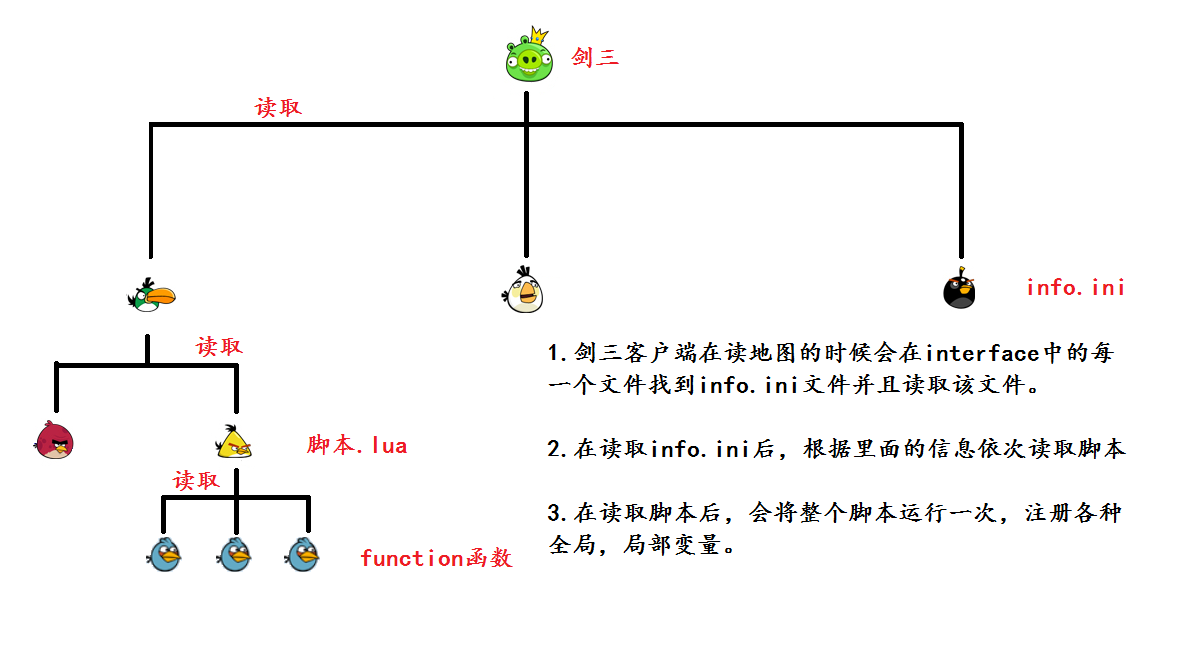
一个插件最基本必须包含两个项目：  
              1.插件信息文件info.ini  
              2.插件主文件xxxxxxx.lua  
        
       其他的窗口配置文件、自定义图片、图片配置文件、声音文件等都属于附加上去的功能，使得插件功能更加强大、丰富、美观。  
  
       为什么插件的名字都很奇怪？像是什么se\_XXXXXX，Xu\_XXXXXX，ChenYu\_XXXXXX，Ly\_XXXXXX，BF\_XXXXXX，Moon\_XXXXXX等等  
  
       首先来了解下剑三是怎么读取插件的，以前B叔画过一幅图（亲笔手绘图啊，虽然很糟糕）不过后来被我弄丢了，只好自己再画一个：  
  
  
        是不是发现了什么？  
        对，在这个读取过程中，文件夹跟脚本的名字并没有起到命名的作用。  
  
        那为什么文件夹名字各有千秋呢？  
  
        1.命名文件夹要做到见其名而知其意  
        2.避免和其他插件产生冲突  
  
       由上图可见，如果文件夹在重名的时候，后面读取的会覆盖前面读取的，这不是我们想见到的事情。

通常我们为了编写和管理插件的方便，有这样一些约定俗成的规矩。  
  
1、为你的插件新建一个文件夹，文件夹的名字尽量做到见即知意。  
2、 为了避免和别的插件冲突，在你的文件夹前加一个自己的前缀，比如Const\_HeadEx。  
3、将你的插件入口程序和info.ini文件放到同一目录下。  
4、在你的程序文件第一行首先定义一个table，这个table的名字是你插件的名字，用于存储所有与你插件相关的信息。  
5、所有的函数、属性、都写在自己定义的table中。  
6、在程序文件的最后一行调用你的入口函数。  
7、将界面配置文件和对应的程序文件取相同的文件名。

