颜色，高亮颜色，第一背景颜色，数字颜色，时间颜色，当前选择颜色，第二背景颜色（可以和第一背景颜色一致）

```
<tplayer_playlist>  
  <PlayList  
    Font="-11,0,0,0,400,0,0,0,134,3,2,4,49,Tahoma"  
    Color_Text="#008CC1"  
    Color_Hilight="#005489"  
    Color_Bkgnd="#EAF5FA"  
    Color_Number="#005489"  
    Color_Duration="#005489"  
    Color_Select="#84CEF9"  
    Color_Bkgnd2="#EAF5FA"  
  />  
</tplayer_playlist>
```

四、下面看一下 Visual.xml 这个配置文件

这个是用来设置视觉显示的效果，我们先给大家介绍一下千千静听都提供哪几种视觉效果：

- 1、频谱分析
- 2、梦幻星空
- 3、视波显示
- 4、专辑封面
- 5、不显示视频效果

这 5 种情况只要在播放器主窗口上面点击鼠标右键即可切换



如图：

下面的配置文件是用来设置各种视觉效果颜色的，大家可以试试看，调出你喜欢的效果来

```
<tplayer_visual>
```

```
<Visual
```

```
    SpectrumTopColor="#FFFFFF"
```

```
    SpectrumBtmColor="#07F7FF"
```

```
    SpectrumMidColor="#8CDCFF"
```

```
    SpectrumPeakColor="#FFFFFF"
```

```
    SpectrumWide="1"
```

```
    BlurSpeed="3"
```

```
    Blur="1"
```

```
    BlurScopeColor="#07F7FF"
```

```
    TextColor="#FFFFFF"
```

```
    Font="-11,0,0,0,400,0,0,0,134,3,2,4,49,Tahoma"
```

```
/>
```

```
</tplayer_visual>
```

(C)打包成皮肤文件

好了，以上就是皮肤制作的整个过程啦，把上面的切图和配置文件制作好后，就可以打包成皮肤文件了，可以用 WinRAR 或 WinZIP 等压缩工具，以 WinRAR 为例，先按键盘上的 Ctrl +A 全选所有文件，然后将全部文件添加到压缩文件夹，压缩文件格式选择"zip"，压缩方式可选择"最好"，再点击"确定"就行了！

(D) 应用皮肤

将这个压缩文件复制到千千安装目录下的 Skin 文件夹，然后在"千千选项..."-"皮肤"中就可以选择应用皮肤了，或直接在主面板上点击右键-"选择皮肤"即可！

或许您已经了解皮肤的制作方法了，为了更深入的理解，我们增加以下部分的内容帮助您消化：

三、几个重要的属性解释

position: 这是众元素中最基本的属性之一，用于定义窗口背景及按钮的位置及大小，参数值格式为 "a, b, c, d"，这 4 个数值既固定了元素的位置也确定了其大小，其中(a, b)为左上角的坐标，(c, d)为右下角的坐标，c-a 就是长度，d-b 就是高度，坐标原点要分两种情况：如果是歌词秀、均衡器、播放列表这 3 个子窗口的 position，则是把主窗口的左上角作为它们的坐标原点；如果是各窗口内部按钮的 position，则是把对应的各窗口的左上角作为坐标原点，比如歌词秀窗口中的"关闭"按钮就是以歌词秀背景图片的左上角为原点，其它依此类推！

特别地：①播放列表中的 scrollbar 元素不需要定义 position 属性，它的位置会自动固定在 playlist 的最右边；②主窗口中的 progress、volume 元素及均衡器窗口中的 balance、surround、preamp 元素的 position 属性表示的是滑块能够移动的范围的坐标；③播放列表均衡器窗口中的 playlist 元素、歌词秀均衡器窗口中的 lyric 元素的 position 属性表示的是播放列表和歌词的显示范围，当播放列表窗口和歌词秀窗口改变大小时，这两个元素会自动跟着改变大小，但它们的四个边和窗口的四个边的距离就是通过这个 position 来体现的；

resize_rect: 歌词秀和播放列表窗口特有的属性，用于定义这两个窗口可被拉伸的部分，其参数格式同 position，代表的是当改变窗口大小时只有这个矩形框内的部分才会被拉长，在这个范围外的部分则不会变化，另外还有一个属性 resize_tile 是对应使用的，其参数值可以为 0 或 1，其中 0 表示在改变窗口大小时采用拉伸的方式，1 表示采用平铺的方式，该属性可省略不写，即使用默认值 0；

