Widget 开发指南

了解了盛大圈圈的形态之后,我们从一个例子开始逐步深入 Widget 的开发。

1. Hello Beijing!

i. 创建第一个 Widget

简单的 Widget 只需要很少量的代码和文件。例如我们以制作一个打开奥运会官网的 Widget 作为起始。

1) 在圈圈安装目录下的 Widgets 目录下创建一个 子目录 Beijing2008

2) 创建一个名为 gadget.xml 的文件,复制入下列代码,并保存

```
<?xml version="1.0" encoding="utf-8"?>
<gadget>
   <name>北京2008</name>
   <namespace>snda.igw</namespace>
   <version>1.0.0.0</version>
   <author name="路人甲">
       <info url="http://www.beijing2008.cn/" text="第29届奥林匹克运动会"/>
   </author>
   <copyright>© 2008</copyright>
   <description>北京2008年第29届奥林匹克运动会</description>
   <icons>
       <icon height="64" width="64" src="icon64.png"/>
       <icon height="32" width="32" src="icon32.png"/>
   </icons>
    <hosts>
       <host name="sidebar">
           <base type="html" apiVersion="1.0.0"</pre>
src="http://www.beijing2008.cn/" width="752" height="459" clientwidth="724"
clientheight="404" xul="WebBrowser.xml" />
           <permissions>full</permissions>
           <platform minPlatformVersion="1.0"/>
            <defaultImage src="drag.png"/>
       </host>
   </hosts>
    <guid>{DF2FDE45-5847-401B-9C27-6965CD25F23E}
</gadget>
```

3) 制作两个奥运会徽的 png 格式图标,尺寸是 32*32 和 64*64,命名为 icon32.png 和 icon64.png;还可以顺手做一个个人的 logo。最后的文件结构如下图:



Beijing2008.zip

ii. 测试 Widget

制作好的 Widget 是易于测试的。虽然盛大圈圈是集成到游戏中运行,但是借助自带的测试环境,一样可以方便的进行验证。

 运行圈圈安装目录下的 WidgetGraphicTest.exe, 点取工具栏上的 "+" 按钮,可以看到 我们刚才制作的 "北京 2008" Widget 已经被陈列出来了;



- 点击添加"北京 2008"下的"添加"按钮,可以看到工具栏上已经出现了"北京 2008" 的图标,点击图标即可执行我们刚才的设定的逻辑
- 可以看到界面中央出现一个浏览器窗口,自动打开奥运会官网,并且能够自如的操作; 同时还具有半透明等诸多特效。测试顺利通过!



