Integrate BPM with ABAP WD Chips – An Example

Applies to:
IT architects and developers considering using SAP NetWeaver BPM in heterogeneous landscapes that include systems on the SAP NetWeaver AS ABAP platform (e.g. ECC, SRM, CRM, etc.).

Pre-requisites to apply the approaches explained in this document are:
- SAP NetWeaver AS ABAP 7.0 Enhancement Package 2 SP5 or above
- SAP NetWeaver AS Java 7.3 or above (with the BPM usage types installed)

Summary
This document provides a step by step example of how to create and integrate user interfaces for SAP NetWeaver BPM tasks using ABAP Web Dynpro CHIPs (Collaborative Human Interface Parts).

Why use ABAP Web Dynpro with SAP NetWeaver Business Process Management? It’s a best of both worlds approach in heterogeneous Java/ABAP and SAP/non-SAP landscapes:
- Drawing on the SAP NetWeaver Business Process Management capabilities to create workflows across heterogeneous landscapes
- Creating simplified user interfaces that are easy to adapt and easy to understand utilizing existing ABAP skills
- Extending existing Web Dynpro Components to be used from multiple entry points – from SAP NetWeaver BPM, from the SAP Enterprise Portal, and from direct URLs
- Drawing on existing ABAP system data directly for data visibility and validation prior to update or transfer to other systems (SAP and non-SAP) in the landscape

Author: Jiby Joseph and Jocelyn Dart
Company: SAP Australia
Created on: 12th March 2012

Author Bio
Jiby Joseph is ABAP/Java/SOA consultant with Services in SAP Australia. He has over six years’ experience in SAP related technology.

Jocelyn Dart is BPM/BRM Product Lead and Principal Consultant with Services in SAP Australia. She joined SAP in 1994. Since then she has worked with over 60 customers in a wide-range of industries in project and advisory roles. She is also well known as a co-author of the popular SAP Press book “Practical Workflow for SAP”; presenter at SAP-related conferences; and blogger on the SAP Developer Network.
Table of Contents

Purpose of this Document ........................................................................................................... 4
Pre-requisites .............................................................................................................................. 4
The Example ............................................................................................................................... 4
Steps to Integration ....................................................................................................................... 5
Understanding the ABAP Web Dynpro Example ....................................................................... 6
Understanding Restrictions on the CHIP ...................................................................................... 8
Preparing the Parameters for the CHIP ....................................................................................... 9
  Create Structures and Table Types ............................................................................................ 9
  Create the input Business Partner structure .............................................................................. 9
  Create the output Business Partner Details structure ............................................................. 10
  Create the output Return structure ......................................................................................... 11
  Create the return table type ................................................................................................. 11
Preparing the ABAP Web Dynpro Component with a BPM-ready CHIP .................................... 11
  Create Import Methods ........................................................................................................ 12
  Create Outport Events .......................................................................................................... 13
  Trigger the Outport Events .................................................................................................... 14
  Create and Configure the CHIP ............................................................................................. 15
    Configure the CHIP Inports .................................................................................................. 17
    Configure the CHIP Outports ............................................................................................ 19
Integrate the ABAP Web Dynpro CHIP into the BPM Task ....................................................... 21
  Create the Task ................................................................................................................... 22
  Assign the ABAP Web Dynpro CHIP to the user interface of the Task ................................. 23
One-Time Runtime Configuration of the Logical ABAP System Alias .................................... 27
  Create/Check the System Alias ...........................................................................................</.... 27
  Configure the System Alias .................................................................................................. 27
  Also Check .......................................................................................................................... 28
Executing the scenario ............................................................................................................. 29
Related Content ....................................................................................................................... 30
Copyright .................................................................................................................................. 31
Table of Figures and Tables

Table 1 – ABAP Types permitted on the CHIP........................................................................8
Table 2 - Properties of the Logical ABAP System Alias 1 ..............................................................28
Table 3 - Properties of the Logical ABAP System Alias 2 ................................................................28

