

aardio 范例: COM 接口 - 嵌入控件

```
//COM 接口 - 嵌入控件
import win.ui;
/*DSG{*/
var winform = win.form(text="嵌入COM控件";right=759;bottom=469)
winform.add(
lbStatus={cls="static";left=25;top=435;right=725;bottom=460;db=1;dl=1;dr=1;transparent=1;z=2};
static={cls="static";text="Static";left=11;top=8;right=744;bottom=426;db=1;dl=1;dr=1;dt=1;transparent=1;z=1}
)
/*}*/
```

```
//嵌入 COM 控件, 宿主窗口这里指定为 winform.static
var embedObject= winform.static.createEmbedEx("Shell.Explorer.2");
```

```
/*
```

上面的代码主要执行以下操作:

- 1、创建容器对象 embedObject, 控件容器通常用于封装 COM 对象, 同时也会注册为 COM 对象默认的 COM 事件监听器。
- 2、创建 COM 对象 (Shell.Explorer 对象) 并保存为 embedObject._object, 注意这才是真正的 COM 对象。
- 3、创建 COM 控件宿主对象并保存为 embedObject._host, 这个对象会提供部分 OLE 接口函数, 一般没必要直接使用这个对象。
- 4、将 COM 对象嵌入宿主窗口并显示, 宿主窗口存为 embedObject._form。
- 5、在宿主窗口调整大小时自动调整 COM 控件窗口大小。
- 6、在宿主窗口销毁前自动解除默认事件监听器并释放 COM 对象。

注意在 aardio 里下划线开头的成员是只读的, 并且在智能提示候选列表里是默认隐藏的 (按 _ 就会出来)

宿主窗口一般不需要绘图, 所以对宿主窗口默认会执行以下操作:

如果宿主窗口是 win.form 对象, 且未指定 onEraseBkgnd 事件, 那么会自动添加 embedObject._form.onEraseBkgnd = lambda() 0;
如果宿主窗口是其他控件, 自动添加 _WS_CLIPCHILDREN 样式, 并移除 _WS_EX_TRANSPARENT 扩展样式。

容器对象 embedObject 通常用于实现 COM 代理对象, 用于拦截、封装对 embedObject._object 的操作。

winform.static.createEmbedEx() 函数会创建一个默认的代理对象, 访问 embedObject 的成员会自动转为调用 embedObject._object
winform.static.createEmbed() 函数不会创建默认代理, 如果希望更精细地封装 COM 接口, 可以使用这个函数。
可在标准库 win.ui.ctrl.metaProperty 内可以查看这 2 个函数的源码。

```
*/
```

```
//这下定义事件回调函数就省事了( 因为 embedObject 是默认的 COM 事件监听器)
```

```
embedObject.StatusTextChanged = function(text) {
    winform.lbStatus.text = text;
}
```

```
//下面这样写是错的, 原生 COM 控件不是事件监听表
```

```
//embedObject._object.StatusTextChanged = function(text) {}
```

```
//通过控件容器调用 COM 对象打开指定的网页
```

```
embedObject.Navigate("about:test");
```

```
/*
```

上面这句会自动转为调用 embedObject._object.Navigate("about:test")

embedObject 已通过元方法成为了上述的 COM 控件代理对象, 访问 embedObject 的成员会自动转为调用 embedObject._object。

如果希望自定义代理层, 请改用 winform.static.createEmbed() 函数创建控件 (函数名字后面少了 Ex)。

```
*/
```

```
winform.show();
```

```
win.loopMessage();
```

[Markdown 格式](#)