iii. Gadget.xml 描述文件

Gadget.xml 是 Widget 的描述文件,它记录 Widget 的身份、界面、参数等信息。 包含完整 xml 元素的信息示例和说明如下:

```
<?xml version="1.0" encoding="utf-8"?>
<!--必须, <gadget>不包含属性, 仅仅包含用于定义widget的子元素-->
<gadget>
   <!--widget的名字,显示在工具集中,以及工具栏的提示信息上-->
   <name>Widget名字,必须</name>
   <!--名称空间,备用-->
   <namespace>snda.igw</namespace>
   <!--版本号,必须,规则是四段式,主版本.次版本.修订号.编译号,每个字段限在4位整数。
圈圈针对同名的Widget会自动加载高版本-->
   <version>1.0.0.0</version>
   <!--作者信息,可选; name,开发商名称-->
   <author name="Shanda Online">
      <!-- 开发商Logo信息; src, 文件名称-->
      <logo src="logo.png"/>
   </author>
   <!--版权信息,可选-->
   <copyright>© 2008</copyright>
   <!--Widget 描述, 可选-->
   <description>盛大个人中心</description>
   <!-- 图标信息, 可选-->
   <icons>
      <!--图标; height, 高度像素; width, 宽度像素; src, 图标文件名-->
```

```
<icon height="64" width="64" src="iconPerson.png"/>
      <icon height="32" width="32" src="icon.png"/>
   </icons>
   <!-- 盛大应用ID,指Single Sign ON验证中的AppID,外部开发者可忽略-->
   <applicationSNDAID>IGWXXX</applicationSNDAID>
   <!--是否单实例,默认为true,如果允许同时开启多份实例,需要显式指定为false;除非
特殊情况,请不要指定。-->
   <single>true</single>
   <!--宿主信息,必须;此元素是多个<host>的容器,<host>是<hosts>的子元素-->
   <hosts>
      <!--Widget的运行宿主,必须; name, 宿主名, 目前填"sidebar", 以后可能会扩充-->
      <host name="sidebar">
          <!--标识widget的文件类型和版本信息; -->
          <!--type,应用类型,目前多为"html"; apiVersion,保留; src,应用地址,
可以是本地网页,也可以是远程的;-->
          <!--LoginMode,登录模式,目前支持SSO和CAS两种模式,外部用户可暂时忽略;
LoginMethod, 登录方式, -->
          <!--width,默认宽度; height,默认高度; xuL,采用何种皮肤和外框,常规填
"WebBrowser.xml"-->
          <base type="html" apiVersion="1.0.0"</pre>
src="http://192.168.90.36/SD0.COM/Index.aspx" loginMode="SS0"
loginMethod="header" width="200" height="200" xul="WebBrowser.xml"/>
          <!--保留, 设为"full"-->
          <permissions>full</permissions>
          <!--Widget对圈圈的版本要求; minPLatform,最低版本号-->
          <platform minPlatformVersion="1.0"/>
          <!--在从小工具备选箱用鼠标拖到工具栏时所显示的Logo-->
          <defaultImage src="drag.png"/>
      </host>
   </hosts>
   <!-- 唯一GUID标识,此标识在发布时统一分配,开发时不用指定-->
   <guid>{B4F41D48-62C3-4921-8CA6-9810597ED56A}
</gadget>
```

i. 创建用户界面

i. 窗口与图标

盛大圈圈在游戏内以工具栏或侧边栏形式展现。

每个应用需要准备 32*32 和 64*64 两种大小的图标一套。

32*32 尺寸的图标作为显示在工具栏上的图标,64*64 尺寸的作为陈列在工具集中的图标。文件名通常是 icon32.png 和 icon.png,也可以是自定义的文件名,在 gadget.xml 中描述出来。

例如:



如果小工具固定于工具栏中,则将其视为处于停靠状态。如果将其拖出工具栏,则其处 于浮动或未停靠状态。

处于停靠状态的 Widget 最大宽度为 130 像素。处于未停靠状态的小工具在理论上没 有宽度限制,可以设计更宽的视图。但从游戏用户角度,我们规定 Widget 页面需在 800*600 分辨率下设计,单一 Widget 界面限定为最大尺寸 700*360。

ii. 图片与背景

常规的网页都能够在圈圈中很好的显示,同时圈圈Widget支持更多的特色界面设计机制。 ▶ 支持网页背景透明

透明色定义: #000100 透明色

网页中指定背景色为 #000100 即可以显示为透明。

▶ 图片半透明

可以在网页中使用 32 位 PNG 格式图像来显示半透明图像效果。注意 PNG 在各 IE 版本 中透明效果的区别,避免可能出现的显示效果不一致问题。

IE5 及以上:可以使用 VML 支持 PNG 透明,如果没有使用 VML 即使使用 PNG 图片 也是不透明的;

IE7:本身即完全支持 PNG 透明。

示例:

