

# allWebPlugin 技术白皮书

V2.0

allWebPlugin中间件

# 目录

allWebPlugin 技术白皮书 .....	1
一、 开发背景 .....	5
二、 软件功能 .....	5
三、 软件特点 .....	5
四、 运行环境 .....	6
五、 浏览器插件集成步骤 .....	6
a) allWebPlugin 与浏览器、插件关系图 .....	6
b) 引入相关 allWebPlugin JS .....	6
c) 添加页面元素 .....	7
d) 创建插件对象 .....	7
e) 插件接口调用 .....	7
f) 插件事件响应 .....	7
六、 接口说明 .....	8
1. 全局接口 .....	8
1) awp_IsInstall 方法 .....	8
2) awp_CreatePluginFromIE 方法 .....	8
3) awp_CreatePlugin 方法 .....	8
4) awp_DestroyPlugin 方法 .....	9
5) awp_getPluginById 方法 .....	9
6) awp_getHttpClient 方法 .....	10
7) awp_getPluginUtility 方法 .....	10
8) awp_getContainer 方法 .....	10
2. allWebPlugin 接口 .....	11
a) 方法 .....	11
9) createContainer 方法 .....	11
10) destroyContainer 方法 .....	12
11) destroyAllContainer 方法 .....	12
12) ShowUI 方法 .....	12
13) IsInstall 方法 .....	13
14) IsNewVersion 方法 .....	13
15) UpdateVersion 方法 .....	13
16) GetVersion 方法 .....	14
17) CheckPlugin 方法 .....	14
b) 属性 .....	14
1) ActivateContainer 属性 .....	15
3. PluginHostCnt 接口 .....	15

a) 方法 .....	15
1) createContainer 方法 .....	15
2) createPlugin 方法 .....	16
3) createProxyPlugin 方法 .....	16
4) get_pcid 方法 .....	17
5) CheckPlugin 方法 .....	17
6) getHttpClient 方法 .....	17
7) getPluginUtility 方法 .....	18
8) RegisterEvent 方法 .....	18
9) IsInstall 方法 .....	18
10) IsNewVersion 方法 .....	19
11) GetVersion 方法 .....	19
12) UpdateVersion 方法 .....	19
b) 属性 .....	20
1) UI 属性 .....	20
4. allWebPluginUI 接口 .....	20
a) 方法 .....	20
1) onUpdatePosition 方法 .....	20
2) ShowUI 方法 .....	21
3) ForceShowUI 方法 .....	21
4) EnabledCanvas 方法 .....	21
5) ShowMsg 方法 .....	22
6) UpdateTitle 方法 .....	22
5. NObject 接口 .....	22
a) 方法 .....	22
1) constructor 方法 .....	22
2) InvokeObject 方法 .....	23
3) get_pcid 方法 .....	23
4) get_obj 方法 .....	24
6. PluginUtility 接口 .....	24
a) 方法 .....	24
1) constructor 方法 .....	24
2) GetTempFileName 方法 .....	24
3) GetTempPath 方法 .....	25
4) CreateDirectory 方法 .....	25
5) DeleteFile 方法 .....	25
6) ClearDirectory 方法 .....	26
7) GetFileSize 方法 .....	26
8) FilePathExists 方法 .....	26
9) DirectoryExists 方法 .....	27
10) GetLocalOpenDialog 方法 .....	27
11) GetLocalSaveDialog 方法 .....	28

12) ShowMessage 方法 .....	28
13) ExecuteApp 方法 .....	29
14) IsExistApp 方法 .....	29
15) KillApp 方法 .....	29
7. httpclient 接口 .....	30
a) 方法 .....	30
1) constructor 方法 .....	30
2) Open 方法 .....	30
3) Close 方法 .....	31
4) SetRequestHeader 方法 .....	32
5) Send 方法 .....	32
6) AddForm 方法 .....	33
7) AddString 方法 .....	33
8) AddFile 方法 .....	34
9) Clear 方法 .....	34
10) GetStatus 方法 .....	34
11) GetStatusText 方法 .....	35
12) GetResponseText 方法 .....	35
13) GetResponseHeader 方法 .....	36
14) GetAllResponseHeaders 方法 .....	36
15) GetErrorCode 方法 .....	36
16) GetErrorText 方法 .....	37
17) ResponseSaveToFile 方法 .....	37
18) SetHttpTimeOut 方法 .....	37
19) SetCookie 方法 .....	37
b) 属性 .....	38
1) ShowProgressUI 方法 .....	38

## 一、开发背景

用户信息化系统往往会存在浏览器使用 ActiveX、NPAPI 插件，解决复杂技术和业务问题。但由于浏览器不断升级，推出各种限制，造成浏览器无法支持 ActiveX 插件，也无法支持新标准的 NPAPI 插件，因此用户系统也无法正常运行，只能让用户保持指定版本的浏览器来使用，造成用户工作不便，经常切换不同浏览器，维护成本大。针对此问题，我们使用全新技术，开发全新 allWebPlugin 产品解决。

## 二、软件功能

allWebPlugin 中间件是一款为用户提供安全、可靠、便捷的浏览器插件服务的中间件产品。将第三方（标准协议 ActiveX）插件直接嵌入 H5 浏览器，实现插件加载、接口调用、事件回调等中间件产品，产品拥有自主知识产权。产品具有支持多浏览器、多操作系统、ActiveX、NPAPI 标准协议控件的特点，并且调用接口友好、集成方便，为用户提供“信息化系统 + allWebPlugin + 插件 + 多浏览器”的解决方案。

