Raising an Event Through a Process Chain to Trigger a Dependent Chain on a Specific Day

Applies to:

SAP BI/BW 3.5 and above.

Summary

The document will demonstrate step by step solution to of raising an event of any given day through a parent process chain and trigger another dependent process chain.

Author: Umesh Pednekar

Company: Howdens Joinery Co.

Created on: 19 September 2011

Author Bio

Umesh is working as a SAP BI/ BW Analyst in Howdens Joinery co and has an techno-functional experience of over 7 years in designing, developing and implementing/upgrading SAP BW-BI solutions across various Industries in India and in the UK.

Table of Contents

Overview

Purpose of this document is to demonstrate a step by step solution that will help understand the mechanism involved triggering process chain/chains on a desired day. The document shows how dependent process chain/ chains can be triggered post an event takes place which in turn can be raised as desired. The event raising is done through the parent chain.

Execution of process chain , in particular those chain which depend on completion of specific processes included in preceding / other chains, using events helps establish a process between the correlating chains that helps deliver desired results as well as helps monitor the errors and take corrective action in executing the correlated chains.

The entire set of activities involved is performed on the BW systems.

Assumptions

The document assumes that the target audience is aware of the processes that involve
- Creating Process Chains

Associated references to the above have been provided in the 'Related Content' section of this document.

Step by Step Solution

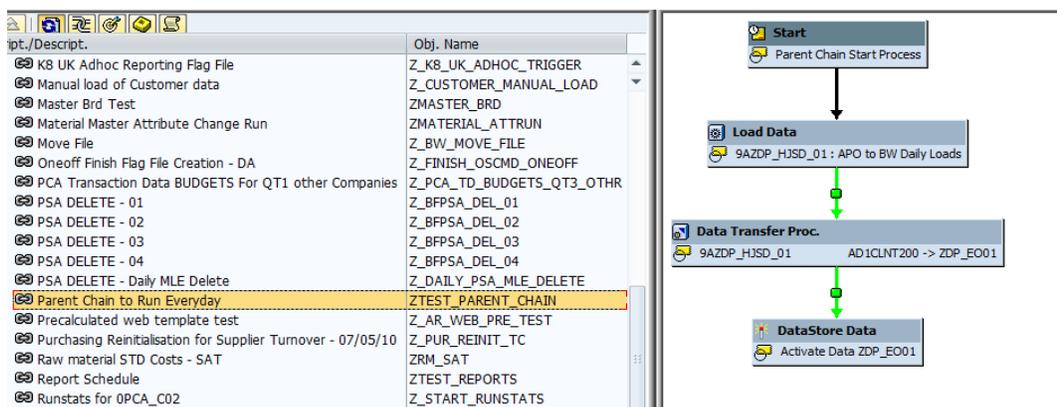The steps involved in the Event based triggering of a Process Chain are
1. Identifying the parent process chain that would help raise an event and the dependent chain that needs to triggered post the Event raise.
2. Creating an Event.
3. ABAP program to raise the Event.
4. Creating variant for the program – the variant will contain parameters viz. Event details, Day of trigger etc.
5. Editing relevant process chains and incorporating the relevant processes.
6. Process chains execution results

Identifying the Parent and Dependent process chains – Parent chain to raise the event and Dependent chain – that would get triggered

Let us consider a scenario that a process chain is executed on a daily basis.   This process chain has to trigger execution of another chain and this execution should be recurring process happening only during a specific day of the week. Let's consider this day as 'Friday'.

The parent chain is our example is a chain that enables data extraction from a source system into BW. The dependent chain is a chain that broadcasts a report.

Original Parent Chain:

Original Dependent Chain:

The dependent chain is original scheduled to run every Friday , however, we want this schedule to be a event based schedule and not time-based (as the preceding or the parent chain required for this chain may encounter a failure thus making the execution of the dependent chain a redundant act).
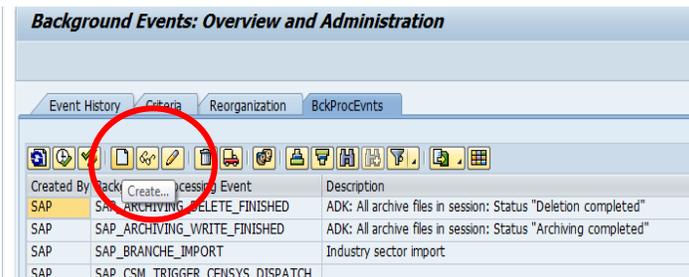
Create an Event

Transaction SM62 enables us to create an event.
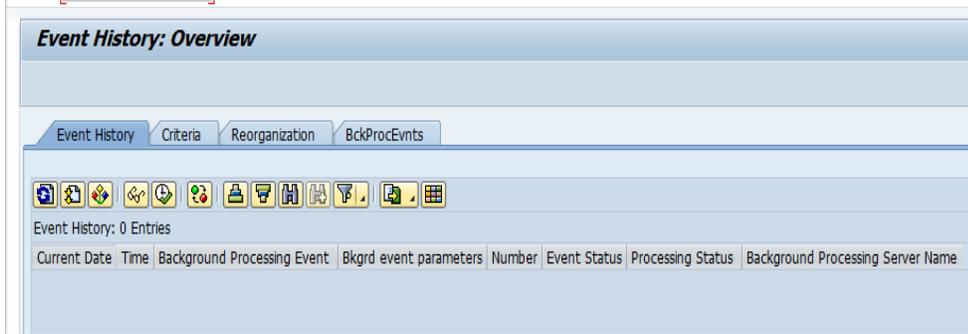
Select 'BckProcEvnts' tab which displays the list of available events.

