牛牛安装包界面控件使用说明

本说明文档旨在对nsNiuniuSkin.dll控件的使用进行说明，指导调用者顺利的将牛捉安装包界面控件控件集成进安装包中，使第三方产品也能拥有完美的安装包的UI体验功能。

官方网址：<http://www.leeqia.com/>

牛牛安装包界面控件是一个可集成于NSIS的插件，采用Duilib开发，在使用时，安装包制作者只需要做如下两件事情：

1. 通过配置Duilib的资源，设计好界面显示的元素

2. 在NSIS的脚本中，通过NSIS脚本调用nsNiuniuSkin.dll的相关接口，集成UI及安装包的业务功能

在控件的资源中，采用的是通过TAB控件来实现不同阶段的安装界面，比如：选择路径、许可协议、安装进度、完成、卸载等，在实际使用中，通过NSIS脚本来设置当前需要显示的TAB页，即可完美的呈现出需要的界面UI了。

在我们的示例中，提供了多份安装的界面如

下：



更多安装包UI，请查看：

<http://www.leeqia.com/nsniuniuskin/template/#ct2>

**注意，最新版本的控件已经将可用的NSIS及7z的相关文件包含在示例包中，可以通过一键打包批处理脚本进行打包了，具体如下：**

1). 将要打包的文件复制到FilesToInstall目录下，可以包含文件及若干层的目录

2). 修改相关项目下的skin目录下的资源文件及xml脚本来配置UI

3). 添加相关（或者在我们原来的示例上修改，这样更快）的NSIS脚本文件，并与UI相匹配

4). 指定好相关的ICO、licence等信息

5). 执行build.bat脚本，一键打包，build脚本会做三件事：将FilesToInstall目录下的文件打包成app.7z；将skin目录下的文件打包成skin.zip；调用nsis编译程序，编译nsi文件，输出安装包。

注：如果是使用build-nozip.bat版本，则是直接将FilesToInstall下的文件纳入到NSIS的压缩与释放过程中。

以上描述的build脚本为全自动执行，可以轻松集成进您现有的软件发布流程中，实现自动化打包发布。

牛牛安装包界面控件包含的接口较多，以下逐一进行介绍[以下接口调用及参数说明按在NSIS脚本中的调用示例来说明]：

# 引擎初始化接口

InitEngine

在使用控件接口前，务必先调用初始化接口，否则将会有不可预测的结果。

调用示例：

nsNiuniuSkin::InitEngine

# DPI放大兼容生效接口

EnableDpi

此接口用于控件根据DPI的设置，自动的使用不同的字号以及图片。

调用示例：

nsNiuniuSkin::EnableDpi 1 1 0 0

从前往后的4个参数依次表示是否支持1.5倍，2倍，3倍，4倍图，在我们这个示例中，表示支持1.5倍和2倍放大兼容。

在设置此接口后，如果是1.5倍以下的DPI，则还是使用原来1:1的显示；如果是1.5至2倍之间的DPI放大，则使用1.5倍的窗口进行显示；如果是2倍或以上的DPI放大，程序中会自动使用2倍的窗口进行显示；需要对每一个图片，都有一个2倍图，并且命名要加上@2x，如：

原图： test.png

1.5倍图： test@1.5x.png

2倍图： test@2x.png

# UI初始化接口

InitSkinPage

此接口函数用于初始化nsNiuniuSkin.dll控件的配置信息.

调用示例：

nsNiuniuSkin::InitSkinPage "$PLUGINSDIR\" "${INSTALL\_LICENCE\_FILENAME}"

参数说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数序号 | 参数类型 | 参数说明 | 备注 |
| 1 | 字符串 | 用于指定NSIS安装包的插件释放路径 | 此路径的指定非常重要，在脚本中指定的插件以及UI资源包将会释放至此目录下，只有正确指定后，界面控件才能调用资源显示窗口 |
| 2 | 字符串 | 许可协议的文件名 | 这是一个txt文档，在界面控件加载时，将会加载此文件来显示许可协议 |