Figure 1 - The desired BPM Task - with the Web Dynpro ABAP screen in the Task Application pane of the BPM Task ..................................................5
Figure 2 – ABAP Web Dynpro View Part 1 - Search a Business Partner .....................................6
Figure 3 – ABAP Web Dynpro View Part 2 – Business Partner Details ..........................................7
Figure 4 - Input Structure .................................................................................................................9
Figure 5 - Successful Completion Output Structure ........................................................................10
Figure 6 – Error/Failure/Cancellation Output Line Structure ...........................................................11
Figure 7 - Add an Interface method for the Inport ........................................................................12
Figure 8 - Add the Inport parameters to the Inport method ..............................................................12
Figure 9 - Inport method SET_BP_DETAILS updating the Component Controller Context ..............13
Figure 10 - Add the Interface events for the successful completion Outport .................................13
Figure 11 - Add the Interface events for the unsuccessful Outport ..................................................14
Figure 12 – Buttons for Outport Events ..........................................................................................14
Figure 13 - Create the CHIP ...........................................................................................................16
Figure 14 - Create the Inbound Port ................................................................................................17
Figure 15 - Access the Tags and Parameters of the Port .................................................................17
Figure 16 - Assign the Inport Tag ....................................................................................................17
Figure 17 - Assign the Inport Parameters .........................................................................................18
Figure 18 - Create the Outbound Ports ............................................................................................19
Figure 19 - Access the Outbound Ports ...........................................................................................19
Figure 20 - Assign the Outport Tag ................................................................................................19
Figure 21 - Assign the Outport Parameters ......................................................................................20
Figure 22 - Outport configuration .....................................................................................................20
Figure 23 - BPM Human Activity in a BPM Process .......................................................................21
Figure 24 - Creating a BPM Task ....................................................................................................21
Figure 25 - Create an empty BPM Task ..........................................................................................22
Figure 26 - Accessing the User Interface Wizards ........................................................................23
Figure 27 - Select the ABAP Web Dynpro Wizard .........................................................................23
Figure 28 - Search for ABAP Web Dynpro CHIPS .........................................................................24
Figure 29 - Checking the correct CHIP has been identified .............................................................25
Figure 30 - Confirm the Completion and Error Events ..................................................................25
Figure 31 - ABAP Web Dynpro as User Interface for BPM ...............................................................26
Figure 32 - Task Mapping of BPM Activity showing CHIP parameters .........................................27
Figure 33 - Logical ABAP System Alias of BPM Task ..................................................................27
Figure 34 - The final BPM Task as it appears to the end user ............................................................29
Purpose of this Document

The purpose of this document is to give a simple step by step example for a developer who wants to integrate SAP NetWeaver Business Process Management (BPM) with ABAP Web Dynpro, i.e. using ABAP Web Dynpro as the user interface of a BPM Task. This requires the ABAP Web Dynpro to implement a CHIP (Collaborative Human Interface Part). This document includes detailed screenshots also and a brief explanation of the core concepts.

A CHIP (Collaborative Human Interface Part) is a small, widget-type, encapsulated, stateful piece of software that can be combined in a layout with other CHIPS to form a page or side panel. When used as part of a BPM Task, the CHIP is combined into the content pane of the task.

The example will focus on the necessary steps to extend a Web Dynpro ABAP Component for use as the user interface of a BPM Task therefore not all parts of the Web Dynpro ABAP or the BPM Process will be shown.

Pre-requisites

It is assumed the developer is already well versed in ABAP Web Dynpro concepts and programming. The developer should be able to create a simple ABAP Web Dynpro Component including a simple form-like View showing some fields, and containing Approve/Reject buttons.

A basic understanding of SAP NetWeaver BPM is assumed, i.e. how to create a process; create a task; deploy and run a process.

Minimum releases/support packs:

- SAP NetWeaver AS ABAP 7.0 Enhancement Package 2 SP5 or above
- SAP NetWeaver AS Java 7.3 or above (with the BPM usage types installed)

The Example

Please Note: All screenshots are based on a SAP NetWeaver AS Java release 7.3, and an ECC 6.0 EHP5 ABAP system.

In this scenario, there is an existing SAP NetWeaver Business Process Management Process. The Process Owner has decided the process needs to be changed to include a step where someone checks the details of the business partner number selected automatically or by someone else in a previous step, perhaps working in a parallel system. This is a typical master data process scenario in an organization that does not yet have a master data management system to synchronize data across all of its parallel systems.

Because the business partner details are held in a SAP ECC system, it makes sense to create the user interface in the ECC system via ABAP Web Dynpro, where the data exists and the necessary information can be readily displayed.

In the scenario, there is an existing ABAP Web Dynpro Component that displays business partner details used elsewhere, e.g. from within the SAP Enterprise Portal. This ABAP Web Dynpro Component is to be extended for use within a SAP NetWeaver Business Process Management Task.

When the user opens the BPM Task from their Universal Worklist, we want them to see the Web Dynpro Application view showing within the Task Application pane of the BPM Task. The standard BPM Task Data will still appear at the top, giving the user access to the Process Log, and standard BPM actions (e.g. Delegate, Put Back) and views (e.g. Attachments, Notes, Collaboration). Within the application pane, the Web Dynpro ABAP application can behave as needed, however as the most common scenario is a simple one-screen application we will use this approach for the example.
Integrate BPM with ABAP WD Chips

Steps to Integration

The following steps are needed to integrate ABAP Web Dynpro as the user interface of a SAP NetWeaver Business Process Management task:

1. **Create the Process in the Process Development perspective of SAP NetWeaver Developer Studio**
2. **Create/Extend the ABAP Web Dynpro Component**
   a. Create/Extend the Component Controller and any context data needed
   b. Create/Extend the View with the desired layout and buttons
   c. Create/Extend methods, events, etc. as needed to support the desired behavior of the View

Creation of the Process and of the ABAP Web Dynpro Component can be done in parallel by different developers. Until the ABAP Web Dynpro Component is ready the Process can use a Default Task as a placeholder.