## 三、软件特点

- 1) 支持任何版本的 H5 浏览器，包括最新谷歌内核版本浏览器，如：谷歌、火狐、Edge、360、搜狗、微信内置浏览器等。
- 2) 一次集成，所有浏览器都能运行，插件接口和事件调用脚本风格简单易懂，保持 IE 及谷歌浏览器脚本调用风格。
- 3) 插件直接在浏览器中运行，不是单独应用程序或窗体。与页面无缝衔接，支持任意比例缩放、支持多标签加载。
- 4) 插件方案相比 H5 “云架构”方案性价比更高，更易扩展，改造量小。
- 5) 面向对象开发接口设计，更易调用。

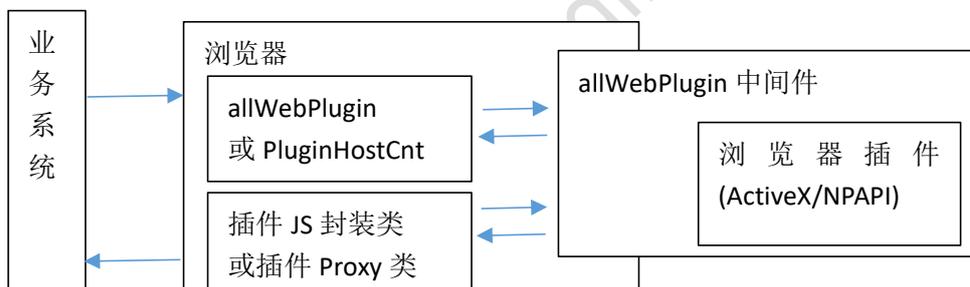
## 四、运行环境

客户机	操作系统: WIN7、WIN8、WIN10、WIN11 最低配置: CPU P4 2.8G, 内存 4G, 硬盘 500G。 推荐配置: Intel i3 及以上处理器, 内存 8G 以上, SSD 硬盘 128G 以上。
-----	---

## 五、浏览器插件集成步骤

### a) allWebPlugin 与浏览器、插件关系图

业务系统调用 allWebPlugin\PluginHostCnt 创建 allWebPlugin 中间件, allWebPlugin 中间件会嵌入浏览器页面, 并在内部容器中创建客户 ActiveX 或 NPAPI 插件, 根据插件标识实例化 JS 封装类对象, 实现插件无缝嵌入浏览器及插件接口的调用。



### b) 引入相关 allWebPlugin JS

为了页面集成插件, 需要引入 allWebPlugin.Common.js allWebPlugin.UI.j 及 allWebPlugin.Main.js 文件。如下图所示:

```
<script type="text/javascript" src="js/allWebPlugin.Common.v2.0.0.28.js"></script>
<script type="text/javascript" src="js/allWebPlugin.UI.v2.0.0.28.js"></script>
<script type="text/javascript" src="js/allWebPlugin.Main.v2.0.0.28.js"></script>
```

## c) 添加页面元素

在需要展示插件的位置添加 canvas 页面元素(注意: 其他页面元素也可以)。如下图所示:

```
<canvas id="allWebOfficeAx_zw" width="1024" height="600">
<p class="error">Sorry, It looks as though your browser does not support the canvas tag.</p>
</canvas>
```

## d) 创建插件对象

使用 awp\_CreatePlugin 接口创建 ActiveX 插件对象。如下图所示,

```
awp_CreatePlugin("allWebOfficeAx_zw","{6888ECA8-57D0-6E0E-9B9B-F1FF9B2DBD86}")
```

## e) 插件接口调用

awp\_CreatePlugin 默认使用 createProxyPlugin 创建插件对象。可以像传统插件接口调用一样, 调用插件对象的方法和属性, 调用读属性与调用插件方法接口一致。如下图所示:

```
allWebOfficeAx_zw.GetDisplayStyle().ShowMenuBar = false;
allWebOfficeAx_zw.GetDisplayStyle().ShowOpenProgress = false;
```

## f) 插件事件响应

为了响应插件事件, 需要在插件创建时, 实现插件响应函数。如下图所示:

```
function OnCreated(){
    alert("OnCreated 事件");//allWebOffice 插件创建事件
}
```

## 六、接口说明

### 1. 全局接口

#### 1) awp\_IsInstall 方法

**功能说明：**判断插件是否安装，是否需要升级。

**参数说明：**

installPackageVersion: 字符串 需要的版本号

installPackageUrl: 字符串，安装包更新地址。

**返回值：**true: 已经安装; false: 未安装或需要升级

**示例：**

```
var installPackageUrl =
"http://127.0.0.1:6651/install/allwebPlugin_x86_v2.0.1.16_20240806.exe";
var installPackageVersion = "2.0.0.28";
if (awp_IsInstall(installPackageVersion, installPackageUrl)) {
    awp_CreatePlugin("allWebOfficeAx_zw", "{6888ECA8-57D0-6E0E-9B9B-F1FF9B2DBD86}");
    allWebOfficeAx_zw.GetDisplayStyle().ShowMenuBar = false;
    allWebOfficeAx_zw.GetDisplayStyle().ShowOpenProgress = false;
}
```