# 重新设置许可协议的文件名

nsNiuniuSkin::ResetLicenseFile $hInstallDlg "newlicensename.rtf"

设置后，许可协议显示控件将会重新加载许可协议文件，这个比较适合用于多语言版本的不同许可协议加载显示。

# 设置安装包标题

SetWindowTile

此接口函数用于指定安装包的标题。

调用示例：

nsNiuniuSkin::SetWindowTile $hInstallDlg "${PRODUCT\_NAME}安装程序"

参数说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数序号 | 参数类型 | 参数说明 | 备注 |
| 1 | 整型 | 用于指定要设置标题的窗口句柄 | 调用init时返回 （可传递子窗口弹窗口的句柄） |
| 2 | 字符串 | 用于指定安装包的标题 |  |

# 设置当前显示的TAB页

nsNiuniuSkin::ShowPageItem $hInstallDlg "wizardTab" ${INSTALL\_PAGE\_CONFIG}

|  |  |  |  |
| --- | --- | --- | --- |
| 参数序号 | 参数类型 | 参数说明 | 备注 |
| 1 | 整型 | 用于指定要设置标题的窗口句柄 | 调用init时返回 （可传递子窗口弹窗口的句柄） |
| 2 | 字符串 | 指定的TAB控件的name |  |
| 3 | int | 需要显示的tab页序号 | 以0为初始值 |

# 路径选择相关接口

4.1 通知接口来浏览安装路径

nsNiuniuSkin::SelectInstallDir

Pop $0

通知界面控件来浏览安装路径，并且将路径获取到变量中。再调用此接口后，应该再调用SetControlAttribute接口来将安装的路径设置到控件界面上。

4.2 通知接口来浏览安装路径扩展接口

nsNiuniuSkin::SelectInstallDirEx $hInstallDlg "请选择安装路径"

Pop $0

此接口是SelectInstallDir的扩展接口，在SelectInstallDir的基础上，支持设置文件选择框的标题。

# 显示界面控件

nsNiuniuSkin::ShowPage 0

当界面绑定事件完成后，通过此接口来调用显示控件，此接口将会阻塞运行。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数序号 | 参数类型 | 参数说明 | 备注 |  |
| 1 | 整型 | 用于指定显示前要回调的NSIS函数 | 如果为0，则表示不回调 |  |
|  |  |  |  |  |

# 弹出提示框接口

1) 以下代码用于初始化子窗口弹窗，同时返回其句柄：

nsNiuniuSkin::InitSkinSubPage "msgBox.xml" "btnOK" "btnCancel,btnClose"

pop $hInstallSubDlg

|  |  |  |  |
| --- | --- | --- | --- |
| 参数序号 | 参数类型 | 参数说明 | 备注 |
| 1 | 字符串 | 指定弹窗要用到的UI配置xml文件 |  |
| 2 | 字符串 | 指定点击后弹窗退出时返回IDOK的按钮ID号 | 如果使用BindCallBack绑定了此按钮回调的NSIS函数，则此设置失效 |
| 3 | 字符串 | 指定点击后弹窗退出时返回IDCANCEL的按钮ID号 | 如果使用BindCallBack绑定了此按钮回调的NSIS函数，则此设置失效 |

2) 以下代码用于根据弹窗窗口的句柄设置其UI控件的属性，控制显示变化：

nsNiuniuSkin::SetControlAttribute $hInstallSubDlg "lblTitle" "text" "提示"

3) 以下代码将弹窗显示出来：

nsNiuniuSkin::ShowSkinSubPage 0

参数说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数序号 | 参数类型 | 参数说明 | 备注 |
| 1 | 整型 | 指定显示弹窗前要回调的NSIS函数地址 | 如果为0表示不回调； |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# 指定界面上指定控件的属性

nsNiuniuSkin::SetControlAttribute $hInstallDlg "btnClose" "enabled" "false"

nsNiuniuSkin::SetControlAttribute $hInstallDlg "lblInstalling" "text" "正在卸载..."