VML 透明



```
<html xmlns:v="urn:schemas-microsoft-com:vml">
<head>
<meta http-equiv=Content-Type content="text/html; charset=gb2312">
<link rel=File-List href="Doc1.files/filelist.xml">
<style>
v\:* {behavior:url(#default#VML);}
.STYLE3 {font-size: 24px; font-weight: bold; }
</style>
</head>
<body bgcolor="#000100">
<v:image src="docked_windy.png"
style="position:absolute;top:0;left:0;width:130;height:67;z-index:-1"/>
<v:image src="docked_windy.png"</pre>
style="position:absolute;top:0;left:130;width:130;height:67;z-index:-1"/>
<span class="STYLE3"><font</pre>
color="red">SNDA</font></span>&nbsp;&nbsp;&nbsp;
class="STYLE3"><font</pre>
color="#FF00FF">EZPLAY</font></span>&nbsp;&nbsp;&nbsp;&nbsp;
class="STYLE3"><font color="yellow">2007</font></span>
</body>
</html>
```

▶ 使用预制的皮肤透明

一般情况下,可以使用圈圈提供的预制的皮肤来达到和 Vista 类似的玻璃边框效果。并可以设置内容区域的透明度。以此到达和游戏良好兼容的效果。

使用方法是在 Widget 的 xml 描述中使用 xul 属性来描述需要使用的外框皮肤。

```
示例:
<?xml version="1.0" encoding="utf-8"?>
<gadget>
   <name>起点文学</name>
    <namespace>snda.igw</namespace>
    <version>1.0.0.0</version>
    <author name="SNDA Corporation">
       <info url="http://www.qidian.com" text="起点中文网"/>
       <logo src="logo.png"/>
    </author>
   <copyright>© 2008</copyright>
    <description>起点文学-小说原创门户--起点中文网</description>
    <hosts>
       <host name="sidebar">
                                   type="html"
                    <base
                                                        apiVersion="1.0.0"
src="http://igwgame.qidian.com/default.aspx" loginMode="SD0" width="752"
height="459" clientwidth="724" clientheight="404" xul="WebBrowser.xml" />
           <permissions>full</permissions>
           <platform minPlatformVersion="1.0"/>
           <defaultImage src="drag.png"/>
       </host>
```

```
</hosts>
    <guid>{DF2FDE45-5859-401B-9C27-6978CD25F23E}</guid>
</gadget>
```

Widget 实际运行的界面效果如下:



iii. Select 元素的处理

常规的 HTML Select 元素会在圈圈中显示异常,因此建议不要直接使用 Select 下接框作为选择,请使用附件中的替代方案



示例:

```
<style>
SELECT {
behavior:url('Select.htc');
}
</style>
<select>
<option>1</option>
<option>2</option>
<option>3</option>
<option>4</option>
</select>
```

注意: 此方案没有 onchange 事件

iv. Widget 中打开新窗口

圈圈 Widget 中的网页可以打开新的浏览器窗口,默认打开新窗口为一个游戏内全屏窗口

示例:

充值

也可以使用 System.Content 对象的 OpenIE 方法来打开新窗口。所有打开的新窗口在所 属 Widget 被关闭时将一起被销毁

以下为具体方法使用说明:

1.	Openl	E

RESULT _stdcall OpenIE([in] BSTR name, [in] BSTR url, [in] long left, [in]
ong top, [in] long width, [in] long height, [in, optional, defaultvalue("")]
<pre>STR mode, [out, retval] IDispatch ** htmlWindow);</pre>
≶数说明:
name: 名称
url: 新窗口 url
left: 新窗口所在座标
top:
width:
height:
mode: 新开窗口模式, 默认为打开一个 widget 内部窗口
F

```
示例:
```

```
<script>
win = System.Content.OpenIE("abc","test.html",200,200,0,0);
</script>
```

2. 透明边框

如果期望用此方式打开的窗口带有边框,则需要如下使用。

```
<script>
win=System.Content.OpenIE("abc","test.html",200,200,0,0,
"xul=WebBrowser.xml");
;alignparent=auto
</script>
```