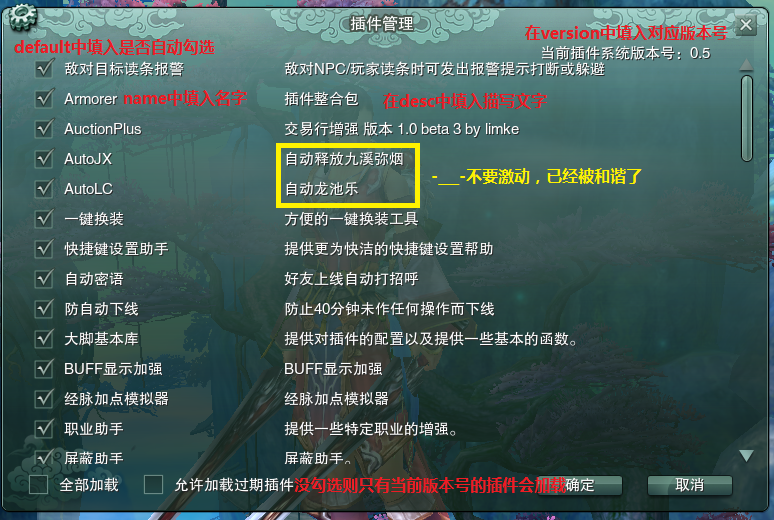
       从上图知道info.ini这个文件是给起到给系统导航的作用，所以它的内容至关重要，见下面由chathelper中提取的代码：

1. [ChatHelper]
2. name=聊天助手
3. desc=by 陈雨 [小队/团队说话泡泡 职业着色 聊天过滤  时间显示 密语聊天泡泡/记录 信息定时发布 密语群发功能等]
4. version=0.5
5. default=1
6. lua\_0=Interface\ChatHelper\WhisperPanel.lua
7. lua\_1=Interface\ChatHelper\BubblePanel.lua
8. lua\_2=Interface\ChatHelper\ChatHelper.lua
9. lua\_3=Interface\ChatHelper\BlackList.lua

*复制代码*

       [ChatHelper]：用于系统辨识  
       name：插件的名字，方便用户进行辨识、加载或禁用  
       desc：插件描述文件，简单介绍插件功能  
       version：版本号，如果版本号与系统不符，是不会加载该插件的，除非勾选了加载过期插件

当我们指定的系统插件版本号低于右上角显示的系统版本号时，插件将变成灰色，这表示官方对插件的一些设置、开放的接口作出了调整，可能你所编写的插件在该版本下不能正常使用，我们可以通过勾选下面的允许加载过期插件来强制运行它，但是不一定可以正常工作哦。

        
       default：默认插件是否加载，1为加载，0为不加载  
       lua\_0/1/2/3：插件脚本位置，为系统导航  
  
见下图：  
原理介绍  
  
  
Lua跟Ini的关系  
  
好多人都说界面插件都是怎么运作的，然后得到的答案大多就是INI结合LUA的操作，于是百思不得其解这两者的关系是怎么建立起来的。简单的来说，INI 文件只是外表，而真正的核心还是LUA文件（- -废话，插件都是LUA写的）。这么说可能还不是太容易理解，举个简单的例子吧。  
  
如果剑三这个游戏是一个房间，那么进入房间我们所看到的就是各种家具，界面就是这些家具所组成的。假设INI文件是桌子，那LUA文件的作用就是桌子该放 在哪里，上面要放什么，放多少，物品的大小，放上去是否能够移动等等的各种属性。没有INI就没有桌子，那么怎么摆放桌子上的物品也就没效果了，反之，有 了桌子不知道怎么摆也是行不通的。  
  
在第一篇里面说到了剑三的加载方式，然而那只是加载LUA并不是INI的加载，而后我们写了第一个插件，这个插件的功能是加载INI与报错！  
  
OpenWindow("窗口文件路径","窗口名字")  
  