此接口用于指定界面的指定元素的指定属性，比如：是否可用、是否可见、是否选中、文字、背景图等等

参数说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数序号 | 参数类型 | 参数说明 | 备注 |
| 1 | 整型 | 用于指定要设置标题的窗口句柄 | 调用init时返回 （可传递子窗口弹窗口的句柄） |
| 2 | 字符串 | 控件的name |  |
| 3 | 字符串 | 控件的属性名 |  |
| 4 | 字符串 | 控件的属性值 |  |

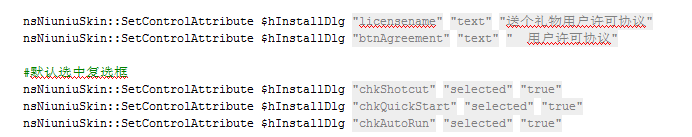
注：

可以通过此接口设置所有的通用属性；一般可以利用此接口来设置指定控件的文本、是否可用、是否可见、位置、大小 、背景图、文本颜色等；比如可以用于设置复选框是否选中等等。

通用的属性列表项可以参照下图：



设置示例如下：



# 获取控件的通用属性

nsNiuniuSkin::GetControlAttribute $hInstallDlg "editDir" "text"

Pop $2

此接口用于获取界面的指定元素的指定属性，比如：是否可用、是否可见，文本等等

参数说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数序号 | 参数类型 | 参数说明 | 备注 |
| 1 | 整型 | 用于指定要设置标题的窗口句柄 | 调用init时返回 （可传递子窗口弹窗口的句柄） |
| 2 | 字符串 | 控件的name |  |
| 3 | 字符串 | 控件的属性名 |  |
|  |  |  |  |

注：

通用的属性列表项可以参照下图：



# 绑定UI上控件的相关事件

GetFunctionAddress $0 OnExitDUISetup

nsNiuniuSkin::BindCallBack $hInstallDlg "btnClose" $0

此接口用于绑定一个按钮点击的回调函数或一个RichEdit控件的文字变化时的回调函数。

参数说明：

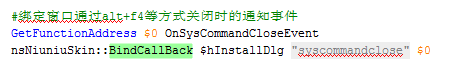
|  |  |  |  |
| --- | --- | --- | --- |
| 参数序号 | 参数类型 | 参数说明 | 备注 |
| 1 | 整型 | 用于指定要设置标题的窗口句柄 | 调用init时返回 （可传递子窗口弹窗口的句柄） |
| 2 | 字符串 | UI界面中按钮或richedit控件的name |  |
| 3 | LONG | 要绑定的NSIS函数的地址 | 当指定名称的按钮被点击或者指定的richedit的内容变化时，绑定的函数将会被触发 |

注：

1). 当绑定的控件是一个RichEdit控件时，其中的文本内容变化时就会触发绑定的函数；可以通过绑定路径的控件，同时在绑定的函数中获取路径，做否合法的判断。

descript

2). 为了能够将通过Ctrl+F4关闭窗口，以及通过在任务栏关闭窗口的事件通知到NSIS中，此处需要绑定一个特殊的名称，这个特殊的名称是：syscommandclose，如下：



# 结束安装

nsNiuniuSkin::ExitDUISetup

此接口被调用时，将会退出安装进程

# 安装窗口显示与隐藏

nsNiuniuSkin::ShowHide hwnd 1

第一个参数为要显示或隐藏的窗口句柄（此处一般转入主窗口句柄），第二个参数控制是显示还是隐藏。

如果要隐藏安装窗口，脚本中按如下方法使用：

nsNiuniuSkin::ShowHide $hInstallDlg 0

# 字符串处理辅助接口

nsNiuniuSkin::StringHelper "c:\test\test\" "\" "" "trimright"; 如果源字符串的最后一个字符是\，则将其去掉

pop $0

nsNiuniuSkin::StringHelper "c:\AAA\dst" "AAA" "bbb" "replace" ;将源码中的AAA替换成bbb

pop $0

nsNiuniuSkin::StringHelper "c:\AAA\dst" "\" "" "getrightbychar" ;将源码中的以\分隔的最后一段返回，此例中返回dst