#### 2) awp\_CreatePluginFromIE 方法

**功能说明：**自动检测网页文档中 object 元素并创建 ActiveX 插件。

**参数说明：**

Lic: 字符串 **【可选】** allWebPlugin 中间件授权，当缺省时，使用默认授权信息。

**返回值，无返回值**

**示例：**

#### 3) awp\_CreatePlugin 方法

**功能说明：**根据传入的页面元素 id，创建 ActiveX 插件。

**参数说明：**

id: 字符串 页面元素 id;

clsid: 字符串 【可选】插件 classid, 当缺省时, 将自动从页面元素的属性 classid 获取 classid 值;

Lic: 字符串 【可选】 allWebPlugin 中间件授权, 当缺省时, 使用默认授权信息。

返回值:

true: 创建插件成功, false: 创建插件失败

示例:

```
awp_CreatePlugin("allWebOfficeAx_zw", "{6888ECA8-57D0-6E0E-9B9B-F1FF9B2DBD86}");
```

#### 4) awp\_DestroyPlugin 方法

功能说明: 根据传入的页面元素 id, 销毁对应的插件容器对象。

参数说明:

id: 字符串 页面元素 id;

返回值:

无

示例:

```
awp_DestroyPlugin("allWebOfficeAx_zw");
```

#### 5) awp\_getPluginById 方法

功能说明: 根据传入的页面元素 id, 获取对应的插件对象。

参数说明:

id: 字符串 页面元素 id;

返回值:

插件对象

示例:

```
var allWebOffice = awp_getPluginById("allWebOfficeAx_zw");
```

## 6) awp\_getHttpClient 方法

**功能说明：** 获取当前插件容器对应的 HttpClient 对象。

**参数说明：**

无

**返回值：**

HttpClient 对象

**示例：**

```
var httpClient= awp_getHttpClient();
```

## 7) awp\_getPluginUtility 方法

**功能说明：** 获取当前插件容器对应的 PluginUtility 对象。

**参数说明：**

无

**返回值：**

PluginUtility 对象

**示例：**

```
var Utility = awp_getPluginUtility();
```

## 8) awp\_getContainer 方法

**功能说明：** 根据传入的页面元素 id，获取对应的插件容器对象。

**参数说明：**

id: 字符串 页面元素 id;

**返回值：**

PluginHostCnt 对象

**示例：**

```
var Cnt = awp_getContainer("allWebOfficeAx_zw");
```

## 2. allWebPlugin 接口

allWebPlugin 类用于管理 PluginHostCnt 类，实现在同一页面上加载多个插件。

### a) 方法

#### 9) createContainer 方法

**功能说明：**创建插件容器。

**参数说明：**

canvas: 网页 canvas 画布元素

progID: 字符串 插件容器标识

license: 字符串 授权信息

bFixSize: 布尔 是否固定大小。**注意：**百分比自适应时，为 false，不显示插件截图。

本参数自 v2.0.0.16 起支持

bUsePluginUI: 布尔 是否使用插件容器对话框。**注意：**false 时，使用浏览器自带提示框，提示框显示时会隐藏插件容器；true 时，使用插件容器对话框，不会隐藏插件容器窗口。自 v2.0.0.20 起支持

**返回值：**

类型: PluginHostCnt 对象

当返回值不为 null 时表示成功，其它表示失败。

**示例**

```
var hostCnt_zw = g_allWebPlugin.createContainer(canvas_zw, "PluginContainer", license, false);
if(hostCnt_zw)
{
    hostCnt_zw.UI.ForceShowUI(false); //创建时隐藏插件, 需要调用这句, 隐藏插件
    hostCnt_zw.RegisterEvent("OnCreated;OnCommand;OnRibbonCtrlCommand;OnOLECommand");
    allWebOfficeAx_zw = CreatePlugin(hostCnt_zw);
}
```

```
<!--设置宽高大小, 创建插件时, bFixSize 为 true, 如: createContainer(canvas, "PluginContainer", license, true)-->
<canvas id="pluginUI" width="1024" height = "768">
<p class="error">Sorry, It looks as though your browser does not support the canvas tag.</p>
</canvas>

<!--百分比控制大小, 创建插件时,
bFixSize 为 false, 如: createContainer(canvas, "PluginContainer", license, false)-->
<!--canvas id="pluginUI" style="width:100%;height:100%">
<p class="error">Sorry, It looks as though your browser does not support the canvas tag.</p>
</canvas-->
```

