Creating Navigations and Embedding Views at Runtime - Web Dynpro ABAP

Applies to:

SAP ECC 6.0 (Release 700, SP 12). For more information, visit the User Interface Technology homepage.

Summary

This Article is all about embedding views in window and creating navigations between the views dynamically at runtime. It is necessary to know because most of the applications require such a kind of design where the views and navigations are not fixed at design time rather at runtime.

Author: Abhimanyu Lagishetti

Company: Satyam Computer Services Ltd.

Created on: 18 Jun 2008

Author Bio

Abhimanyu L, Satyam Computer Services Ltd.

B.Tech Computer Science Graduate, working on Technologies like Web Dynpro ABAP, Web Dynpro JAVA, Enterprise Portals, ABAP and Business Workflows.

Table of Contents

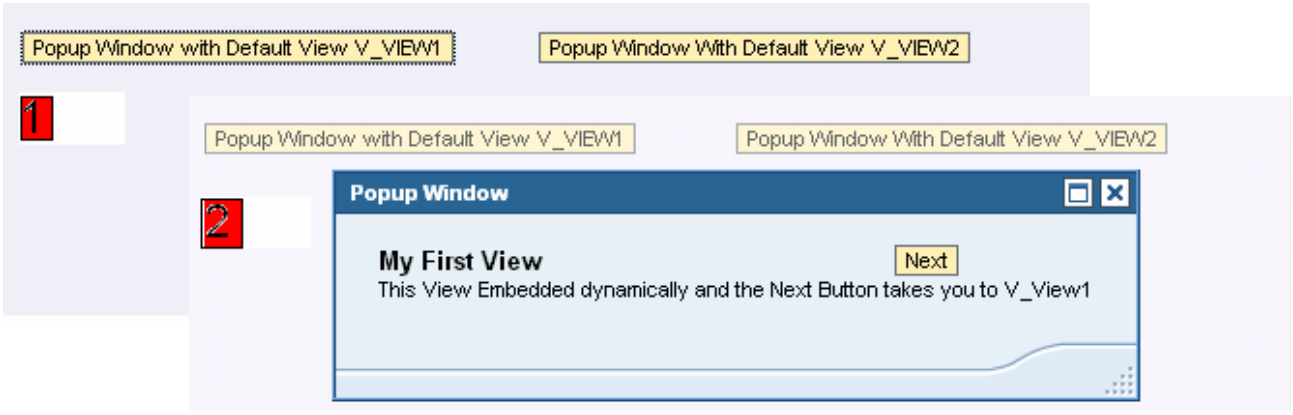
Introduction

Some of the business scenarios require such a design where the navigations and views to be shown are not fixed at design time; it is only possible at runtime. In Web Dynpro embedding views in window and creating navigation link all managed as metadata, the code for which is generated at runtime.