此外，歌词秀和播放列表窗口还有一个可选择的元素：title，可在有特定需要时(比如在改变窗口大小时标题保持居中等)使用，格式如下：

```
<title position="..." image="..." align="..." />
```

align: 用于定义元素的对齐方式，参数值分两种情况：一种是存在于主窗口中的 led、info、stereo、status 元素内，此时可以取值为 left、center、right，代表这些文字的缩进方式；第二种是存在于歌词秀窗口中的 title、close、ontop 元素内，播放列表窗口的 title、close、toolbar 元素内，此时可以取值为 left、center、right、top、bottom 等，代表当调整窗口大小时元素位置相对于边框移动，如果要同时设置垂直对齐和水平对齐方式，可以用英文加号将二者连在一起，比如"top+left"表示在垂直方向上顶部对齐、在水平方向上左对齐；

vertical: 存在于主窗口中的 progress、volume 元素，参数值可以为 true 或 false，其中取 true 时指滑块按垂直方向移动，取 false 时滑块按水平方向移动；

thumb_resize_center: 存在于播放列表窗口中的 scrollbar 元素内，用于定义 scrollbar 的 thumb 滑块中间可以进行平铺缩放的部分的大小，如果取值为 0，则代表在改变播放列表窗口高度大小时滑块进行不缩放；

thumb_resize_tile: 存在于播放列表窗口中的 scrollbar 元素内，作用与播放列表窗口和歌词秀窗口的 resize_tile 相同；

hot_image: 存在于播放列表窗口中的 toolbar 元素内，用于定义播放列表工具栏中当鼠标经过时的图片形态。此属性可选择，如省略不写的话程序会自动生成鼠标经过时的按钮状态；

eq_interval: 存在于均衡器窗口中的 equalizer_window 元素内，指 eqfactor 元素中 10 个波段的间隔大小(另：eqfactor 元素的 position 属性表示的是 10 个滑块中第一个滑块的位置，而其它属性对于所有 10 个滑块都有效)；

icon: 存在于主窗口中的 icon 元素内，用于自定义皮肤图标，必须将图标文件(*.ico, 16*16)放于皮肤文件夹中并一起打包压缩。此属性可选择，如省略不写的话则使用默认的程序图标；

left_top_color、right_bottom_color: 存在于歌词秀窗口中的 mini_border 元素内，用于定义在迷你模式下歌词秀窗口的左上边框和右下边框的颜色；

迷你窗口：

迷你模式其实是独立于主窗口外的另外一个皮肤，不过在迷你模式下省略了播放列表和均衡器窗口、简化了歌词秀窗口和主窗口。迷你窗口里的所有元素、属性及参数都是和主窗口一样的，它们都被包含于 <mini_window> 和 </mini_window> 中，相当于主窗口中的 <player_window> </player_window>；

迷你模式就是为了减小屏幕大小占用及简化按钮，故迷你窗口各按钮也要相应调整缩小，并省略部分不常用的按钮，比如音量调节等，一般只保留下"播放/暂停"、"停止"、"后退"、"前进"、"静音"、"图标"、"视觉效果"等即可。另外，迷你模式下的歌词秀窗口的位置和长度是固定的，高度则是和迷你模式的背景图片高度相同。