## 10)destroyContainer 方法

**功能说明：** 销毁插件容器。

**参数说明：**

    \_container: PluginHostCnt 对象

**返回值：**

    无返回值

**示例**

```
g_allWebPlugin.destroyContainer(g_allWebPlugin.ActivateContainer);
```

## 11)destroyAllContainer 方法

**功能说明：** 销毁全部插件容器。

**参数说明：**

    无

**返回值：**

    无返回值

**示例**

```
g_allWebPlugin.destroyAllContainer();
```

## 12)ShowUI 方法

**功能说明：** 设置所有插件容器是否显示。

**参数说明：**

    blShow: 布尔 是否显示, true: 所有插件容器窗口显示; false: 所有插件容器窗口不显示

**返回值：**

    无

**注意：** ShowUI 只用于临时性设置窗口是否可见，便于消息提醒、浏览器页面切换等操作。

**示例**

```
g_allWebPlugin.ShowUI(true);
```

## 13)IsInstall 方法

**功能说明：**检测本地是否安装 allWebPlugin 中间件服务。

**参数说明：**无

**返回值：**

类型：布尔

True: 已安装; false: 未安装

**示例**

```
alert(g_allWebPlugin.IsInstall());
```

## 14)IsNewVersion 方法

**功能说明：**判断给定版本号与当前安装版本是否更新。

**参数说明：**

**version:** 字符串 版本号。

**返回值：**

类型：布尔

当返回值为 true 时表示给定版本号是新的版本，需要更新；其它表示给定版本号不是新的版本。

**示例**

```
Alert(g_allWebPlugin.IsNewVersion("2.0.0.10"))
```

## 15)UpdateVersion 方法

**功能说明：**按照指定的 url 地址，下载更新包并安装。

**参数说明：**

**srcUrl:** 字符串 更新包 Url 地址。

**返回值：**

类型：json 对象

code: 0 表示更新成功，其他值表示失败。

msg: code 码详细描述

## 示例

```
g_allWebPlugin.UpdateVersion("http://127.0.0.1:6651/install/allWebPlugin_x86_v2.0.0.10_20240518.exe").then(function(data) {
    if(data.code == 0) {
        alert("升级成功。");
    }else{
        alert("升级失败。错误码: " + data.code);
    }
});
```

## 16)GetVersion 方法

**功能说明：**获取当前安装版本号。

**参数说明：**无

**返回值：**

类型：字符串

返回当前版本号。

**示例**

```
alert(g_allWebPlugin.GetVersion());
```

## 17)CheckPlugin 方法

**功能说明：**检查本机是否安装插件，安装插件是 32 位还是 64 位。自 v2.0.28.0 起支持

**参数说明：**

**clsid：** 字符串 插件的 CLASSID；所有字符大写。

**返回值：**

类型：整形。0：未安装插件；1: 32 位插件；2: 64 位插件。

**示例**

```
g_allWebPlugin.CheckPlugin("{9BE31822-FDAD-461B-AD51-BE1D1C159921}");
```

## b) 属性

## 1) ActivateContainer 属性

**功能说明：** 获取或设置当前活跃的 PluginHostCnt 对象。

**参数说明：** 无

**类 型：** 返回或设置 PluginHostCnt 对象

## 3. PluginHostCnt 接口

创建插件之前，必须先创建插件容器程序；不同插件类型，使用不同插件容器程序。

ActiveX 插件使用 PluginContainer（32 位）、PluginContainerX64（64 位，自 V2.0.28.0 版本支持）插件表示；NPAPI 插件使用 NPluginContainer（预留，NPAPI 插件支持正在夜以继日开发中）。

### a) 方法

#### 1) createContainer 方法

**功能说明：** 创建插件容器。

**参数说明：**

canvas: 网页 canvas 画布元素

progID: 字符串 插件容器标识

license: 字符串 授权信息

bFixSize: 布尔 是否固定大小。**注意：**百分比自适应时，为 false，不显示插件截图。

本参数自 v2.0.0.16 起支持

blUsePluginUI: 布尔 是否使用插件容器对话框。**注意：**false 时，使用浏览器自带提示框，提示框显示时会隐藏插件容器；true 时，使用插件容器对话框，不会隐藏插件容器窗口。自 v2.0.0.20 起支持

**返回值：**

类型: 布尔

当返回值为 true 时表示成功，其它表示失败。

## 示例

```
if(hostCnt.createContainer(canvas,"PluginContainer",license) == true){
    hostCnt.RegisterEvent("OnCreated;OnCommand;OnRibbonCtrlCommand");
    CreatePlugin();
}
```

```
<!--设置宽高大小,创建插件时,bFixSize 为 true,如: hostCnt.createContainer(canvas,"PluginContainer",license,true)-->
<canvas id="pluginUI" width="1024" height="768">
<p class="error">Sorry, It looks as though your browser does not support the canvas tag.</p>
</canvas>
<!--百分比控制大小,创建插件时,bFixSize 为 false,如:
hostCnt.createContainer(canvas,"PluginContainer",license,false)-->
<!--canvas id="pluginUI" style="width:100%;height:100%">
<p class="error">Sorry, It looks as though your browser does not support the canvas tag.</p>
</canvas-->
```

## 2) createPlugin 方法

**功能说明:** 创建插件,通过获取插件标识,创建对应的 JavaScript 插件对象。

**参数说明:**

**clsid:** 字符串 插件 ID (ActiveX classID)。

**Params:** Json 对象(可选) 插件对象初始化参数。

如: var params = {Copyright:" XX" };

**返回值:**

类型: 布尔

当返回值为 true 时表示成功,其它表示失败。

**示例**

```
var varResponse = hostCnt.createPlugin("{6888ECA8-57D0-6E0E-9B9B-F1FF9B2DBD86}");
if(varResponse != null){
    pluginObj = new allWebPluginOffice(varResponse.pcid,varResponse.ret.v,varResponse.ret.t);
}
```

## 3) createProxyPlugin 方法

**功能说明:** 创建插件,不需要提前编写插件脚本对象,可直接调用。

**参数说明:**

**clsid:** 字符串 插件 ID (ActiveX classID)。

**Params:** Json 对象(可选) 插件对象初始化参数。

如: `var params = {Copyright:" XX" };`

#### 返回值:

类型: 布尔

当返回值为 true 时表示成功, 其它表示失败。

#### 示例

```
var AxProxy = hostCnt.createPlugin("{6888ECA8-57D0-6E0E-9B9B-F1FF9B2DBD86}");
if(AxProxy!= null) {
    var = AxProxy.Version();
}
```

## 4) get\_pcid 方法

功能说明: 获取插件容器 ID。

参数说明: 无

#### 返回值:

类型: 字符串

返回当前插件容器 ID, 每个插件拥有唯一的 ID。

#### 示例

```
var varResponse = hostCnt.createPlugin("{6888ECA8-57D0-6E0E-9B9B-F1FF9B2DBD86}");
if(varResponse != null) {
    pluginObj = new allWebPluginOffice(varResponse.pcid, varResponse.ret.v, varResponse.ret.t);
}
```