However some understanding of the parameters to be passed from the final task to the ABAP Web Dynpro and vice versa is helpful at this stage so that appropriate Data Objects can be prepared. It is helpful to understand the restrictions on the CHIP at the design stage.

**Note:** The actual creation of contexts in the component controller and mapping to the context in the view controller are not shown in this example. It is assumed the developer has basic understanding on how to create ABAP Web Dynpro and carry out the mapping from view controller to component controller. The business logic on how to get
the data is not shown. An example screen shot of the ABAP Web Dynpro will be shown in the later section so that reader can understand the flow.

The final steps are:

3. **Add the BPM-ready CHIP to the ABAP Web Dynpro component**
4. **Integrate the BPM Task with the created ABAP Web Dynpro CHIP**

These steps must be performed in sequence. That is, it is not possible to integrate the CHIP into the BPM Task until it physically exists.

If this is the very first time the ABAP system has been integrated with SAP NetWeaver Business Process Management there is an additional step.

5. **Configure the runtime system alias (one-time only)**

Once all steps have been completed the task can be tested by deploying and executing its parent process. This document will explain steps 3, 4 and 5. In the final section we will show the task as it will appear at runtime when opened by the end user.

The first 2 steps are explained in other tutorials such as:

* SCN Article – Create a Process Driven Composite Application (tutorial)*
* SCN Article – Create a Simple Web Dynpro Application*

### Understanding the ABAP Web Dynpro Example

ABAP Web Dynpro components can be created or extended to be used as a BPM Task of a BPM Business Process. In this document, we extend an existing ABAP Web Dynpro view to confirm the business partner. This view is a simple ABAP Web Dynpro form which contains 2 main sections:

At the top of the view are the incoming details passed from the BPM Task, i.e. the provided Business Partner Number, Name and Address, and a button Check Selected Business Partner.

The user will use these to try to determine the correct business partner.

These details may be provided in part or in full.

The Business Partner Number is editable and has a search help on it so that the user can search for appropriate business partner numbers.

![Search for a Business Partner](image)

*Figure 2 – ABAP Web Dynpro View Part 1- Search a Business Partner*
After the user has clicked on the button *Check Selected Business Partner*, the Business Partner Details of the selected business partner will be shown.

<table>
<thead>
<tr>
<th>Business Partner Details</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Business Partner Number:</strong> 0000000172</td>
</tr>
<tr>
<td><strong>First Name:</strong> FI</td>
</tr>
<tr>
<td><strong>Last Name:</strong> CUSTOMER 2</td>
</tr>
<tr>
<td><strong>Full Name:</strong> Rector FI CUSTOMER 2</td>
</tr>
<tr>
<td><strong>Title:</strong></td>
</tr>
<tr>
<td><strong>Gender:</strong> Female</td>
</tr>
<tr>
<td><strong>Birth Date:</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Address Details</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>c/o:</strong></td>
</tr>
<tr>
<td><strong>Street:</strong> St Kilda Road</td>
</tr>
<tr>
<td><strong>City:</strong> MELBOURNE</td>
</tr>
<tr>
<td><strong>District:</strong></td>
</tr>
<tr>
<td><strong>Region:</strong> VIC</td>
</tr>
<tr>
<td><strong>Country:</strong> AU</td>
</tr>
<tr>
<td><strong>Postal Code:</strong> 3004</td>
</tr>
</tbody>
</table>

If the user is satisfied that the correct business partner has now been found, they will confirm this by using the button *Business Partner Found*.

If they are not satisfied they can select a different business partner and search again. If they still cannot determine an appropriate business partner they can indicate this by clicking using the button *Business Partner Not Found*.

When either of the buttons are selected, the ABAP Web Dynpro is closed, the BPM task ends and the BPM Process continues.
Understanding Restrictions on the CHIP

CHIPs usually consist of:

- Inports with associated tags, incoming parameters and interface methods to handle them
- Outports with associated tags, outgoing events and interface events to fire them

When creating a CHIP to integrate to a SAP NetWeaver Business Process Management task, the CHIP must consist of:

- One and only one Inport and its associated Inport method
- At least one Outport:
  - One “successfully completed” Outport
  - Any other outports needed to notify of exceptions, failures or cancellations
- The specific tags `PROCESS_STEP_START_POINT` for the Inport and `PROCESS_STEP_END_POINT` for the Outports.
  - Only CHIPs with these tags can be integrated to SAP NetWeaver Business Process Management
- Parameters that are either: single values, a single structure or a single table type per in/outport.
  - Complex structures, e.g. tables within structures, are not permitted
  - In addition, individual values, elements of structures or tables are only permitted to use 3 standard ABAP Types are permitted as shown below.