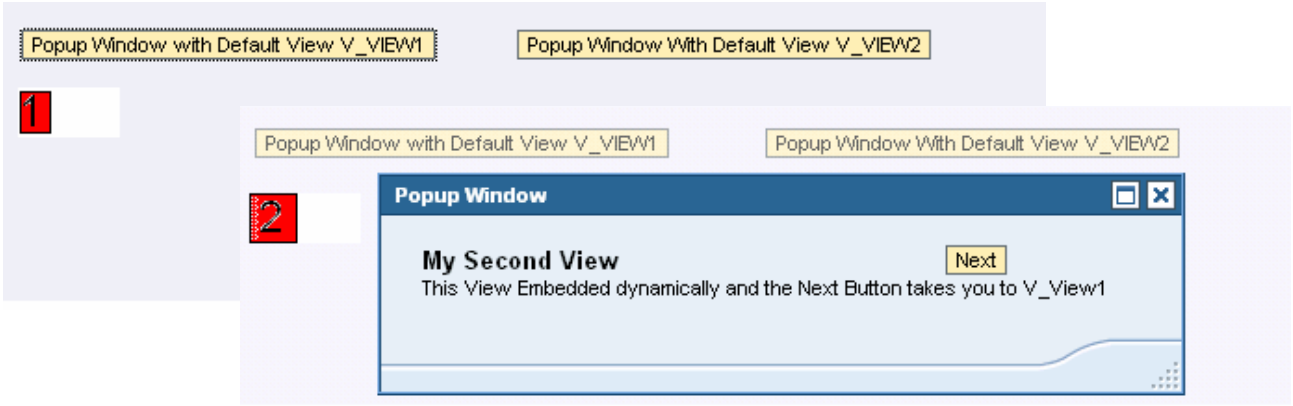
Get Started with an Application

The Goal of the application is to embed views in window and create navigation between the views dynamically.

1. Click on the first button
2. Shows a popup window with one view embedded as default



1. Click on the second button
2. Shows the same popup window with another view embedded as default.

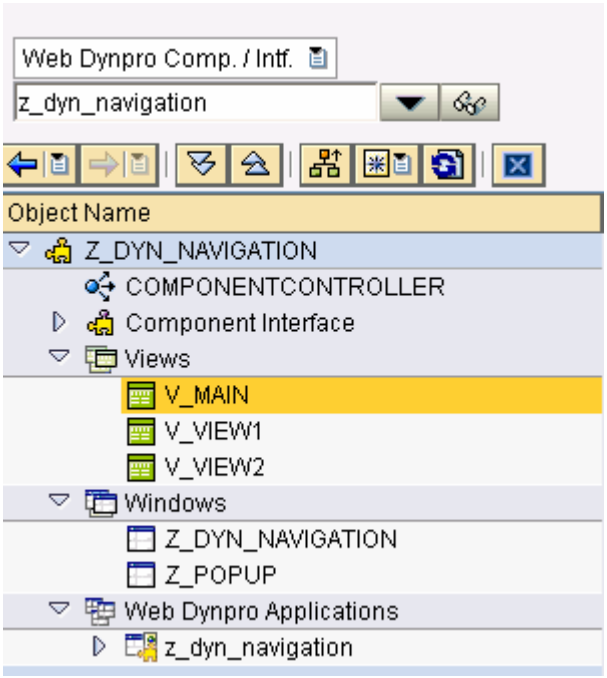


Building the Base Application

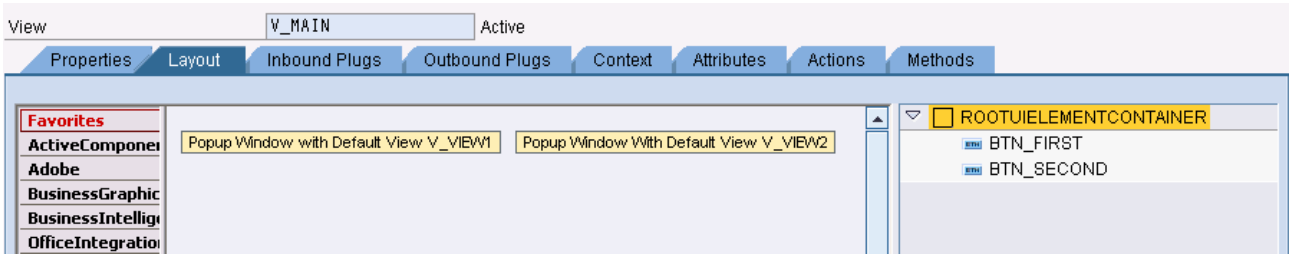
1. Create a Web Dynpro Component of the following structure hierarchy, where it contains :