## 5) CheckPlugin 方法

功能说明: 检查本机是否安装插件, 安装插件是 32 位还是 64 位。自 v2.0.28.0 起支持

参数说明:

`clsid`: 字符串 插件的 CLASSID; 所有字符大写。

#### 返回值:

类型: 整形。0: 未安装插件; 1: 32 位插件; 2: 64 位插件。

#### 示例

```
hostCnt.CheckPlugin("{9BE31822-FDAD-461B-AD51-BE1D1C159921}");
```

## 6) getHttpClient 方法

功能说明: 获取 httpClient 对象。

**参数说明：**无

**返回值：**

类型：NObject 对象

当返回值为 null 时表示失败，其它表示成功。

**示例**

```
httpClientObj = hostCnt.getHttpClient();
```

## 7) getPluginUtility 方法

**功能说明：**获取 PluginUtility 对象。

**参数说明：**无

**返回值：**

类型：NObject 对象

当返回值为 null 时表示失败，其它表示成功。

**示例**

```
pluginUtilityObj = hostCnt.getPluginUtility();
```

## 8) RegisterEvent 方法

**功能说明：**注册 createPlugin 插件的响应事件，事件之间通过";"隔开。

**参数说明：**无

**返回值：**

无

**示例**

```
hostCnt.RegisterEvent("OnCreated;OnCommand;OnRibbonCtrlCommand");
```

## 9) IsInstall 方法

**功能说明：**检测本地是否安装 allWebPlugin 中间件服务。

**参数说明：**无

**返回值：**

类型：布尔

True: 已安装; false: 未安装

## 示例

```
alert(hostCnt.IsInstall());
```

## 10)IsNewVersion 方法

功能说明：判断给定版本号与当前安装版本是否更新。

参数说明：

**version:** 字符串 版本号。

返回值：

类型：布尔

当返回值为 true 时表示给定版本号是新的版本，需要更新；其它表示给定版本号不是新的版本。

## 示例

```
Alert(hostCnt.IsNewVersion("2.0.0.10"))
```

## 11)GetVersion 方法

功能说明：获取当前安装版本号。

参数说明：无

返回值：

类型：字符串

返回当前版本号。

## 示例

```
alert(hostCnt.GetVersion());
```

## 12)UpdateVersion 方法

功能说明：按照指定的 url 地址，下载更新包并安装。

参数说明：

**srcUrl:** 字符串 更新包 Url 地址。

返回值：

类型：json 对象

code: 0 表示更新成功, 其他值表示失败。

msg: code 码详细描述

### 示例

```
hostCnt.UpdateVersion("http://127.0.0.1:6651/install/allWebPlugin_x86_v2.0.0.10_20240518.exe")
    .then(function(data) {
    if(data.code == 0) {
        alert("升级成功。");
    }else{
        alert("升级失败。错误码: " + data.code);
    }
    });
```

## b) 属性

### 1) UI 属性

功能说明: 获取 allWebPluginUI 对象。

参数说明: 无

类 型: 返回 allWebPluginUI 对象

## 4. allWebPluginUI 接口

allWebPlugin 中间件与页面元素关联及界面展示相关接口。

## a) 方法

### 1) onUpdatePosition 方法

功能说明: 根据页面元素位置更新插件容器窗口位置。

参数说明:

blResize: 整型 是否窗口变化时, 1: 浏览器窗口大小变化; 0: 浏览器窗口大小不变

返回值:

无

示例

```
hostCnt.UI.onUpdatePosition(0);
```

## 2) ShowUI 方法

**功能说明：** 设置插件容器是否显示。

**参数说明：**

blShow: 布尔 是否显示, true: 插件容器窗口显示; false: 插件容器窗口不显示

**返回值：**

无

**示例**

```
hostCnt.UI.ShowUI(true);
```

## 3) ForceShowUI 方法

**功能说明：** 设置插件容器是否显示。

**参数说明：**

blShow: 布尔 是否显示, true: 插件容器窗口强制显示; false: 插件容器窗口强制不显示

**返回值：**

无

**示例**

```
hostCnt.UI.ForceShowUI (true);
```

## 4) EnabledCanvas 方法

**功能说明：** 设置页面元素 Canvas 是否可用。

**参数说明：**

blEnabled: 布尔 是否可用。true: 页面 canvas 元素可用, 且使用插件容器界面更新 canvas 界面, 并隐藏插件容器窗口; false: 显示插件容器窗口。

**返回值：**

无

示例

```
hostCnt.UI.EnabledCanvas (true);
```

## 5) ShowMsg 方法

**功能说明：** 使用浏览器自带 alert 提示框，提示文本内容。本接口可以有效避免窗口遮挡提示框。

**参数说明：**

msg: 字符串 提示类容。

**返回值：**

无

示例

```
hostCnt.UI.ShowMsg("allWebPlugin");
```

## 6) UpdateTitle 方法

**功能说明：** 通知插件容器程序更新页面标题。

**参数说明：**

strTitle: 字符串 页面标题。

**返回值：**

无

示例

```
hostCnt.UI.UpdateTitle (document.title);
```

## 5. NObject 接口

所有插件可操作类都派生自 NObject 类。

### a) 方法

#### 1) constructor 方法

**功能说明：** 构建 NObject 对象。

**参数说明:**

pcid: 字符串 插件容器 ID

obj: 字符串 插件对象标识

返回值: 无

## 2) InvokeObject 方法

**功能说明:** 调用插件对象方法或属性。

**参数说明:** 不定长参数

fun: 字符串 插件对象方法名

params: 不定长参数 方法的参数

**返回值:**

类型: 可变类型。

返回接口结果

**示例**

```
get audio() {  
    var resp = this.InvokeObject("get_audio");  
    var ret;  
    if(resp === undefined)  
    {  
        return ret;  
    }  
    ret = new audio(this.pcid, resp.v);  
    return ret;  
}
```

