Workitems, UWL and Web Dynpro for ABAP

Thomas Kosog
Platinum SAP Workflow Consultant
Topics

- UWL (Universal Worklist) on the Enterprise Portal EP 6.0
- UWL Configuration Steps
- Generate XML File to define Workitems for the UWL
- Advanced Configuration for a User Decision on the UWL
- Design and Program for a Web Dynpro for ABAP Application, so it can be called from a Workitem
- Test Web Dynpro to execute a User Decision and look at the Workflow Log
- Integrate the Web Dynpro for ABAP Application into your Workflow
- Advantages of Web Dynpro for Workflow
Business Task Management is Evolutionary

- Business Workflow
- Universal Worklist
- BPM
- + Collaboration Tasks
- + Guided Procedures
- + Work Item calling Web Dynpro for ABAP

1995
EP 5.0
2003
SAP NetWeaver ‘04
SAP NetWeaver 2004s

Web Dynpro for ABAP is available with SAP NW 2004s
No changes in the Workflow Engine for the Web Dynpro for ABAP approach
No changes in the Workflow Builder for the Web Dynpro for ABAP approach
Universal Worklist

UWL is part of the standard user end role, accessed via Home - Work

**Hint:** To Refresh your lists of tasks, use the drop down menu near the Hide Preview link.

**UWL advantages:** connection to several systems, User Decision with mandatory text, launch Web Dynpro for ABAP (Web Dynpro for ABAP applications should be started from the Portal)
Workflow Steps based on User Decisions

Continue?  End?
Continue?  End?
Continue?  End?
Continue?  End?
Continue?  End?
Continue?  End?

Standard Task
TS00008267
SAP-GUI

Standard Task
TS00008267
UWL
Standard

Custom Task
TS98700001
UWL
SAP GUI
for HTML

Custom Task
TS98700002
Custom
UWL XML
configuration
for user
decision

Custom Task
TS98700003
Custom
UWL XML
configuration
file for
Web Dynpro
Launcher

Start-Date: ______
Initiator:     ______
Reason:
UWL with custom configured User Decision

Select primary WI or WI delegated to you

Display log of all your Workflows

Warning messages

Display log of current Workitem, upload files and create “Ad Hoc” Tasks

USER DECISION with mandatory Text “Reason”

_workflow Attachment

_UWL MEMO_
UWL Configuration Steps

Goal: Workitem in UWL calls a Web Dynpro for ABAP application

1. Build Web Dynpro for ABAP application via transaction SE80
2. Test via transaction SE80 or URL (Logon to SAP is required)
3. Use transaction SWFVISU to prepare standard setting for your Workitem in the UWL
4. Portal UWL Administration
5. Test Workitem from UWL (No extra Logon is required)
6. Advanced Configuration via custom XML file for special UWL features or Portal Wizard
Basic Configuration for the Universal Worklist

Sequence of major steps:
- Portal Setup
- Landscape configuration
- Menu: System Administration / System Configuration / Universal Worklist Administration

Important:
Do not forget to clear the cache and also delete the expired workitems

Result:
Portal will load one XML file, where each task has its own definition
Additional Configuration for the Universal Worklist

Universal Worklist Content Configuration
Click to Administrate Item Types and View Definitions
After configuration changes, SAP recommends clearing the persistent cache on the: Cache Administration Page
Click to Configure Item Types and Customize Views Using a Wizard

Optional Universal Worklist Service Configuration

Welcome to UWL Configuration Wizard
You can use this wizard to customize the UWL configuration. Select one of the following operations
- Define custom attributes and customize the corresponding view
- Define and configure what you want to launch when an item is clicked
- Customize attributes and define what you want to launch when an item is clicked (both of the above options)
- Customize the look of the UWL main page

Set to TRUE to see more Workitem information during Testing
Change settings without custom created XML files. Available with recent Portal Support Pack
At runtime, when you refresh your UWL in the Portal, then the Portal will search through all XML files (or the cache) in the priority sequence from high to low.

In the display on the left, each line is its own XML file.