   3 views – V_MAIN, V_VIEW1, V_VIEW2

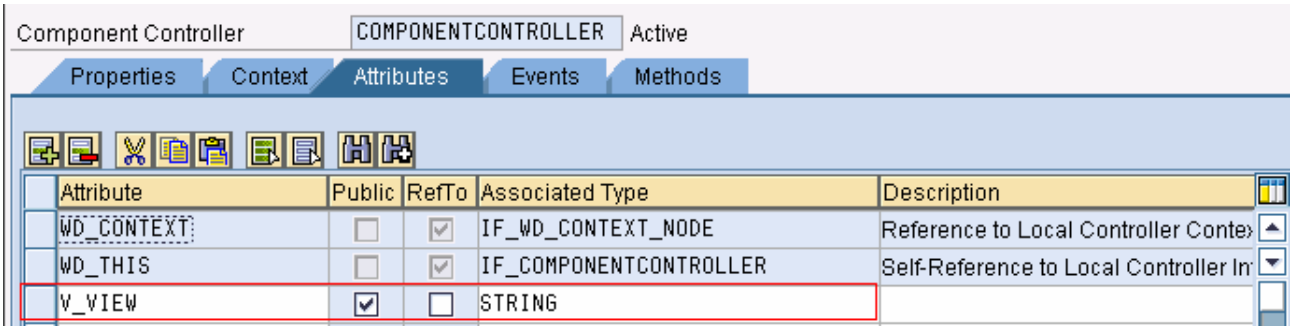
   2 windows – Z_DYN_NAVIGATION, Z_POPUP