<table>
<thead>
<tr>
<th>ABAP Type</th>
<th>BPM Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>STRING</td>
<td>xsd:string</td>
</tr>
<tr>
<td>INT4</td>
<td>xsd:int</td>
</tr>
<tr>
<td>WDY_BOOLEAN</td>
<td>xsd:boolean</td>
</tr>
</tbody>
</table>

Where conversion to additional types is required, this should be handled through the mapping options in SAP NetWeaver Business Process Management, e.g. using the built-in mapping functions.

**Tip:** The intention here is to pass only the critical information that needs to be passed from the BPM Process to start the ABAP Web Dynpro, and the results from the ABAP Web Dynpro to the BPM Process that will be passed to subsequent steps of the context. Passing large amounts of data can potentially have a negative impact on performance.

As SAP NetWeaver Business Process Management is a usage type of a SAP NetWeaver AS Java system, which is separate to the ABAP system that hosts the ABAP Web Dynpro Component, the CHIP must also be remote-enabled.
Preparing the Parameters for the CHIP

Create Structures and Table Types

As part of this exercise, we need to create three structures and one table type in the ABAP Dictionary, i.e. transaction SE11 of the ABAP system where the ABAP Web Dynpro Component has been created.

The first structure is to take a business partner number, name and address as input. These are optional parameters – some or all of these may be passed from the SAP NetWeaver Business Process Management task to the ABAP Web Dynpro Component.

The other structures are used to return information from the ABAP Web Dynpro Component to the SAP NetWeaver Business Process Management task.

The second structure is for the output of business partner details, and the third structure is a return message. The table type is used for the return message as we may need to return multiple messages.

Create the input Business Partner structure

![Input Structure](image)

Figure 4 - Input Structure
Create the output Business Partner Details structure

<table>
<thead>
<tr>
<th>Component</th>
<th>Typing Method</th>
<th>Component Type</th>
<th>Data Type</th>
<th>Length</th>
<th>Dec.</th>
<th>Short Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FIRST_NAME</td>
<td>Types</td>
<td>▼</td>
<td>STRING</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LAST_NAME</td>
<td>Types</td>
<td>▼</td>
<td>STRING</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FULL_NAME</td>
<td>Types</td>
<td>▼</td>
<td>STRING</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BIRTHNAME</td>
<td>Types</td>
<td>▼</td>
<td>STRING</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MIDDLENAME</td>
<td>Types</td>
<td>▼</td>
<td>STRING</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SECONDNAME</td>
<td>Types</td>
<td>▼</td>
<td>STRING</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TITLE_ACA1</td>
<td>Types</td>
<td>▼</td>
<td>STRING</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TITLE_ACA2</td>
<td>Types</td>
<td>▼</td>
<td>STRING</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TITLE_SPL</td>
<td>Types</td>
<td>▼</td>
<td>STRING</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PREFIX1</td>
<td>Types</td>
<td>▼</td>
<td>STRING</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PREFIX2</td>
<td>Types</td>
<td>▼</td>
<td>STRING</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NICKNAME</td>
<td>Types</td>
<td>▼</td>
<td>STRING</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>INITIALS</td>
<td>Types</td>
<td>▼</td>
<td>STRING</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NAMEFORMAT</td>
<td>Types</td>
<td>▼</td>
<td>STRING</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NAMCOUNTRY</td>
<td>Types</td>
<td>▼</td>
<td>STRING</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NAMCOUNTRYISO</td>
<td>Types</td>
<td>▼</td>
<td>STRING</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SEX</td>
<td>Types</td>
<td>▼</td>
<td>STRING</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BIRTHPLACE</td>
<td>Types</td>
<td>▼</td>
<td>STRING</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BIRTHDATE</td>
<td>Types</td>
<td>▼</td>
<td>STRING</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DEATHDATE</td>
<td>Types</td>
<td>▼</td>
<td>STRING</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MARITALSTATUS</td>
<td>Types</td>
<td>▼</td>
<td>STRING</td>
<td>0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 5 - Successful Completion Output Structure
Create the output Return structure

![Figure 6 – Error/Failure/Cancellation Output Line Structure](image)

Create the return table type

### Preparing the ABAP Web Dynpro Component with a BPM-ready CHIP

The following need to be added to the existing ABAP Web Dynpro Component before the CHIP Contract can be created:

- An Import Method added to the Component Controller to handle the inport parameters, e.g. to update the Component Controller context, and navigate to the appropriate View
- Outport Events added to the Component Controller to send the outport parameters, i.e. one event per outport
- Methods added to the Component Controller to fire the Outport Events
- An Interface View and plug (e.g. the Default plug), used to navigate to the appropriate Window and View at runtime

The following needs to be added to the ABAP Web Dynpro View:

- Buttons (or equivalent element) and associated view actions/methods that call the component controller methods that fire the outport events
  - Alternatively existing buttons (or equivalent element) and associated view actions/methods can be extended

At runtime, when the user opens the BPM Task, the CHIP will be called to take the user to the relevant View. The user selects the buttons on the View to indicate they have completed the work and that the task should now be closed.
Create Import Methods

An Import method is an Interface method that handles the import parameters when the CHIP is called, i.e. when the SAP NetWeaver Business Process Management task is opened by the user at runtime. The signature of the import method is the import parameters.

For example, create a method in the Component Controller called SET_BP_DETAILS. Make sure the checkbox Interface is selected. In this example we’ll use this method to default the display data based on the proposed business partner number, address and name details.

![Figure 7 - Add an Interface method for the Import](image)

Once this is done, we need to create the signature for the method. The method takes in the business partner details structure created earlier, as the input parameter.

![Figure 8 - Add the Import parameters to the Import method](image)

When the CHIP has been called, the import method handles the incoming parameters and determines what happens next. Usually all that we need to do in the import method is update the Context of the Component Controller. We could also default some display data based on the incoming parameters.

**Tip:** We only need to update the Component Controller Context as we use the Interface View of the CHIP to navigate to the appropriate Window and View at runtime, and the Window and View will derive their context from the Component Controller.

Below is an extract of the code from SET_BP_DETAILS showing the Component Controller Context being updated from the incoming parameters:

```
@componentcontroller
@method
    SET_BP_DETAILS
    
    @input
        BP_Details
    
    @output
        BP_Details

```
Create Output Events

To return data from the ABAP Web Dynpro to the SAP NetWeaver Business Process Management task, we need to create at least one Interface event representing successful completion. Any other events are used to return exceptions/failures/cancellations.

In this example, we create two events. The first event will be called if the business logic call was successful and the second event will be called if the business logic call was unsuccessful. So that we can easily complete the handshake with BPM, it’s helpful if the event name clearly indicates whether it is a successful completion or an error event.

Create an event called TRIGGER_SUCCESS with the business partner details structure as its signature. Make sure the Interface check box is selected.

Create an event called TRIGGER_ERROR with the returned messages table as its signature. Make sure the Interface check box is selected.
Integrate BPM with ABAP WD Chips – An Example

Figure 11 - Add the Interface events for the unsuccessful Outport

Trigger the Outport Events

Search for a Business Partner

<table>
<thead>
<tr>
<th>Business Partner Number: 172</th>
<th>Provided Business Partner Name: John Smith</th>
</tr>
</thead>
<tbody>
<tr>
<td>Provided Business Partner Address: 168 Walker Street North Sydney NSW 2060</td>
<td></td>
</tr>
</tbody>
</table>

Check Selected Business Partner

Business Partner Details

<table>
<thead>
<tr>
<th>Business Partner Number: 000000172</th>
<th>First Name: Fi</th>
</tr>
</thead>
<tbody>
<tr>
<td>Last Name: CUSTOMER 2</td>
<td>Full Name: Rector Fi CUSTOMER 2</td>
</tr>
<tr>
<td>Title:</td>
<td></td>
</tr>
<tr>
<td>Gender: Female</td>
<td>Birth Date:</td>
</tr>
</tbody>
</table>

Address Details

<table>
<thead>
<tr>
<th>c/o:</th>
<th>Street: St Kilda Road</th>
</tr>
</thead>
<tbody>
<tr>
<td>City: MELBOURNE</td>
<td>District:</td>
</tr>
<tr>
<td>Region: VIC</td>
<td>Country: AU</td>
</tr>
<tr>
<td>Postal Code: 3004</td>
<td></td>
</tr>
</tbody>
</table>

Business Partner Found

Business Partner Not Found

Figure 12 – Buttons for Outport Events

The two highlighted buttons in the web dynpro we created above are used to fire the outport events.

Create Component Controller methods to fire the outport events:
• If the button *Business Partner Found* is selected, set up the output parameters and call the method `fire_trigger_success`.

• If the button *Business Partner Not Found* is selected, set up the output parameters and call the method `fire_trigger_error`.