INI中的Frame有个ScriptFile这个属性，这个是指INI所运行的脚本文件路径，再加上上面这一个简单的API就将Lua跟Ini连接了起来。  
  
当窗口打开后或者说Ini被加载后，房间里有了桌子才能开始摆放调整。可桌子上能够放那么多东西，怎么知道动的是哪个。这就涉及到另一个命令  
  
Lookup()  
  
简单来说这个命令就是在房间里寻找所需要的物品，找到后才能对该物品进行各项调整。  
  
  
Ini的架构（常用控件）  
  
Frame（窗体）  
每个Ini文件的架构中都有一个窗体，这个窗体可以理解为前面说到的家具骨架，是最顶层的控件必须要有的东西，所有其他的东西都是在Frame里面的，一般来说一个Ini里面就是只有一个Frame，如果有两个Frame的话完全可以拆分为两个Ini。  
  
Handle（容器）  
类似于文件夹这种东西，可以看成在桌子上的杯子，碗啊之类的东西，主要用于管理以及几个控件一起调用，比如说桌子上有很多弹珠跟橡皮筋，你想全部拿起来，当然是放在盒子里面一起拿比较简单啦。  
  
Image（图片）  
图片控件，用于专门的图片操作，当然也可以在其他控件里面加载图片，但是一些九十度转角这类的命令只对图片控件有效果  
  
Text（文字）  
文字控件，用于文字的字体设置等，只要是文字用这个就没错。  
  
Box（盒子）  
盒子控件相对于来说比较简单也很方便，比如说背包格子、技能条的格子那些都是简单的盒子组成的，有一个特点是它可以显示转圈百分比（技能冷却时的那种效果），除此之外，就是比文字加图片构成的盒子来说要简单操作。  
  
Shadow（影子）  
用于绘制一些简单的图像，例如之前大脚的气场范围以及海鳗的鹰眼显示哪些都是用影子控件加上Lua绘制出来的。  
  
Animate（动画）  
用于一些简单动态显示，如冷却CD的闪烁，成就完成时的盖章。  
  
TreeLeaf（树节点）  
树节点控件，常见于任务面板的下拉菜单，但对于多数插件来说可以说是完全用不到的一个控件，所以在这篇中没有介绍如何使用树节点，如有兴趣可以自行查看任务面板的Lua文件  
  