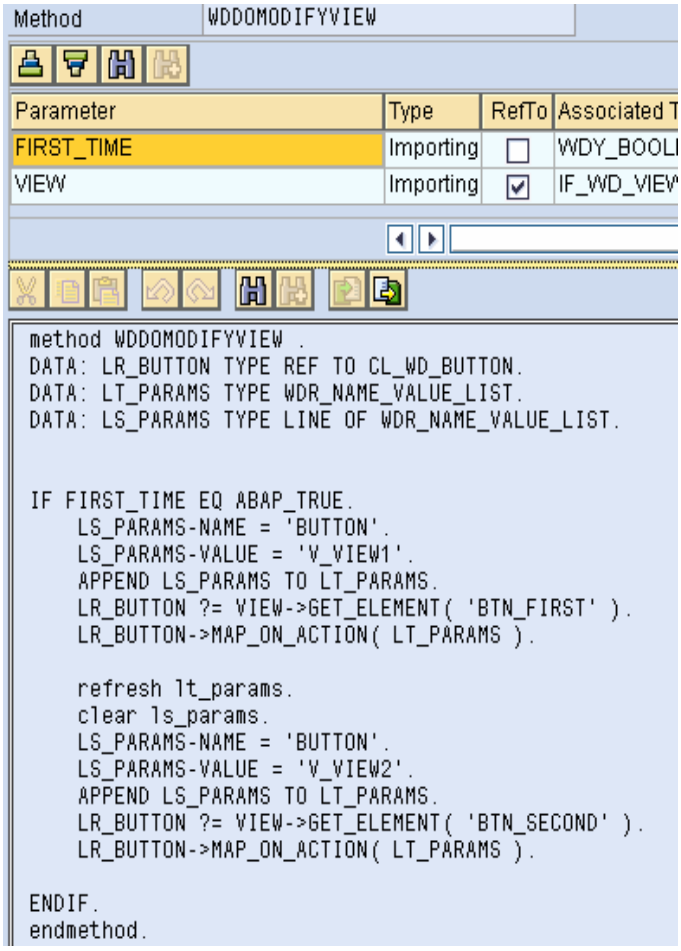
2. Design the V_MAIN as follows

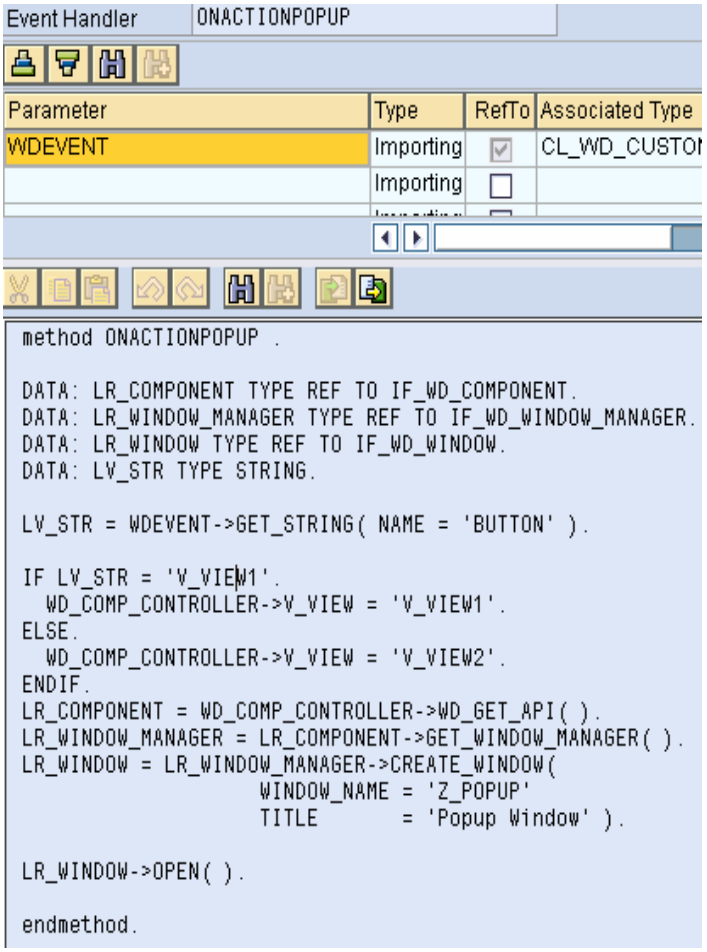


3. Create one Action ‘POPUP’ and assign to both the buttons.
4. Create an Attribute ‘V_VIEW’ of type string in the component controller.

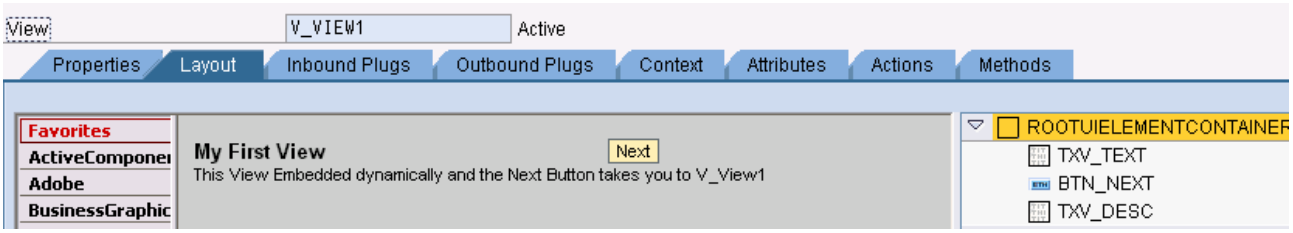


5. Place the code in the WDDOMODIFYVIEW of V_MAIN, where it is mapping parameter to the buttons
6. Also, place code in ONACTIONPOPUP event handler to capture the source mapping parameter and set the attribute V_VIEW and open the popup window Z_POPUP.





7. Design V_VIEW1 and V_VIEW2 as follows, create Inbound Plug ‘IN’ and Outbound Plug ‘OUT’ in both the views and on button click fire the ‘OUT’ plug. (remember the navigation link is created when you embed the views in window)