Make sure that the buttons are associated to actions/methods in the view controller which in turn calls the component controller methods that fire the outport events.

```
method SUCCESS .

data lo_nd_bp_details type ref to if_wd_context_node.
data lo_el_bp_details type ref to if_wd_context_element.
data ls_bp_details type wd_this->element_bp_details.

* Set up the Output Parameters
  lo_nd_bp_details = wd_context->get_child_node( name = wd_this->wdctx_bp_details ).
  lo_el_bp_details = lo_nd_bp_details->get_element( ).
  lo_el_bp_details->get_static_attributes( 
    importing
    static_attributes = ls_bp_details ).

* Fire the event
  wd_this->fire_trigger_success_evt( exporting es_bp_person_details = ls_bp_details ).
endmethod.
```

Similarly the method to fire the error event might look like this:

```
method ERROR .

data: lt_error type ztab_message,
  ls_error type line of ztab_message,
  lo_nd_selected_bp type ref to if_wd_context_node,
  lo_el_selected_bp type ref to if_wd_context_element,
  ls_selected_bp type wd_this->element_selected_bp,
  lv_selected_bp_number type wd_this->element_selected_bp->selected_bp_number.

* Set up the Output Parameters
  lo_nd_selected_bp = wd_context->get_child_node( name = wd_this->wdctx_selected_bp ).
  lo_el_selected_bp = lo_nd_selected_bp->get_element( ).
  lo_el_selected_bp->get_attribute( 
    exporting
    name = 'SELECTED_BP_NUMBER'
    importing
    value = lv_selected_bp_number ).

* Send out Message: Business Partner & not found
  ls_error->type = 'E'.
  ls_error->ID = 'Z_BPM_CHIP_INT'.
  ls_error->number = '000'.
  ls_error->message_v1 = lv_selected_bp_number.
  append ls_error to lt_error.

* Fire the event
  wd_this->fire_trigger_error_evt( exporting et_message = lt_error ).
endmethod.
```

**Create and Configure the CHIP**

Create a Web Dynpro Chip using the context menu (right-click) of the Web Dynpro Component. Choose Create and then select Web Dynpro Chip.
The **Display Name** will be used to find the correct CHIP when configuring the SAP NetWeaver Business Process Management task. This should be business meaningful and sufficiently specific to ensure the correct CHIP is identified.

Select the Interface View and Plug that will navigate to the appropriate Window and View.

Set the **Loading Application** field to WDR_CHIP_LAUNCHER.

Make sure the checkbox **Remote-enabled CHIP** is selected.

Save the new CHIP.

![Web Dynpro CHIP](image)

**Figure 13 - Create the CHIP**

Once the CHIP itself has been created, we need to wire the CHIP to the inport method and outport events we created previously.
Configure the CHIP Inports

On the Inports tab of the CHIP, we need to specify our Inport, tag, and inport parameters.

In the Inbound Ports section we select the Create button, then select the Interface method SET_BP_DETAILS method as the inport and press Enter.

![Figure 14 - Create the Inbound Port](image1)

An Inbound Port has been created. We select the port then select Show Tags to view the tags.

![Figure 15 - Access the Tags and Parameters of the Port](image2)

Once, this is done we need to create two more items to complete the Inport configuration. The first step is to assign the tag to the Inport.

We use the Create button in the section Tags of Port SET_BP_DETAILS to assign the tag CHIP_CONTRACT:PROCESS_STEP_START_POINT.

![Figure 16 - Assign the Inport Tag](image3)
The second item is to specify the Inport parameters. We select the Create button in the section **Parameters of port SET_BP_DETAILS** and then confirm the parameters by selecting the BP_NUMBER, BP_NAME, BP_ADDRESS fields.

**Figure 17 - Assign the Inport Parameters**
Configure the CHIP Outports

Lastly we need to create the Outports on the Outports tab.

In the Outbound Ports section, we select the Create button, select all the interface events and click Enter to confirm.

![Figure 18 - Create the Outbound Ports](image)

The Outbound ports should look like this:

![Figure 19 - Access the Outbound Ports](image)

As with the inport, we need to do two more things, one is to create the tag for the Outport and the other is to create the parameters for the Outport. We need to create these for each Outport.

Select the TRIGGER_SUCCESS Outport and the Show Tags button. In the Tags of Port TRIGGER_SUCCESS section, we create a new tag but this time search for and assign the tag CHIP_CONTRACT:PROCESS_STEP_END_POINT.

![Figure 20 - Assign the Outport Tag](image)

In the section Parameters of Port TRIGGER_SUCCESS, we select Create, add all the selected outgoing parameters and press Enter.
Figure 21 - Assign the Outport Parameters

The outport should now look something like this:

We repeat these steps for the TRIGGER_ERROR Outport. That is, search for and assign the tag CHIP_CONTRACT:PROCESS_STEP_END_POINT, and then add the outgoing parameters relevant to this Outport – i.e. the fields of the Return messages table type parameter.