## 3) get\_pcid 方法

**功能说明:** 获取当前插件容器标识。

**参数说明:** 无

**返回值:**

类型: 字符串。

返回插件容器标识

**示例**

```
ret = ActiveDocument.InvokeObject("Range", 0, 0);  
if(ret === null)
```

```
{  
    return;  
}  
var myRange = new NObject(pluginObj.get_pcid(), ret.v);
```

#### 4) get\_obj 方法

**功能说明：** 获取当前对象标识。

**参数说明：** 无

**返回值：**

类型：整形。

返回当前对象

## 6. PluginUtility 接口

插件实用工具类方法

### a) 方法

#### 1) constructor 方法

**功能说明：** 构建 PluginUtility 象。

**参数说明：**

pcid: 字符串 插件容器 ID

obj: 字符串 插件对象标识

**返回值：** 无

**示例**

```
resp = new PluginUtility(response.pcid, response.ret.v);
```

#### 2) GetTempFileName 方法

**功能说明：** 获取指定目录下临时文件。

**参数说明：**

strPath: 字符串 文件路径

strPrefix: 字符串 文件前缀

**返回值:**

类型: 字符串

返回临时路径

**示例**

```
pluginUtilityObj.ShowMessage(pluginUtilityObj.GetTempFileName("D:\\", "all"), "allwebplugin");
```

### 3) GetTempPath 方法

**功能说明:** 获取系统临时目录。

**参数说明:** 无

**返回值:**

类型: 字符串。

返回系统临时目录

**示例**

```
pluginUtilityObj.GetTempPath();
```

### 4) CreateDirectory 方法

**功能说明:** 创建目录。

**参数说明:**

strDir 字符串 指定目录

**返回值:**

类型: 布尔。

返回 true 表示目录创建成功, false 表示目录创建失败。

**示例**

```
pluginUtilityObj.CreateDirectory("D:\\allWebPlugin");
```

### 5) DeleteFile 方法

**功能说明:** 删除指定文件。

**参数说明:**

strFileName 字符串 指定文件路径

**返回值:**

类型：布尔。

返回 true 表示文件删除成功，false 表示文件删除失败。

#### 示例

```
pluginUtilityObj.DeleteFile("D:\\1.png")
```

## 6) ClearDirectory 方法

**功能说明：** 清空目录。

#### 参数说明：

bstrDir 字符串 目录路径

#### 返回值：

类型：布尔。

返回 true 表示目录清除成功，false 表示目录清除失败。

#### 示例

```
pluginUtilityObj.ClearDirectory("D:\\allWebPlugin");
```

## 7) GetFileSize 方法

**功能说明：** 获取指定文件存储大小。

#### 参数说明：

strFilePath 字符串 文件路径

#### 返回值：

类型：整型。

返回文件存储空间大小。

#### 示例

```
pluginUtilityObj.GetFileSize("D:\\1.png");
```

## 8) FilePathExists 方法

**功能说明：** 文件路径是否存在。

#### 参数说明：

strFilePath 字符串 文件路径

#### 返回值：

类型：整型。

返回文件存储空间大小。

#### 示例

```
pluginUtilityObj.FilePathExists ("D:\\1.png");
```

## 9) DirectoryExists 方法

**功能说明：** 文件目录是否存在。

**参数说明：**

strFilePath 字符串 文件路径

**返回值：**

类型：整型。

返回文件存储空间大小。

#### 示例

```
pluginUtilityObj.FilePathExists ("D:\\1.png");
```

## 10) GetLocalOpenDialog 方法

**功能说明：** 打开本地文档选择对话框。

**参数说明：**

strFilter 字符串 文件类型过滤器 不同类型的过滤器，通过";;"隔开。例如：

"docx 文件 (\*.docx;\*.doc);;excel 文件 (\*.xls;\*.xlsx);;所有文件 (\*.\*)"。

**返回值：**

类型：字符串。

返回选择文件路径。

**注意：** 接口异步调用，避免堵塞浏览器，出现页面无响应的情况。

#### 示例

```
function OpenLocalFile() {
  pluginUtilityObj = hostCnt.getPluginUtility();
  pluginUtilityObj.GetLocalOpenDialog("docx 文件 (*.docx;*.doc);;excel 文件 (*.xls;*.xlsx);;所有文件 (*.*)").then(function(data) {
    console.log(data);
    if(data != "")
      pluginObj.Open(data).then(function(nOpen) {
```

```

        var varDisplayStyle = pluginObj.GetDisplayStyle();
        if(varDisplayStyle)
        {
            varDisplayStyle.ShowToolBar = false;
        }
    })
});
}

```

## 11) GetLocalSaveDialog 方法

**功能说明：** 打开另存为对话框。

**参数说明：**

strFilter 字符串 文件类型过滤器 不同类型的过滤器，通过“;”隔开。例如：  
 “docx 文件 (\*.docx;\*.doc);;excel 文件 (\*.xls;\*.xlsx);;所有文件 (\*.\*)”。