8. Place V_MAIN view in Window Z_DYN_NAVIGATION. (Do not place any view in Z_POPUP)

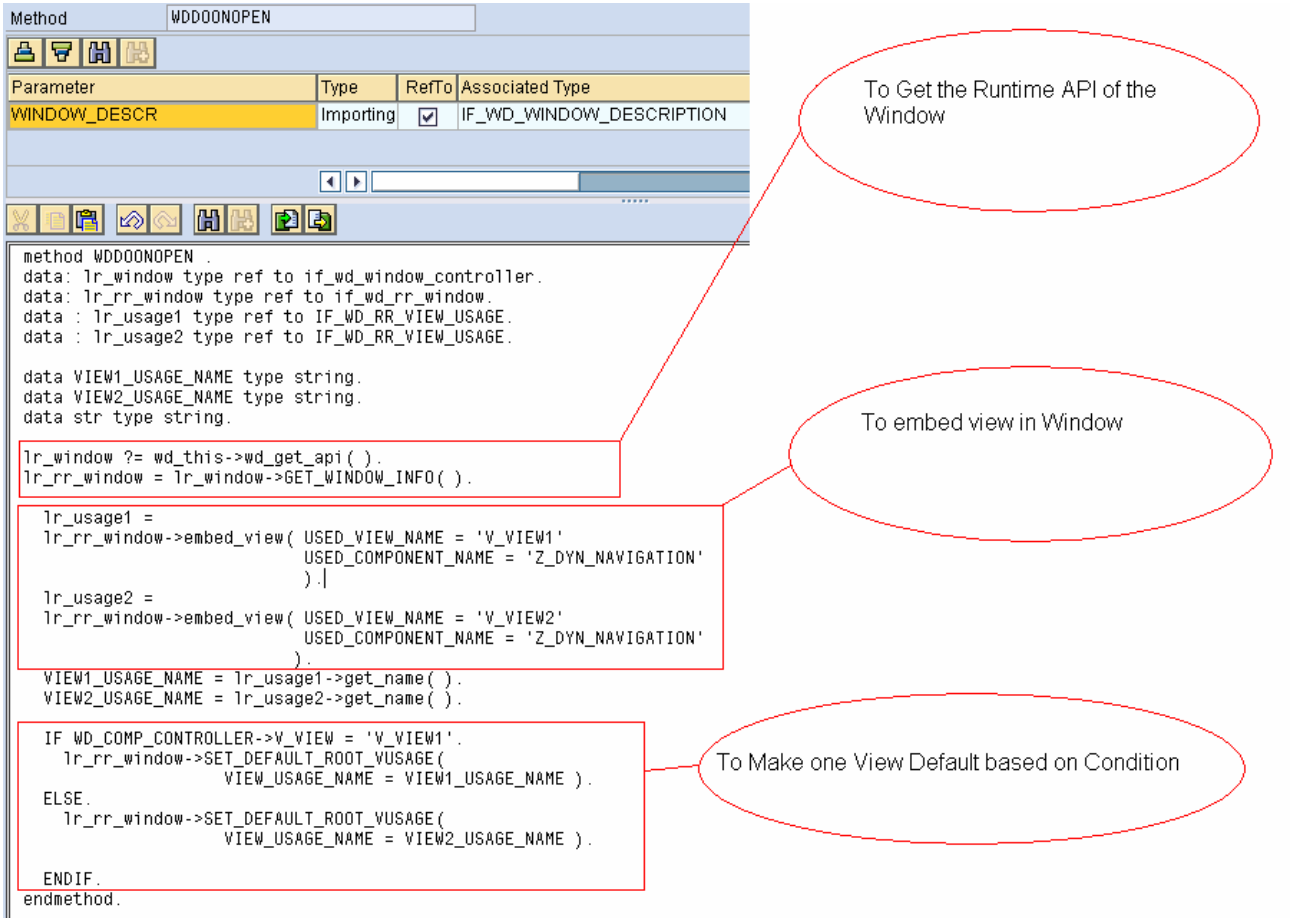
Embedding the Views in Window Dynamically

Place the following code sample in WDDOONOPEN of Z_POPUP window.