Finally we save the CHIP. The CHIP is now complete.
Tip: So long as the CHIP itself does not need to change we can continue to make further changes to the ABAP Web Dynpro Component.

For the final step we move out of the ABAP system and back to SAP NetWeaver Business Process Management.

**Integrate the ABAP Web Dynpro CHIP into the BPM Task**

Development for SAP NetWeaver Business Process Management is performed in the SAP NetWeaver Developer Studio (NWDS). It is here that we assign the ABAP Web Dynpro CHIP as the user interface of our BPM Task.

Assigning the ABAP Web Dynpro CHIP is performed in the Process Development perspective of SAP NetWeaver Developer Studio. A human activity needs to be created in the BPM Process to indicate at which step of the process the BPM Task will be used.

![Figure 23 - BPM Human Activity in a BPM Process](image)

In the Properties of the Human Activity you start the creation of a new task to which the user interface will be assigned:

![Figure 24 - Creating a BPM Task](image)
Create the Task

You create a new, empty task with an appropriate name, e.g. “Confirm Business Partner”. Make sure that the checkbox **Generate UI Component** is NOT checked, and then select **Finish**.

![Create an empty BPM Task](image)

**Figure 25 - Create an empty BPM Task**

The task overview screen now appears.
Assign the ABAP Web Dynpro CHIP to the user interface of the Task

In the User Interface tray, you select Choose… to access the user interface selection wizards.

In the Select a UI Wizard window, select Online Technologies (Web Dynpro Java, Web Dynpro ABAP, Visual Composer).

Figure 26 - Accessing the User Interface Wizards

Figure 27 - Select the ABAP Web Dynpro Wizard
In the **Select a UI Component** screen, choose the UI Technology option **Web Dynpro ABAP CHIP from:** and select your ABAP system from the list of ABAP systems.

![Select a UI Component screen](image)

**Figure 28 - Search for ABAP Web Dynpro CHIPS**

**Tip:** The list of ABAP systems can be preset via the menu Window > Preferences of your NWDS, or use the link Configure R/3 Destinations to update your NWDS preferences on the spot.

Press **Search** to start the search for your Web Dynpro ABAP CHIP.

Because the **Show supported components only** checkbox is checked, only Web Dynpro ABAP CHIP interfaces that are valid for BPM will be shown. The Choose UI component pane will show the CHIP Name.

**Tip:** If you can’t find your CHIP application, try searching again but this time with **Show supported components only NOT** checked. It will show all activated CHIP interfaces and give a reason why they are not suitable. Common problems are:

- Not enough or too many inports/outports;
- **Unsupported data types** in the inport/outport parameters;
- Process step tags **not assigned** to the inport/outport.
You can confirm you have found the correct CHIP, by expanding the CHIP to see its inports/outports and selecting any of the inports/outports to see the related parameters in the Input/Output details pane.

Select the CHIP in the **Choose UI Component** pane and select **Next**.

![Figure 29 - Checking the correct CHIP has been identified](image)

Lastly confirm which outport is the Completion Event for successful completion and which outports are to be treated as Error Events for handling exceptions/failures/cancellations, and select **Finish**.

![Figure 30 - Confirm the Completion and Error Events](image)

**Tip:** If you change your mind later and want to add more boundary events, you will need to run the UI Wizard again, so as a rule of thumb, it’s best to add all your error/cancellation/failure outports as Error Events.
whenever you add a ABAP Web Dynpro CHIP. The error events won’t be used as Boundary Events unless the process developer specifically assigns them to as Boundary Events of the task itself.

Your task overview should show the selected CHIP and system something like this:

![Figure 31 - ABAP Web Dynpro as User Interface for BPM](image-url)

**Figure 31 - ABAP Web Dynpro as User Interface for BPM**
All of the input and output parameters will now appear in the Task Mapping of the parent BPM Human Activity for the task.

![Figure 32 - Task Mapping of BPM Activity showing CHIP parameters](image)

Finally at this stage, in the BPM Task User Interface section you can change the System Alias to reflect the System Alias to be used at runtime, i.e. the system alias defined on the SAP Enterprise Portal hosting the BPM Task. The system alias should be a logical system alias that is pointed to the appropriate ABAP system as the BPM Process is migrated from development to test to production environments.

![Figure 33 - Logical ABAP System Alias of BPM Task](image)

**One-Time Runtime Configuration of the Logical ABAP System Alias**

If this is the very first time you have used a ABAP Web Dynpro CHIP from this ABAP system, you will need to configure the Logical ABAP System Alias.

**Create/Check the System Alias**

If you have assigned a Web Dynpro ABAP CHIP to a BPM Task, you need to specify the remote system from which the UI is loaded. To do that, you specify the necessary system and system alias in the SAP Enterprise Portal hosting the BPM Task.