**返回值：**

类型：字符串。

返回选择文件路径。

**注意：** 接口异步调用，避免堵塞浏览器，出现页面无响应的情况。

**示例**

```

function SaveAs() {
    pluginUtilityObj = hostCnt.getPluginUtility();
    pluginUtilityObj.GetLocalSaveDialog("docx 文件 (*.docx;*.doc);;excel 文件 (*.xlsx)").then(function(varFile) {
        pluginObj.Save(varFile, 0, false).then(function(data) {
            pluginUtilityObj.ShowMessage(data);
        });
    });
}

```

## 12) ShowMessage 方法

**功能说明：** 消息提示框。

**参数说明：**

strMsg 字符串 消息文本内容

strTitle 字符串 消息提示框标题

uType 整型 对话框的内容和行为。 此参数可以是以下标志组中的标志的组合。

具体参考 windows MessageBox uType 参数含义

**返回值:**

类型: 整数

具体参考 windows MessageBox 返回值含义。

**注意:** 接口异步调用, 避免堵塞浏览器, 出现页面无响应的情况。

### 13)ExecuteApp 方法

v2.0.0.16 起

**功能说明:** 启动进程。

**参数说明:**

strAppFilePath 字符串 程序路径  
strCmdLine 字符串 命令行参数

**返回值:**

类型: 布尔型 创建进程是否成功。

### 14)IsExistApp 方法

v2.0.0.16 起

**功能说明:** 进程是否存在。

**参数说明:**

strAppName 字符串 进程名称

**返回值:**

类型: 布尔型 进程是否存在。

### 15)KillApp 方法

**功能说明:** 杀掉进程。v2.0.0.16 起

**参数说明:**

strAppName 字符串 进程名称

**返回值:**

类型: 布尔型 进程杀出是否成功。

## 示例

```
function GetTempPath() {
    if(pluginUtilityObj)
    {
        pluginUtilityObj.ShowMessage(pluginUtilityObj.GetTempPath(), "allwebplugin");
    }
}
```

## 7. httpclient 接口

提供 http Client 请求接口，可轻松实现跨域请求。

### a) 方法

#### 1) constructor 方法

**功能说明：**构建 httpclient 象。

**参数说明：**

pcid: 字符串 插件容器 ID

obj: 字符串 插件对象标识

**返回值：**无

**示例**

```
resp = new httpclient(response.pcid, response.ret.v);
```

#### 2) Open 方法

**功能说明：**打开网络请求。

**参数说明：**

httpMethod: 整型 0: 表示 GET 请求, 1: 表示 POST 请求

strUrl: 字符串 URL 请求路径

bAsync: 布尔 是否异步通信, true: 异步, false 同步

strUser: 字符串 服务器用户名。无用户名, 可为空。

strPassword: 字符串 服务器用户口令。无用户口令, 可为空。

#### 返回值:

类型: 布尔

返回 true 表示网络连接成功, false 表示网络连接失败。

#### 示例

```
function LoadFile(){
    pluginUtilityObj = hostCnt.getPluginUtility();
    pluginUtilityObj.GetLocalSaveDialog("pdf 文件(*.pdf);;").then(function(varFile){
        httpclientObj = hostCnt.getHttpClient();
        if (httpclientObj.Open(0, "https://www.gov.cn/gzdt/att/att/site1/20100917/0019b90463ff0dfd641102.pdf", false)) {
            httpclientObj.Send().then(function(data){
                if (httpclientObj.GetStatus() == 200) {
                    httpclientObj.ResponseSaveToFile(varFile);
                    pluginUtilityObj.ShowMessage("LoadFile Status:" +
httpclientObj.GetStatus(), "allWebPlugin");
                }else{
                    pluginUtilityObj.ShowMessage("LoadFile Status:" +
httpclientObj.GetStatus(), "allWebPlugin");
                }
                httpclientObj.Close();
            });
        }
    });
}
```

### 3) Close 方法

**功能说明:** 关闭一个服务器地址连接。

**参数说明:** 无

#### 返回值:

类型: 布尔

返回 true 表示网络关闭成功, false 表示网络关闭失败。

#### 示例

```
function LoadFile(){
    pluginUtilityObj = hostCnt.getPluginUtility();
    pluginUtilityObj.GetLocalSaveDialog("pdf 文件(*.pdf);;").then(function(varFile){
        httpclientObj = hostCnt.getHttpClient();
        if (httpclientObj.Open(0, "https://www.gov.cn/gzdt/att/att/site1/20100917/
```

```
0019b90463ff0dfd641102.pdf", false)) {
    httpClientObj.Send().then(function(data) {
        if (httpClientObj.GetStatus() == 200) {
            httpClientObj.ResponseSaveToFile(varFile);
            pluginUtilityObj.ShowMessage("LoadFile Status:" +
                httpClientObj.GetStatus(), "allWebPlugin");
        }else{
            pluginUtilityObj.ShowMessage("LoadFile Status:" +
                httpClientObj.GetStatus(), "allWebPlugin");
        }
        httpClientObj.Close();
    });
}
});
}
```