One Workflow Task can be defined in many XML files.

In the UWL each Workitem will be displayed based on the configuration setting of the corresponding Workflow Task, which was found first.

**Hint:** Upload custom XML files from your PC as “High”
When you download a XML file, you don’t have to delete this XML file in the Portal. Only keep those Item Types in your file, which you want to change and erase the others. Afterwards upload the file with a custom name and the priority “High”.

Approval with optional text

Rejection requires a reason
Let SAP generate a XML file for You

Transaction SWFVISU on your SAP system, where you defined your Workflow

Steps:
- Enter your Task
- Select “ABAP Web Dynpro”
- Enter your Web Dynpro Application
- The namespace is always “SAP”

Hint: Add this configuration to a customizing transport
The generated XML is “ready to go”.
You do not have to change the XML file, which was generated by SAP
Advantage of User Decision for Web Dynpro

- No special Business Object required

- Decision of the Web Dynpro applications can easily be mapped into options of the User Decision, for example:
  - Approved (or ‘Yes’)
  - Rejected (or ‘No’)

- Workflow Builder provides a special step type

- Workflow provides one branch for every option

- Workflow Log shows, which decision was selected by whom
Accessing Workflow Data without a Web Dynpro Application

**Workflow Definition (Design Time)**

- **Step 1**
- **Step 2**
- **Step 3**
  - Task 98700001
  - Binding

**Workflow Container**

**Workflow Runtime**

- **Workflow Instance**
- **Workflow Container (global)**
- **Workitem For Step 3**
  - Load data when WI is created
  - Return data when WI completed
- **Workitem Container (local)**
  - Launch
  - Business Object Method
  - Automatic Key & Parameter Access
Accessing Workflow Data from a Web Dynpro Application

When the task is based on a Web Dynpro, then the application has to read the WI Container and update the WI Container explicitly.

Workflow at runtime

1. Automatic

Data

2. Automatic

Work item ID

3. Automatic

UWL

4.A Automatic

User Decision

4.B Custom Program

5. Automatic

Work item Container

Workflow Container

Work item Container
SAP Web Dynpro uses principles of MVC paradigm

- Model holds business logic, when BAPI calls are used
- Views define
  - field layout & context mapping
  - buttons & actions
- Controller handle the user input
- Window Method HandleStart is called by inbound plug of your Web Dynpro when launched from UWL
Prepare Web Dynpro for ABAP Application for UWL

- Define START As an Inbound Plug
- Declare START as “the” Plug, when application is called
- Provide default For testing

Web Dynpro Event Handler will call assigned Method, when application is launched
Context for Component Controller & Window

- Define Context for the Component Controller
- Define Controller Usage for your window

Drag & Drop Context into your window context (Mapping + Binding)
CALL FUNCTION 'SAP_WAPI_GET_HEADER' 
EXPORTING 
  WORKITEM_ID = l_WI_ID 
IMPORTING 
  WORKITEM_ATTRIBUTES = lv_WORKITEM_ATTRIBUTES 
  RETURN_CODE = rc.

if rc = 0.
  * Load context attribute
  lo_node->set_attribute( 
    value = lv_WORKITEM_ATTRIBUTES WI_TEXT 
    name = 'TEXT' ).
  lo_node->set_attribute( 
    value = lv_WORKITEM_ATTRIBUTES WI_CD 
    name = 'DATE' ).
  endif.

  * Set View-Status to 'ON'
  clear lo_node.
  CALL METHOD wd_context->get_child_node 
  EXPORTING 
    name = 'STATUS' 
  RECEIVING 
    child_node = lo_node.
  lo_node->set_attribute( 
    value = 'X' 
    name = 'FRESH_FLAG' ).