四、皮肤制作注意事项及技巧：

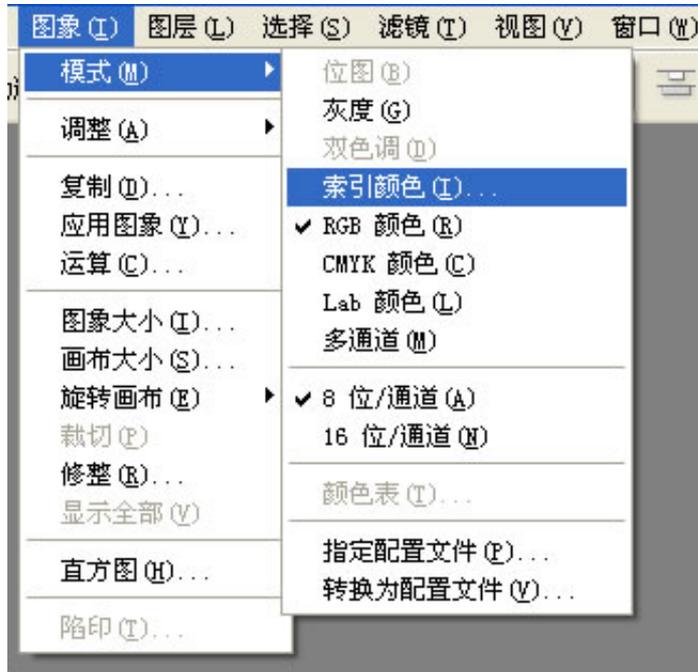
1. 压缩皮肤文件时，不是压缩整个文件夹，而是应该进入文件夹后按 Ctrl+A 全选所有文件，然后再添加到压缩文件(zip 格式)，否则皮肤无效；

2. 播放列表和歌词秀窗口的 position 属性定义了这两个窗口初始化时的大小，这个大小可以不是图片的实际大小。这两个窗口在初始化时就会按照 resize_rect 的规则拉伸窗口至所设置的大小。此外，这两个窗口最大可以拉伸到与屏幕同样大小，但最小只能缩小到与原始图片同样的大小，所以原始图片应该尽量画得小一些，这样可以方便用户把窗口缩成最小，同时还可以稍微减少图片及皮肤大小；

注：这个时候窗口上的按钮的 position 属性是按图片的实际大小来确定坐标的；

3. bmp 图片(尤其是几个面积较大的窗口背景图片)应尽量转换为 8 位的索引颜色，这样可以极大地减少图片及皮肤的大小，同时在应用皮肤时可以减少内存占用率。

具体的方法为：在 Adobe Photoshop 中，打开 RGB 模式的图片，然后点击“图像”-“模式”-“索引颜色”，再保存即可！注：如果有透明色时要注意两点：①在填充透明色时，一定不要选“容差”；②转换时一定要选中“保留原实际颜色”，以防止填充的透明色被改掉。



4. 当按钮很小的时候，不要将其透明，而是和背景图片连在一起！因为我们知道，皮肤中透明的部分是不感应鼠标动作的，因此当按钮比较小的时候如果中间有很小的缝隙，鼠标移动到上面时就会乱跳，不便于点击操作！如图所示：