**IMPORTANT**: The system alias you define in the portal must match the System Alias defined on the User Interface section of the BPM Task and vice versa. If you have not defined the System Alias correctly on the Enterprise Portal when the task opens a Portal Runtime Error message will be shown.

For more information about configuring systems in the portal, see [Defining the System Landscape](http://help.sap.com/saphelp_nw73/helpdata/en/48/8eab6e86ba3cb8e10000000a42189d/frameset.htm).

**Configure the System Alias**

In addition, you need to specify in the portal the following properties of the remote system:
### Table 2 - Properties of the Logical ABAP System Alias 1

<table>
<thead>
<tr>
<th>Category: Web Application Server (WAS)</th>
<th>Property Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ICM host name</td>
<td>&lt;host name&gt;:&lt;port&gt;</td>
</tr>
<tr>
<td></td>
<td>ICM protocol</td>
<td>http or https, depending on the ABAP back-end system</td>
</tr>
</tbody>
</table>

**Tip:** You can find the correct values for these properties in your ABAP system using transaction SMICM, and go to the Services section.

### Table 3 - Properties of the Logical ABAP System Alias 2

<table>
<thead>
<tr>
<th>Category: User Management</th>
<th>Property Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Authentication Ticket Type</td>
<td>&lt;empty&gt;</td>
</tr>
<tr>
<td></td>
<td>Logon Method</td>
<td>UIDPW</td>
</tr>
<tr>
<td></td>
<td>User Mapping Type</td>
<td>admin,user</td>
</tr>
</tbody>
</table>

**Also Check**

Make sure the service WDR_CHIP_LAUNCHER is activated in the ABAP system (transaction SICF).

Make sure the Portal user is mapped to a user in the ABAP system.
Executing the scenario

When the task is executed at runtime, launching the task from the inbox will pass the parameters from the task to the ABAP Web Dynpro CHIP, triggering the import methods.

The ABAP Web Dynpro CHIP will appear in the BPM Task Application pane like this:

![Confirm Business Partner Task]

Clicking on Business Partner Found will trigger the successful completion event, pass the output parameters to the BPM Process and close the task. The task will then be removed from the user’s inbox and the BPM Process continues to the next activity.

Figure 34 - The final BPM Task as it appears to the end user
Related Content

SAP Help - Web Dynpro Chips
SAP Help – SAP NetWeaver BPM Task User Interfaces with Web Dynpro ABAP
SCN Article – Create a Process Driven Composite Application (tutorial)
SCN Article – Create a Simple Web Dynpro Application
Integrate BPM with ABAP WD Chips – An Example

Copyright
© Copyright 2012 SAP AG. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.

Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.

Microsoft, Windows, Excel, Outlook, and PowerPoint are registered trademarks of Microsoft Corporation.

IBM, DB2, DB2 Universal Database, System i, System i5, System p, System p5, System x, System z, System z10, System z9, z10, z9, iSeries, pSeries, xSeries, zSeries, eServer, z/VM, z/OS, i5/OS, S/390, OS/390, OS/400, AS/400, S/390 Parallel Enterprise Server, PowerVM, Power Architecture, POWER6+, POWER6, POWER5+, POWER5, POWER, OpenPower, PowerPC, BatchPipes, BladeCenter, System Storage, GPFS, HACMP, RETAIN, DB2 Connect, RACF, Redbooks, OS/2, Parallel Sysplex, MVS/ESA, AIX, Intelligent Miner, WebSphere, Netfinity, Tivoli and Informix are trademarks or registered trademarks of IBM Corporation.

Linux is the registered trademark of Linus Torvalds in the U.S. and other countries.

Adobe, the Adobe logo, Acrobat, PostScript, and Reader are either trademarks or registered trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Oracle is a registered trademark of Oracle Corporation.

UNIX, X/Open, OSF/1, and Motif are registered trademarks of the Open Group.

Citrix, ICA, Program Neighborhood, MetaFrame, WinFrame, VideoFrame, and MultiWin are trademarks or registered trademarks of Citrix Systems, Inc.

HTML, XML, XHTML and W3C are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.

Java is a registered trademark of Oracle Corporation.

JavaScript is a registered trademark of Oracle Corporation, used under license for technology invented and implemented by Netscape.

SAP, R/3, SAP NetWeaver, Duet, PartnerEdge, ByDesign, SAP Business ByDesign, and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and other countries.

Business Objects and the Business Objects logo, BusinessObjects, Crystal Reports, Crystal Decisions, Web Intelligence, Xcelsius, and other Business Objects products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of Business Objects S.A. in the United States and in other countries. Business Objects is an SAP company.

All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only. National product specifications may vary.

These materials are subject to change without notice. These materials are provided by SAP AG and its affiliated companies ("SAP Group") for informational purposes only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.