WndScrollBar（滚动条）  
用于列表类，非常常见的一个控件。  
  
WndButton（按钮）  
各种按钮，无论是关闭还是确认等。  
  
WndCheckBox（复选框）  
见于各种选择配置类，单选多选，最常用于用户对插件进行配置。  
  
WndEdit（输入框）  
比GetUserInput()更加高级一点的获取用户输入的控件  
  
WndPage（标签页）  
参考人物属性下面的标签  
  
WndPageSet（标签页面集）  
配合标签页使用，先得建立页面集再建立标签页。  
  
  
Frame的基本属性  
  
.\_WndType  
控件类型，用于标注是什么控件  
  
.\_Parent  
父级，层级问题，简单来说就是是谁的下属  
  
Left跟Top  
控件的X轴跟Y轴，用于定位窗体的左上角初始坐标从哪里开始，（0,0）即是整个屏幕的左上角  
  
Width跟Height  
控件的宽跟高，决定大小  
  
DragAreaLeft、DragAreaTop、DragAreaWidth、DragAreaHeight  
拖动的范围，X轴，Y轴，宽跟高  
  
AnimateStartPosX、AnimateStartPosY  
动画开始位置，X轴，Y轴  
  
AnimateEndPosX、AnimateEndPosY  
动画结束位置，X轴，Y轴  
  
AnimateTimeSpace  
动画时空，即表现方式  
  
AnimateMoveSpeed  
动画移动速度  
  
IsCustomDragable  
是否能被拖动，1为能被拖动，0为不能  
  
ScriptFile  
脚本文件路径，用于连接Lua跟Ini。

控件命令  
  
  
作为一个控件所受到的命令就是怎么表现，基本上没什么好说的。  
  
常用命令  
  
通用  
  
Show() --显示  
Hide() --隐藏  
SetRelPos(x轴，y轴) --设定物件相对坐标,(0,0)即其父级的起始坐标点  
GetRelPos() --获取相对坐标  
SetAbsPos(x坐标,y坐标) --设定绝对坐标,(0,0)即屏幕左上角  
GetAbsPos() --获取绝对坐标  
SetSize(x大小, y大小) --设定物件大小  
GetSize() --获取大小  
SetPosType(位置) -POSITION\_BY\_CUSTOM,POSITION\_LEFT\_BOTTOM,POSITION\_LEFT\_TOP,POSITION\_LEFT\_CENTER,POSITION\_TOP\_LEFT,POSITION\_TOP\_RIGHT,POSITION\_TOP\_CENTER,POSITION\_RIGHT\_TOP,POSITION\_RIGHT\_BOTTOM,POSITION\_RIGHT\_CENTER,POSITION\_BOTTOM\_LEFT,POSITION\_BOTTOM\_RIGHT,POSITION\_BOTTOM\_CENTER,  
GetPosType()  
IsVisible() --是否可见,true为可见  
GetName() --获取名字  
SetName() --设定名字  
SetTip() --设定帮助tip  
GetTip() --获取帮助tip  
EnableScale(布尔) --允许缩放,true为允许  
Scale(x轴缩放比例,y轴缩放比例) --就当前大小缩放  
LockShowAndHide() --锁定显示与隐藏,1为锁定  
SetAlpha(参数) --设定透明度,0-255,255为不透明  
GetAlpha() --获取透明度  
GetParent() --获取父级  
GetRoot() --获取根级,即frame  
IsValid() --是否有效  
Clear() --清除物件  
Enable(true) --是否启用  
EnableDrag(true) --是否能拖动  
FormatAllItemPos() --更新控件下所有物件位置  
Lookup("物件名字") --找寻物件  
SetDragArea(起始点x，起始点y，结束点x，结束点y) --设定拖拽面积  
SetPoint("CENTER", 0, 0, "CENTER", 0, 0) --居中显示，可替换成TOPCENTER（上居中），BOTTOMCENTER（下居中），BOTTOMRIGHT（右下），BOTTOMLEFT左 下），TOPRIGHT（右上），TOPLEFT（右下），最后两个参数为X轴跟Y轴  
  
Text  
  
AutoSize()  
GetTextLength()  
SetFontColor(r,g,b)  
SetFontScale(比例)  
SetLimit(长度上限)  
SetText("内容")  
  
Image  
  
FromTextureFile("路径")  
FromIconID(图标ID)  
FromUITex("路径",帧数)  
SetFrame(帧数)  
SetRotate(参数）1-255对应0-360  
SetPercentage(小数)  
  
Box  
  
SetBoxIndex(数字) --设定序号  
GetBoxIndex() --获取序号  
SetObject(1,0) --设定box内的物件  
GetObject() --获取box内的物件  
GetObjectType() --获取box内的物件类型  
GetObjectData() --获取box内的物件资料  
ClearObject() --清楚box内的物件  
IsEmpty() --box是否为空  
EnableObject(布尔) --是否启用box内物件,true为启用  
IsObjectEnable() --box内物件是否启用  
EnableObjectEquip(布尔)  --是否启用物件可装备性  
IsObjectEquipable() --物件是否可装备  
SetObjectCoolDown(数字) --物件是否可转圈,1开启,0关闭  
IsObjectCoolDown() --物件是否已冷却完毕  
SetObjectSparking(数字) --设定物件闪烁类型  
SetObjectInUse() --设定物件在使用时类型  
SetObjectStaring() --设定物件空闲时类型  
SetCoolDownPercentage(转圈百分比) --设定冷却百分比  
GetCoolDownPercentage() --获取冷却百分比  
SetObjectIcon(图标ID) --设定图标  
GetObjectIcon() --获取图标  
ClearObjectIcon() --清楚图标  
SetOverText(0,"内容") --设定最上层的显示字  
GetOverText() --获取最上层的字  
SetOverTextFontScheme(0, 样式) --设定最上层字的字体  
GetOverTextFontScheme() --获取最上层字的字体  
SetOverTextPosition(0, ITEM\_POSITION.RIGHT\_BOTTOM) --设定最上层字的位置，左:Left,右:right,上:top,下:bottom  
GetOverTextPosition() --获取最上层字的位置  
  
  
上面这些都是比较常用的功能，还记得前面几篇说过，平时写插件的时候养成收集的习惯，这会大大的加快以后的速度，同时也可以将收集到的信息分享给他人。