3. 新窗口自动对齐原窗口

```
<script>
System.Content.OpenIE("abc","test.html",200,200,0,0,
"xul=WebBrowser.xml; alignparent=auto"
</script>
```

v. Widget 中使用 Flash

Widget 界面中可以嵌入 Flash 动画,来实现更好的界面效果。在使用 Flash 时,强热建议使用透明(无窗口)模式。

```
示例:
```

vi. 处理 停靠/浮动 事件

当 Widget 被拖放停靠到工具栏上时, 会触发 System.Widget.onDock 事件。 当 Widget 从工具栏上拖出来时, 会触发 System.Widget.onUndock 事件。 开发者可以在这两个事件触发时, 进行调整界面布局, 改变窗口尺寸的操作。

示例:

```
<script>
System.Widget.onDock=DockStatCheck;
System.Widget.onUndock=DockStatCheck;
function DockStatCheck(){
    if (System.Widget.Docked){
        t.innerHTML="Docked";
        window.resizeTo(130,130);
    }
    else
    {
        window.resizeTo(240,200);
        t.innerHTML="Not Docked";
    }
}
</script>
```

ii. 开发规范与 API

圈圈提供了一系列的系统 API 供 Widget 调用,同时 Widget 需要遵守为了在游戏环境中 良好运行而做出的限定。

vii. 圈圈提供的系统对象

圈圈提供了一组 API 供 Widget 调用, Widget 开发者可以灵活利用这些 API 与圈圈进行 沟通,完成常规脚本难以实现的功能。

对象列表:

对象名称	功能定义、方法			
System.Debug	调试对象			
	方法: System.Debug.outputString(strOutputString)			
	输出调试信息,使用DebugView工具查看			
	DebugView.下井地址			
	Debug view下致地址:			
System Diagnostics	<u>http://technet.nncrosoft.com/en-us/sysinternais/00890047.aspx</u> 诊断对象			
System.Diagnostics.EventLog	▲ 「			
~;	小元ず[[日心/]]多			
	方法: System.Diagnostics.EventLog.writeEntry(strEventLogEntry])			
	记录事件日志,在系统事件查看其中查看			
System.Environment	系统环境变量			
	方法: strValue = System.Environment.getEnvironmentVariable(strEnvVar)			
	获取系统和用户环境变量			
	同姓 1·N			
	周性: machineName			
System Widget Settings	· 获取平地机器名 - 东取Withor的职策			
System. widget.Settings	行政Widget的距直项			
	方注, strSetting = System Gadget Settings readString(strName)			
	读取一个字符串键值			
	方法: System.Gadget.Settings.write(strName, varValue)			
	保存一个不确定类型的键值			
	方法: System.Gadget.Settings.writeString(strName, strValue)			
	写入一个字符串键值			
System.Sound	系统音效			
	方注, System Sound been()			
	游戏····································			
	方法: System.Sound.playSound(strSound)			
	播放wave文件			
System.Time	获取当前时区的时间			
	方法: strLocalTime = System.Time.getLocalTime(oTimeZone)			
	秋 取当前时间			
	国州 aumontTimeZone			
	周任: Current I me Zone 本田 当前时区			
System Time TimeZone				
System Widget	Widget的基础事件 方注 届性			
	事件: System.Widget.onDock = handler			
	当Widget被拖放停靠到工具栏上时触发			
	事件: System. Widget.onUndock = handler			
	当Widget从工具栏上拖出来时触发			
System.Content	圈圈内容控制			
	力法: System.Content.RegisterLIB(strName)			

	方法: System.Content.CreateComObject(strName)		
	方法: System.Content.CreateActiveX(strName)		
	Widget中打开新窗口 使用COM组件		
System.Online	在线行为相关,保留		

viii. WEB 服务器如何识别浏览器是圈圈

如果您的WEB服务器需要针对圈圈进行特殊的处理,这需要识别浏览器的UserAgent。 圈圈Widget通过UserAgent中附加信息来标识自己:

IGW 1.0 (IGW 版本号)