pop $0

nsNiuniuSkin::StringHelper "c:\AAA\dst" "\" "" "tolower" ;将源码中的以\分隔的最后一段返回，此例中返回dst

pop $0

这是一个为了简化在NSIS脚本中的字符串处理而提供的辅助函数，有需要者可以使用（**字符串最大支持1KB**）。

|  |  |  |  |
| --- | --- | --- | --- |
| 参数序号 | 参数类型 | 参数说明 | 备注 |
| 1 | 字符串 | 等待处理的字符串 |  |
| 2 | 字符串 | 指定在处理时，需要用来进行辅助查询的参数值 |  |
| 3 | 字符串 | 要被替换成的目标字符串 | 仅当指令为replace时有效 |
| 4 | 字符串 | 用于指定要对字符串处理的操作类型 | replace：  在源串中将第二个参数的值替换成第三个参数的值，再返回  trimright：  将源串中的最后一个字符去掉（如果等于第二个参数）  trimleft：  将源串中的第一个字符去掉（如果等于第二个参数）  getrightbychar:  从源串中，取根据第二个参数分割的最后一段值  tolower:  将字符串转换成小写 |

# 文件读写处理辅助接口

这是一个为了方便在NSIS脚本中的读取与写入文件而提供的辅助函数，有需要者可以使用。

**参数说明：**

|  |  |  |  |
| --- | --- | --- | --- |
| 参数序号 | 参数类型 | 参数说明 | 备注 |
| 1 | int | 标识是读还是写 | 1表示读  2表示写 |
| 2 | int | 表示文件编码 | 0表示ansi  1表示utf8  此值用于控制插件内部的编码转换 |
| 3 | 字符串 | 要操作的文件路径 |  |
| 4 | 字符串 | 要操作的文件内容 | 当第一个参数为1，表示读文件时，第4个参数不传；内容最大支持1KB |

**返回值：**

如果是读文件，则会返回读取结果和读取内容两个字段；如果是写，则只返回写的结果。

**示例代码：**

nsNiuniuSkin::FileHelper 1 1 "c:\\test.conf" #读文件内容

pop $R0

pop $R1

MessageBox MB\_OK|MB\_ICONINFORMATION "retval: $R0, retstr: $R1"

nsNiuniuSkin::StringHelper "$R1" "{test\_ip}" "127.0.0.1" "replace" #替换字符串

Pop $R1

nsNiuniuSkin::StringHelper "$R1" "{test\_port}" "443" "replace" #替换字符串

Pop $R1

MessageBox MB\_OK|MB\_ICONINFORMATION "str: $R1"

nsNiuniuSkin::FileHelper 2 1 "c:\\test.conf" "$R1" #将替换后的字符串写入文件

pop $R0

MessageBox MB\_OK|MB\_ICONINFORMATION "retval: $R0"

# HTTP请求接口

nsNiuniuSkin::HttpInvoke

此接口用于在NSIS中发起HTTP请求，主要用于安装行为上报以及安全验证等。

|  |  |  |  |
| --- | --- | --- | --- |
| 参数序号 | 参数类型 | 参数说明 | 备注 |
| 1 | 字符串 | 要请求的URL |  |
| 2 | 字符串 | 要携带的httpheader |  |
| 3 | 字符串 | 要post的数据包 | 如果为空字符串，则直接GET请求 |
| 4 | int | 是否异步请求 | 0：表示同步请求  请求后，会在结果中返回请求是否成功（0表示成功）以及内容  1：表示异步请求  将会在插件内启动线程去访问，不管成功与否，都不会有结果通知到脚本中（适用于安装行为上报，不需要处理结果的场景） |