5. 在制作皮肤时最好做一部分预览一下效果，这样有问题的地方可以及时修改过来

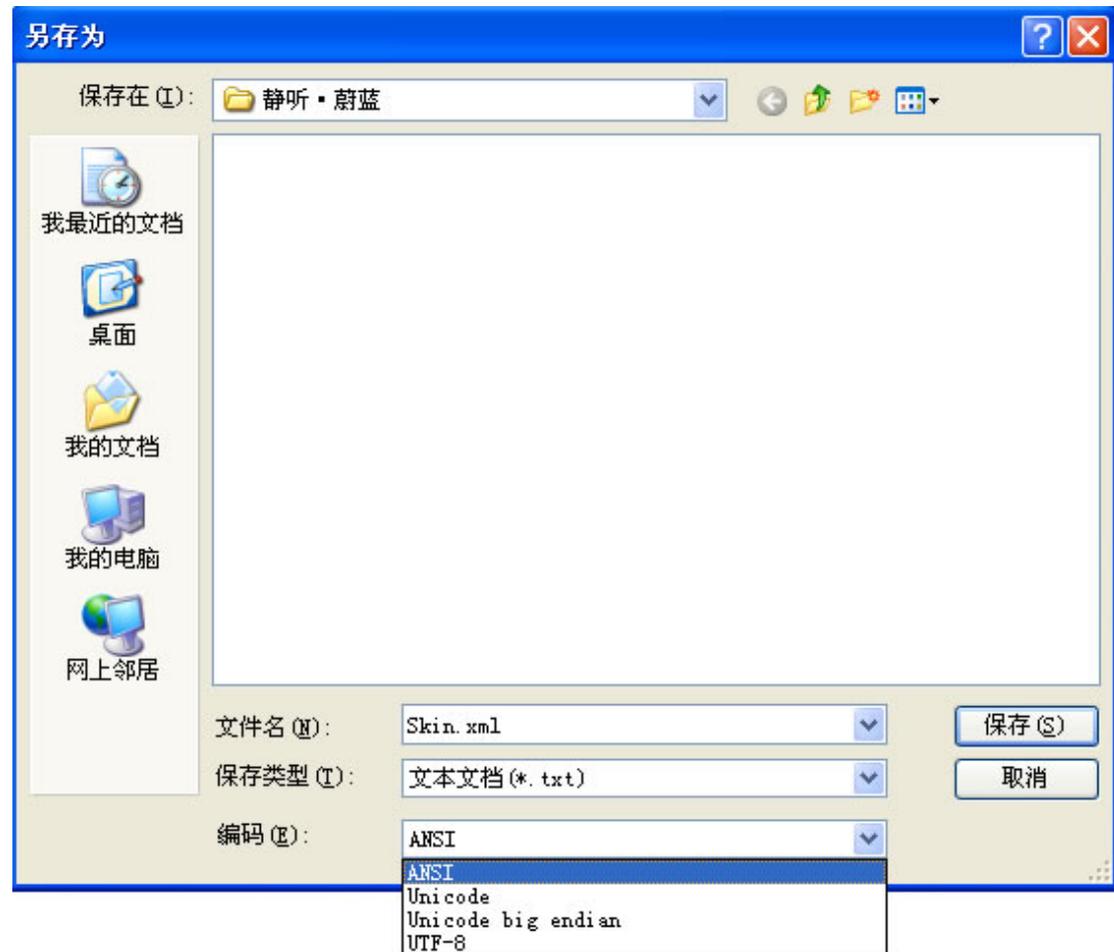
6. 如果不想在主窗口或迷你模式中显示千千静听的图标，可以将图标的 position 设置为超过窗口大小，比如“500, 500, 516, 516”；

7. 在主窗口中的 led 元素(即播放时间)有两种显示方式，一种是已播放的时间，一种是未播放的时间，鼠标点击即可在两种显示方式之间切换。后者比前者多了一个负号，因此主窗口上应至少给 led 留出 6 位的空间，以防止在显示剩余时间时数字覆盖到面板上的其它部分而影响美观；led 元素使用的图片必须是 12 张同样大小的图片排在一起，分别代表 0-9 十

个数字、冒号和减号(可以使用透明色);

8. 如果不想显示某个窗口或者各窗口上的某些元素, 把相应的元素代码全部删除即可;

9. 如果是用记事本打开 XML 文件的, 那存储的时候尽量编码设置成 ANSI; 如下图所示:



10、关于坐标定位的小技巧:

大家注意一下坐标定义的问题, 拿一个 20*20 的小图为例, 它的位置是 150, 30, 170, 50, 大家来看这个坐标, 一定认为没有问题, 但实际效果会显示不全, 为什么呢, 因为坐标的 X 轴有一个小小的误差问题, 要多加 2 像素才可以哦, 正确的定位是 150, 30, 172, 50。大家在制作时注意一下。

制作千千皮肤的基本方法就只有这么多, 但是技巧远不止这些, 大家可以在制作的过程中不断实践摸索, 并发挥自己的创造力, 设计出各种有创意的皮肤来!

鸣谢: 发呆的猴子, 很高很瘦