示例:

```
Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; Maxthon; Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1) ; InfoPath.1; .NET CLR 1.1.4322; .NET CLR 2.0.50727; IGW 1.0 (0.9.0.110))
```

ix. 使用 COM 组件

尽量不要使用常规脚本中的 new ActiveXObject() 方法,而使用以下替代方法:

System.Content.CreateComObject System.Content.CreateActiveX

示例 1:

System.Content.CreateComObject("SDSpeakBox.SDGroupSpeaker"); System.Content.CreateActiveX("{D27CDB6E-AE6D-11cf-96B8-444553540000}")

示例 2:

```
try
{
    g_groupSpeaker =
System.Content.CreateComObject("SDSpeakBox.SDGroupSpeaker");
    if (isNaN(g_groupSpeaker)) {
        if (System.Content.RegisterLIB("com\\sdspeakbox.dll"))
            g_groupSpeaker =
System.Content.CreateComObject("SDSpeakBox.SDGroupSpeaker");
    }
}
catch (e)
{
    g_groupSpeaker = new ActiveXObject("SDSpeakBox.SDGroupSpeaker");
}
```

x. 注册 COM 组件

某些应用需要使用外部组件来扩充功能,如果外部组件需要注册,可以使用如下方法: System.Content.RegisterLIB(LibName) LibName 为相对此 Widget 存放目录的文件名

示例:

```
try
{
    g_groupSpeaker = System.Content.CreateComObject("SDSpeakBox.SDGroupSpeaker");
    if (isNaN(g_groupSpeaker)) {
        if (System.Content.RegisterLIB("com\\sdspeakbox.dll"))
            g_groupSpeaker =
    System.Content.CreateComObject("SDSpeakBox.SDGroupSpeaker");
    }
}
catch (e)
{
    g_groupSpeaker = new ActiveXObject("SDSpeakBox.SDGroupSpeaker");
}
```

xi. 浏览器的功能限制

为了适合于游戏内使用,圈圈对浏览器的一些默认行为和功能设定做了处理,在开发时 需要留意,如下:

- 1. 禁止弹出对话框
 - 包括 confirm/prompt 弹出的对话框,只部分支持 alert 效果。
- 2. 屏蔽右键弹出菜单(但不屏蔽右键消息)
- 3. 不接受拖拽
- 隐藏边框/隐藏滚动条 如果一定要用边框/滚动条,请使用<div>标签设置 style 来做出边框/滚动条
- 5. 允许使用Windows主题
- 6. 自动忽略脚本执行错误
- 7. 禁止文件下载(IE5.5及以上有效)
- 8. 禁止Ctrl+快捷键(N:新建窗口;F:查找;P:打印)

xii. 处理脚本异常

HTML 类型的 Widget 默认忽略所有脚本异常,XUL 类型的 Widget 脚本中发生异常也不会弹出对话框。这样不会给用户带来干扰,但编写 Widget 时调试会不太方便,无法从异常对话框中看到异常信息,圈圈自动将脚本异常信息通过 OutputDebugString 输出,因此可以使用 DebugView 等工具来查看异常信息

xiii. 游戏基本数据获取

在 Gadget.xml 的 hosts/host/base 标签的 src 属性中所指定的该 Widget 所访问的地址中 可以通过一些简单的转义标记获得一些基本的游戏信息,标记如下:

\$gid\$	游戏的 AppID 编号
\$gname\$	游戏的名称
\$ga\$	游戏当前区的名称
\$gaid\$	游戏当前区的 ID

\$gs\$	游戏当前游戏组的名称	(注意:	游戏组的信息不一定准确)
\$gsid\$	游戏当前游戏器组的 ID		
\$gnick\$	游戏当前玩家的昵称		
\$sex\$	游戏当前玩家的性别		

iii. 发布 Widget

Widget 制作测试完成后,就可以打包进行发布了。发布时,需要提供 Widget 目录的压缩包及产品描述,发送到大赛指定的接收方。