#### 4) SetRequestHeader 方法

**功能说明：**设置 HTTP 请求头信息。

**参数说明：**

strName: 请求参数名称。

strValue: 请求参数名称对应的值。

**返回值：**无

**示例：**

```
httpClientObj = hostCnt.getHttpClient();
httpClientObj.SetRequestHeader("user", "admin");
```

#### 5) Send 方法

**功能说明：**发送数据包到服务器地址。

**参数说明：**无

**返回值：**

类型: 布尔

返回 true 表示数据发送成功, false 表示数据发送失败。

**示例：**

```
function LoadFile() {
    pluginUtilityObj = hostCnt.getPluginUtility();
```

```
pluginUtilityObj.GetLocalSaveDialog("pdf 文件(*.pdf);;").then(function(varFile) {
    httpclientObj = hostCnt.getHttpClient();
    if (httpclientObj.Open(0, "https://www.gov.cn/gzdt/att/att/site1/20100917/0019b90463ff0dfd641102.pdf", false)) {
        httpclientObj.Send().then(function(data) {
            if (httpclientObj.GetStatus() == 200) {
                httpclientObj.ResponseSaveToFile(varFile);
                pluginUtilityObj.ShowMessage("LoadFile Status:" +
                    httpclientObj.GetStatus(), "allWebPlugin");
            } else {
                pluginUtilityObj.ShowMessage("LoadFile Status:" +
                    httpclientObj.GetStatus(), "allWebPlugin");
            }
            httpclientObj.Close();
        });
    }
});
}
```

## 6) AddForm 方法

**功能说明：**添加数据到 HTTP 数据包中一起发送到服务器。

**参数说明：**

**strName：**字符串 数据项名称。

**strValue：**字符串 数据项名称对应的值。

**返回值：**

无

返回 **true** 表示数据发送成功，**false** 表示数据发送失败。

**示例：**

```
httpclientObj = hostCnt.getHttpClient();
httpclientObj.AddForm("user", "admin");
```

## 7) AddString 方法

**功能说明：**添加文本到 HTTP 数据包中一起发送到服务器。

**参数说明：**

**strData：**字符串 文本内容。

**返回值:**

无

**示例:**

```
httpClientObj = hostCnt.getHttpClient();  
httpClientObj.AddString("allWebPlugin");
```

## 8) AddFile 方法

**功能说明:** 添加文件到 HTTP 数据包中一起发送到服务器。

**参数说明:**

**strName:** 字符串 文件数据项名称。

**strFileName:** 字符串 文件数据项名称对应的文件路径。

**返回值:**

无

**示例:**

```
httpClientObj = hostCnt.getHttpClient();  
httpClientObj.AddForm("user", "admin");
```

## 9) Clear 方法

**功能说明:** 清除自定义的头信息和数据包内容。

**参数说明:**

无

**返回值:**

无

**示例:**

```
httpClientObj = hostCnt.getHttpClient();  
httpClientObj.Clear();
```

## 10) GetStatus 方法

**功能说明:** 返回网络连接后的状态值。

**参数说明:**

无

**返回值:**

类型 : 整型

网络连接成功返回 200,请参考 HTTP 返回状态值。

**示例:**

```
httpClientObj = hostCnt.getHttpClient();
if (httpClientObj.Open(0, "https://www.gov.cn/gzdt/att/att/site1/20100917/0019b90463ff0dfd641102.pdf", false)) {
    httpClientObj.Send().then(function(data) {
        if (httpClientObj.GetStatus() == 200) {
            httpClientObj.ResponseSaveToFile(varFile);
            pluginUtilityObj.ShowMessage("LoadFile Status:" +
httpClientObj.GetStatus(), "allWebPlugin");
        }else{
            pluginUtilityObj.ShowMessage("LoadFile Status:" +
httpClientObj.GetStatus(), "allWebPlugin");
        }
        httpClientObj.Close();
    });
}
```

## 11)GetStatusText 方法

**功能说明:** 返回网络连接后的状态文本。

**参数说明:**

无

**返回值:**

类型 : 字符串

返回网络连接后的状态文本。

**示例:**

## 12)GetResponseText 方法

**功能说明:** 返回网络连接后的服务器文本。

**参数说明:**

无

**返回值:**

类型 : 字符串

返回网络连接后的服务器文本。

示例：

### 13)GetResponseHeader 方法

**功能说明：**获取服务器响应的头信息。

**参数说明：**

strHeader 字符串 数据项名称

**返回值：**

类型：字符串

返回服务器响应的头信息。

示例：

### 14)GetAllResponseHeaders 方法

**功能说明：**获取服务器响应的头信息。

**参数说明：**无

**返回值：**

类型：字符串

取服务器响应的头信息。

示例：

### 15)GetErrorCode 方法

**功能说明：**返回网络连接后的错误码值。

**参数说明：**无

**返回值：**

类型：整型

返回网络连接后的错误码值。

示例：

## 16)GetErrorText 方法

**功能说明：**返回网络连接后的错误文本值。

**参数说明：**无

**返回值：**

类型：字符串

返回网络连接后的错误文本值。

**示例：**

## 17)ResponseSaveToFile 方法

**功能说明：**保存服务器响应的数据到文件。

**参数说明：**

strFilePath 字符串 保存文件路径。

**返回值：**

无

**示例：**

## 18)SetHttpTimeOut 方法

**功能说明：**设置网络连接超时时间。

**参数说明：**

lTime 整型 超时时长，时间单位为秒。

**返回值：**

无

**示例：**

## 19)SetCookie 方法

**功能说明：**设置 Cookie 信息。

**参数说明：**

strCookie 字符串 cookie 内容

返回值:

无

示例:

## b) 属性

### 1) ShowProgressUI 方法

功能说明: 是否显示进度条。

参数说明: 无

类 型: 布尔型

allWebPlugin中间件