Read Workitem Container and load Context
View Layout based on Context

REASON is dynamically enabled, when FRESH_FLAG = ‘X’ at runtime
Web Dynpro Plugs and Navigation

Window

MAIN-View

Approve or Reject Notification of Absence

Create Action

Update Workitem

Confirmation View

Your vacation has been approved.
Define an Action to navigate to next View

Plugs are called by Actions and enable navigation routes
- Read current field values from Context
- Prepare Workitem Container
- Complete Workitem
- Navigate to next View

CALL FUNCTION 'SAP_WAPR_WORKITEM' 
EXPORTING 
  WORKITEM_ID = l_wi_id 
  DO_COMMIT = 'X' 
IMPORTING 
  NEW_STATUS = lv_NEW_STATUS 
  RETURN_CODE = rc. 
clear l_cont. refresh l_cont.
  l_cont_line-element = 'REASON'.
  l_cont_line-value = lv_reason.
append l_cont_line to l_cont. 
  l_cont_line-element = 'RESULT'.
  l_cont_line-value = '0001'.
append l_cont_line to l_cont. 
  l_cont_line-element = 'WI_Reason'.
  l_cont_line-value = '0001'.
append l_cont_line to l_cont. 
CALL FUNCTION 'SAP_WAPR_WORKITEM_COMPLETE' 
EXPORTING 
  WORKITEM_ID = l_wi_id 
  DO_COMMIT = 'X' 
IMPORTING 
  RETURN_CODE = rc 
  NEW_STATUS = lv_STATUS 
TABLES 
  SIMPLE_CONTAINER = l_cont. 
  wd This->Fire_Decision_Was_Made Plg( 
). 
endmethod.
Test from Transaction SWU3 and SE80

SWU3
- Start Verification Workflow and get Workitem Number

SE80

Workitem number of the user decision step “Verification Workflow” (SWU3)

Clicking on the “Approved” button completed the Workitem of the Verification Workflow via the Web Dynpro for ABAP application
The Workflow Log shows the user who completed the workitem and the Result as assigned in the ACTION-Program.

The Workitem Container shows the data, which was passed from the Web Dynpro, when the function was called to complete the Workitem.
Adjust Workflow for Custom "User Decision" Step

Workflow Builder - Display 'Z_WD4A'

TS98000130 was copied from TS00008267
Custom Business Object Type ZDECISION

Standard Task: Display

- **Name**: WD4A decision task
- **Package**: $TMP

Display Object Type ZDECISION

- **Object type**: ZDECISION

Object Type: Editor Edit Program ZDECISION

- **Line 109**: Stop this method.
- **Line 110**: Workitem may only be completed from Web Dynpro for ABAP
  - EXIT_return 9001 'Workitem may only be launched from Web Dynpro'
  - space space space
- **Line 113**: EXITCANCELLED
- **Line 115**: CALL FUNCTION 'SWU_PROCESS_MANUEL_DECISION'
WD Workitem may only be launched from UWL

- Executing the WI from the SAP GUI transaction SBWP would just call the User Decision, but not the Web Dynpro for ABAP application.

- Selecting one of the User Decision options would complete this workitem.

- To prevent this from happening, put a break in this method-call:
  - Define the subtype ZDECISION of the Business Object Type DECISION.
  - Do *not* define this as a system wide delegation.
  - Redefine the Method PROCESS.
  - Example on previous slide.
  - When this Workitem is now executed from the SAP GUI, then the user gets the error message and the Workitem keeps the status “STARTED”.
**Web Dynpro for ABAP - Advantages**

- ABAP programmers will be able to learn Web Dynpro for ABAP relatively fast, after an initial training and some tutorials.

- The Web Dynpro program is based on standard ABAP, you can use any Table / Field / Data Type, Function, or Object Method.

- Good documentation about “Web Dynpro for ABAP” on SAP SDN

- Web based Front-End for SAP Portal Users

- Integrated in Portal UWL via SAP-Web-Dynpro-ABAP-Launcher

- Existing Workflow Steps can be transformed into Web Pages

- New Workflows can be designed with Web Pages in mind, for example Form-Based Workflows:
  - Initial-Request and Workflow Approval Steps will use the same form with different buttons (“Submit / Cancel” or “Approve / Reject”)
  - Fields can be dynamically enabled
  - After Approval, a Workflow Background Step calling a provided BAPI can create or change SAP data.