**在单文档应用程序中切换视图窗口的方法**

一、原理

在单文档应用程序中切换视图的方法基本思路如下：

1、定义多个视图类

2、在切换时，创建一个新的视图对象

3、定义一个CCreateContext对象，为将文档类于新的视图类联系起来，方法是设置CCreateContext的m\_pCurrentDoc成员，调用视图类的Create函数

4、初始化视图、设置为当前视图，隐藏旧视图，显示新视图、将当前视图对象的ID设置为AFX\_IDW\_PANE\_FIRST，这里用到的函数分别为：OnInitialUpdate、SetActiveView、ShowWindow、SetDlgCtrlID

5、删除旧视图

6、RecalcLayout()

二、实现

下面以两个视图的切换为例，说明该方法

1、定义两个基于CView类的视图类，分别为CView3和CView4，并修改其构造函数由原来的protected改为public

2、定义CMainFrame类中定义SwitchToView()函数，函数定义如下：

void CMainFrame::SwitchToView()

{

//获取当前视图指针

CView\* pOldActiveView=GetActiveView();

CView\* pNewActiveView;

//判断当前视图属于哪个视图类，据此创建另一个类的视图对象，并将其指针传给pNewActiveView

if (pOldActiveView->GetRuntimeClass()==RUNTIME\_CLASS(CView3))

{

pNewActiveView=(CView\*)new CView4;

}

else

{

pNewActiveView=(CView\*)new CView3;

}

//定义CCreateContext对象，将文档类与新的视图类相关联

CCreateContext context;

context.m\_pCurrentDoc=pOldActiveView->GetDocument();

pNewActiveView->Create(NULL,NULL,WS\_BORDER|WS\_CHILD,

CFrameWnd::rectDefault,this,NULL,&context);

//将新的视图类对象设置为当前视图对象，并将旧的视图隐藏

pNewActiveView->OnInitialUpdate();

SetActiveView(pNewActiveView);

pNewActiveView->ShowWindow(SW\_SHOW);

pOldActiveView->ShowWindow(SW\_HIDE);

//将当前视图的ID设置为AFX+IDW\_PANE\_FIRST

pNewActiveView->SetDlgCtrlID(AFX\_IDW\_PANE\_FIRST);

//删除旧的视图

delete pOldActiveView;

RecalcLayout();

}

说明：AFX\_IDW\_PANE\_FIRST，对于SDI，视图窗口的ID就是这个值，在CMainFrame中通过GetDlgItem(AFX\_IDW\_PANE\_FIRST)就能得到当前视图窗口指针，对于MDI，客户区窗口的ID就是这个值，MDI的客户区就是在MDI下，没有打开作任何文档时工具条下面的窗口。实际上AFX\_IDW\_PANE\_FIRST是为了解决多个视图的情况下消息转发的问题，这是MFC内部使用的一个固定值，所有当前的活动视图都会切换到AFX\_IDW\_PANE\_FIRST这个ID，主窗口收到的一些消息（比如命令、通知等消息）会转发给活动视图来处理，框架通过这个ID来定位活动视图。当某个SDI应用程序中使用多个视图并支持视图切换时，很容易忽略这点，造成某此消息得不到正确的响应，因此当激活某个视图时也要把这个视图的ID改成AFX\_IDW\_PANE\_FIRST。

三、网上一种类似的实现方法

1、利用资源编辑器新建2个对话框，ID分别为IDD\_FORMVIEW1、IDD\_FORMVIEW2

2、为每个对话框创建一个CFormView类的派生类，分别为CView1、CView2

3、编写SwitchToFrom函数，代码如下：

void CMainFrame::SwitchToForm(UINT ID\_FORM)

{

CView \*pOldActiveView=GetActiveView();

CView \*pNewActiveView=(CView\*)GetDlgItem(ID\_FORM);

if (pNewActiveView==NULL)

{

switch(ID\_FORM)

{

case IDD\_FORMVIEW1:

pNewActiveView=(CView1\*)new CView1;

break;

case IDD\_FORMVIEW2:

pNewActiveView=(CView2\*)new CView2;

break;

}

CCreateContext context;

context.m\_pCurrentDoc=pOldActiveView->GetDocument();

pNewActiveView->Create(NULL,NULL,WS\_BORDER|WS\_CHILD,

CFrameWnd::rectDefault,this,

ID\_FORM,&context);

pNewActiveView->OnInitialUpdate();

}

SetActiveView(pNewActiveView);

pNewActiveView->ShowWindow(SW\_SHOW);

pOldActiveView->ShowWindow(SW\_HIDE);

if (pOldActiveView->GetRuntimeClass()==RUNTIME\_CLASS(CView1))

{

pOldActiveView->SetDlgCtrlID(IDD\_FORMVIEW1);

}

else if (pOldActiveView->GetRuntimeClass()==RUNTIME\_CLASS(CView2))

{

pOldActiveView->SetDlgCtrlID(IDD\_FORMVIEW2);

}

pNewActiveView->SetDlgCtrlID(AFX\_IDW\_PANE\_FIRST);

delete pOldActiveView;

RecalcLayout();

}

以上步骤中，省略了一步，即在APP初始化代码中将文档类模板的初始化参数中视图类参数改成相应的视图参数：

CSingleDocTemplate\* pDocTemplate;

pDocTemplate = new CSingleDocTemplate(

IDR\_MAINFRAME,

RUNTIME\_CLASS(CSwitchViewDemoDoc),

RUNTIME\_CLASS(CMainFrame),

RUNTIME\_CLASS(CView3));