**httpheader一般为（用在post keyvalue数据的场景）：**

Content-Type: application/x-www-form-urlencoded;charset=utf-8

**或（用在post json数据的时候）：**

Content-Type:application/xml;charset=UTF-8

**同步调用示例：**

*Function TestSyncHttpRequest*

*nsNiuniuSkin::HttpInvoke "http://www.ggniu.cn/test/test.html" "" "" 0*

*pop $R0*

*pop $R1*

*MessageBox MB\_OK|MB\_ICONINFORMATION "retval: $R0, retstr: $R1"*

*FunctionEnd*

*GetFunctionAddress $0 TestSyncHttpRequest*

*BgWorker::CallAndWait*

**异步调用示例：**

*nsNiuniuSkin::HttpInvoke "http://www.ggniu.cn/test/test.html" "" "" 1*

# 单实例运行

同一个软件的安装包，一般情况下我们只允许同时运行一个，这主要是为了避免两个安装包同时安装时，选择文件数据错乱。

*nsNiuniuSkin::OnlyOneInstance "${PRODUCT\_NAME}安装程序"*

通过此接口，我们会根据安装包类名，以及传入的窗口标题，查找是否存在相同名称的窗口，如果存在，则认为已经存在了安装实例，此时我们将显示原来的窗口，同时退出当前的安装流程。

注：

在安装包首次启动，正常初始化时，需要设置窗口标题，否则会导致找不到：

*nsNiuniuSkin::SetWindowTile $hInstallDlg "${PRODUCT\_NAME}安装程序"*

# 特殊使用场景

此章描述如DPI放大、在线安装包、静默安装以及升级安装等等场景的集成与使用。

## 19.1 DPI放大

显示器分辨率越来越高，为了便于使用，大部分用户都会选择高的分辨率，同时设置DPI放大，在这种场景下，如果安装包不能跟随放大显示，则会显示成一个很小的窗口，非常影响美观。

nsNiuniuSkin内置了支持DPI放大的能力，会自动将原来控件的尺寸字号等使用放大后的尺寸与字号，同时会根据DPI放大的比例，使用对应比例的图片，确保在不同的DPI放大下，都有最好的显示效果。

目前我们支持1倍图和2倍图，如果DPI放大是在2倍以下，我们按不放大处理；如果是2倍或以上，则我们按2倍显示。

使用中需要两处调整：

1. 在控件引擎初始化后，立即进行DPI放大的支持设置（如在DUIPage函数中）：nsNiuniuSkin::EnableDpi 1 1 0 0
2. 在skin下的图片资源中，对每一个图片，增加其1.5倍和2倍图，如：

原图： test.png

1.5倍图： test@1.5x.png

2倍图： test@2x.png

## 19.2 升级包覆盖安装

**使用背景：**

现在基本所有软件都有升级检测功能，当在主程序中检测到新版本后，一般会在主程序中将升级包下载下来，然后调用升级包，对原有的程序进行升级，同时将新的主程序拉起。

在这样的使用背景下，安装包需要具备能够通过命令行自动传入安装路径，自动开始安装，完成后，自动打开主程序等能力，nsNiuniuSkin不仅仅是一套安装包UI，更是一整套安装包的解决方案，对此，我们已经提供了能力支撑，只需要在启动安装包时，传入指定的命令行参数即可：

*Setup.exe -AutoInstall=1 -AutoOpen=1 -fakecmd=1 /D=E:\Software\Test\*

如此，便指定了安装路径，同时自动安装，完成后，自动打开主程序，特别需要注意的是

**-fakecmd**，这个参数并无实际意义，主要用于占位，否则在安装包中解析前面的一个参数会失败。

注：

这种安装模式是在运行时通过命令行参数决定的，升级包与原始安装包是同一个包，通过命令行参数控制其不同的行为。

## 19.3 静默安装

在一些特殊的场景中，需要将一个安装包嵌入到其他的安装包中，在安装时，不需要用户干预，在后台自动安装，期间不要有任何的UI显示；nsNiuniuSkin也提供了这样的能力，按下面的方式启动即可：

*Setup.exe /S*

当然，如果需要在安装完成后，自动打开主程序，或者指定安装路径，也可以在参数中指定：

*Setup.exe -AutoOpen=1 -fakecmd=1 /S /D=E:\Software\Test\*

## 19.4 在线安装

在一些安装场景中，由于完整的安装包很大，下载时间长，且下载后需要人工干预来进行安装，这样会一定程度的降低用户使用体验；nsNiuniuSkin提供了一种精简版本的安装包，支持在线下载安装文件，自动进行安装的功能。