In this window we have not placed any View, we are going to place the views using the following API

IF_WD_RR_WINDOW -> Runtime Repository API: Window

EMBED_VIEW	Temporarily Embeds View in Specified Position
SET_DEFAULT_ROOT_VUSAGE	Sets Temporary Default View




```
method WDDOONOPEN .
data: lr_window type ref to if_wd_window_controller.
data: lr_rr_window type ref to if_wd_rr_window.
data : lr_usage1 type ref to IF_WD_RR_VIEW_USAGE.
data : lr_usage2 type ref to IF_WD_RR_VIEW_USAGE.

data VIEW1_USAGE_NAME type string.
data VIEW2_USAGE_NAME type string.
data str type string.

lr_window ?= wd_this->wd_get_api( ).
lr_rr_window = lr_window->GET_WINDOW_INFO( ).

  lr_usage1 =
  lr_rr_window->embed_view( USED_VIEW_NAME = 'V_VIEW1'
                            USED_COMPONENT_NAME = 'Z_DYN_NAVIGATION'
                            ).
  lr_usage2 =
  lr_rr_window->embed_view( USED_VIEW_NAME = 'V_VIEW2'
                            USED_COMPONENT_NAME = 'Z_DYN_NAVIGATION'
                           ).
  VIEW1_USAGE_NAME = lr_usage1->get_name( ).
  VIEW2_USAGE_NAME = lr_usage2->get_name( ).

  IF WD_COMP_CONTROLLER->V_VIEW = 'V_VIEW1'.
    lr_rr_window->SET_DEFAULT_ROOT_VUSAGE(
                  VIEW_USAGE_NAME = VIEW1_USAGE_NAME ).
  ELSE.
    lr_rr_window->SET_DEFAULT_ROOT_VUSAGE(
                  VIEW_USAGE_NAME = VIEW2_USAGE_NAME ).

  ENDIF.
endmethod.
```

Creating Navigation Link Dynamically

Use the following code to create navigation link between the embedded views, for which we can use the method CREATE_NAVIGATION_TARGET of the Runtime API of window .

Append the following code to WDDOOPEN method of window Z_POPUP, to navigate between V_VIEW1 and V_VIEW2 in the Z_POPUP popup window.

```
  lr_USAGE1->CREATE_NAVIGATION_TARGET(
                       NAME = 'IN'
                       SOURCE_PLUG_NAME = 'OUT'
                       TARGET_PLUG_NAME = 'IN'
                       TARGET_VIEW_USAGE_NAME = VIEW2_USAGE_NAME ).

  lr_USAGE2->CREATE_NAVIGATION_TARGET(
                       NAME = 'IN'
                       SOURCE_PLUG_NAME = 'OUT'
                       TARGET_PLUG_NAME = 'IN'
                       TARGET_VIEW_USAGE_NAME = VIEW1_USAGE_NAME ).
endmethod.
```

Summary

To work with dynamic navigations and embedding views, the API involved

IF_WD_RR_WINDOW and IF_WD_RR_VIEW_USAGE

Related Content

The Dynamic Navigation is implemented in SAP Standard Component LORD_MAINTAIN_COMP, which replaces the SAPGUI transactions for sales order processing VA01 through VA05.

https://www.sdn.sap.com/irj/sdn/go/portal/prtroot/docs/library/uuid/6081997d-b33d-2a10-2ea5-f945b3467927

For more information, visit the User Interface Technology homepage.

Disclaimer and Liability Notice

This document may discuss sample coding or other information that does not include SAP official interfaces and therefore is not supported by SAP. Changes made based on this information are not supported and can be overwritten during an upgrade.

SAP will not be held liable for any damages caused by using or misusing the information, code or methods suggested in this document, and anyone using these methods does so at his/her own risk.

SAP offers no guarantees and assumes no responsibility or liability of any type with respect to the content of this technical article or code sample, including any liability resulting from incompatibility between the content within this document and the materials and services offered by SAP. You agree that you will not hold, or seek to hold, SAP responsible or liable with respect to the content of this document.