在线安装的脚本与完整安装的脚本是同一套，区别在于编译时指定了宏：INSTALL\_DOWNLOAD\_7Z（通过bat脚本动态的传入，无需修改脚本），当这个宏被定义时，安装过程中间插入了下载的过程，下载过程中，我们将显示安装文件总大小，文件下载速度，下载完成百分比等信息，通过进度条来体现下载进度；同时在下载安装后，对下载的文件进行解压，并提供安装进度，无缝的衔接好下载与安装的两个流程。

**支持条件：**

由于在线安装文件的下载可能因为种种原因出现错误，所以我们会在下载完成后，对文件的md5以及大小进行校验；在安装的下载过程中，我们分成了如下几步：

1. 下载config.ini
2. 根据config.ini中指定的文件名称进行文件下载，并实时的显示进度
3. 下载到文件后，取文件大小以及md5值，与配置文件中的值进行校验，如果不正确，则报错；如果正确，再进一步对文件进行解压安装

**在实际使用中，需要做如下几件事情：**

1. 对需要在线安装的文件及列表进行7z压缩（一般是需要放到FilesToInstall目录下的列表），命名为app.7z
2. 将app.7z上传到http服务器上的指定目录，如：test\_online\_install
3. 取文件的md5值和大小，写入服务器上test\_online\_install目录下的config.ini文件中，如：

*[set]*

*md5=e3e718669b80a484ce06de00adf52e95*

*size=5157529*

*filename=app.7z*

1. 在nsi文件中修改宏INSTALL\_DOWNLOAD\_BASEURL，输入服务器上test\_online\_install目录对应的http请求路径
2. 执行build-xxx-online.bat 生成在线安装包的安装程序

当启动在线安装程序时，它先下载app.7z，校验后解压来对在线安装数据进行安装。

# 其他信息

具体的NSIS脚本及DUILIB的配置文件可以参看DEMO，有如下几个注意事项：

## 20.1 其他自行定义的功能说明

目前提供的示例仅包含了常见的与UI界面相关的NSIS脚本；需要进一步的处理特定的注册表写入、文件备份等操作，需要自行另外再写NSIS脚本。

## 20.2 轮播图

界面的安装过程中的图片支持轮播，并且可以设置间隔时间，在XML中的写法如下：

*<ImageShow images="images\banner1.png images\banner3.png" padding="60,33,0,0" imagecount="2" elapse="1500" loop="true" width="440" height="200" />*

其中比较重要的参数是elapse，用于控制图片轮换的间隔时长，单位为毫秒。

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名称 | 参数类型 | 参数说明 | 备注 |
| images | 字符串 | 填写要轮换的图片，空格分开 | 图片大小要一致 |
| imagecount | int | 图片数量 | 与images匹配 |
| elapse | int | 轮换间隔 | 毫秒 |
| loop | bool | 是否循环 | 所有图片轮换完成后，是否循环 |
| moveimage | bool | 轮换时是否使用渐变 |  |
| movestep | int | 每次移动像素值 | moveimage为true时生效，默认值为10 |
| moveinterval | int | 隔多久移动一次 | 定时器参数，毫秒  moveimage为true时生效，默认值为8 |

## 20.3 gif动画显示

在部分场景下，我们需要使用一个gif动画来显示当前的安装状态，，在XML中的写法如下：

*<GifAnim bkimage="images\walk.gif" autosize="false" autoplay="true" />*

其中比较重要的参数是elapse，用于控制图片轮换的间隔时长，单位为毫秒。

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名称 | 参数类型 | 参数说明 | 备注 |
| bkimage | 字符串 | gif图片路径 |  |
| autoplay | bool | 是否自动播放 |  |
| autosize | bool | 是否自适应gif大小 | 如果为true，则会将控件大小设置为gif大小 |