Select 'Create' to create a new event

At the next prompt, enter a technical name and description for the event.

'Save' the entry. **ZDP_FRIDAY_BROADCAST'** is now available in the Event History

ABAP Program to raise the Event Created Earlier

We will now write an ABAP program that will help raise the event created. Go to transaction SE38 and create a custom ABAP program . We are creating a program named **'Z_EVENT_RAISE_DP_FRIDAY'**.


 (Please refer the Appendix for sample code that helps raise the Event of a specific day – you will need to create the program in conjunction with the Variant for the program – described in the next section).

Save and activate the program.

Creating Variant for the Program

Using transaction SE38 , select the program created earlier. At the 'Subobjects' menu select the radio button for 'Variants' and click on 'Change'.
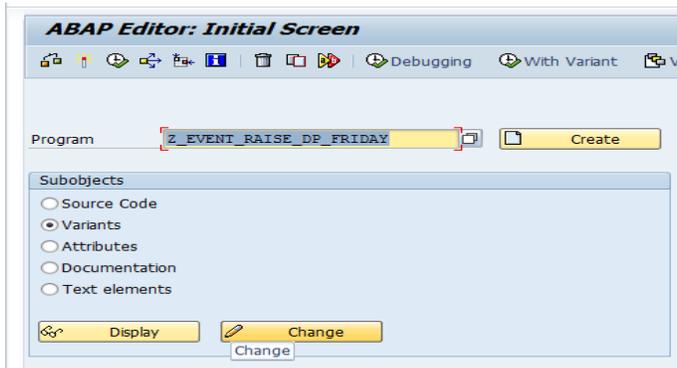
At the next prompt, the system will ask you to create the variant. We will create variant '**ZFRIDAY_RUN'**. Type in the variant name and select 'Create'.
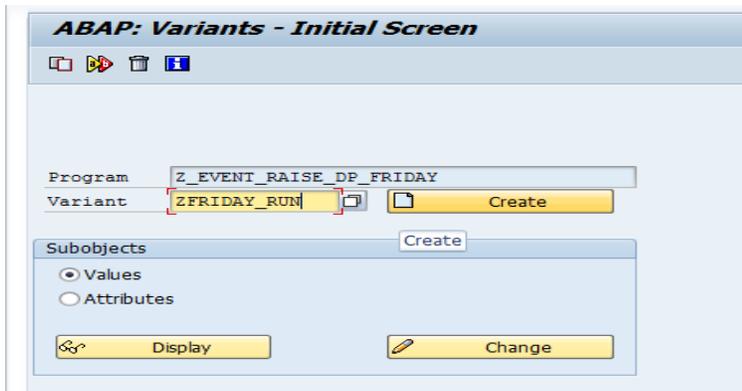
Enter the following values for each of the parameter

- **EVT_ID** : Event name that we created earlier. Selecting this event name we are indicating that whenever the ABAP program raises an event , the event would be **ZDP_FRIDAY_BROADCAST**.
- **TGT_SRV**: You need to check this entry as applicable for your BW server.
- **Z_DAYNUM**: This is a custom parameter included in the ABAP program. The value in this parameter determines the Day # of the week and it is on this date that the ABAP program will raise the event. As we want the dependent chain to run every Friday, the value of this parameter is entered as 5 (Friday is 5th day of the calendar week).
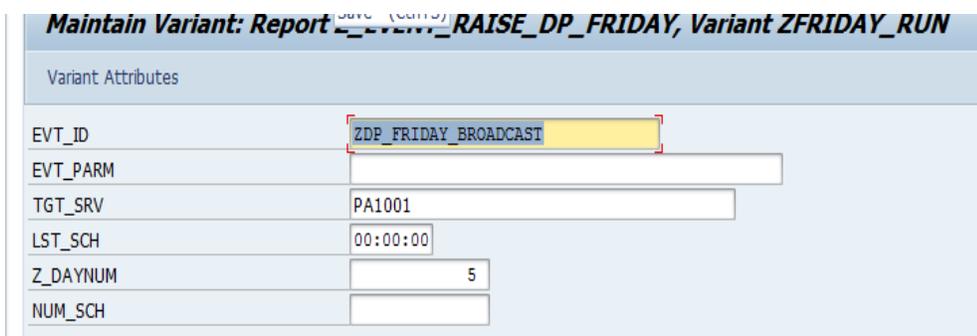
Save the parameter values and the variant.
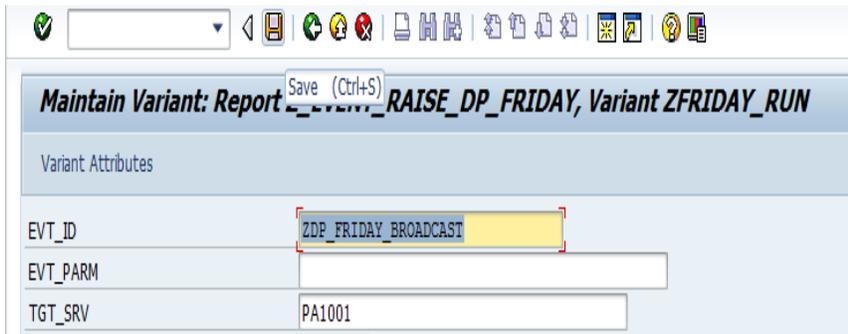
We have now created the event, ABAP program to raise the event and the variant for the ABAP program that carries the relevant parameters that would help raise the event.

Editing Parent Process Chain

We now make changes in the Parent process chain that will help run the process related to the ABAP program execution and raise the EVENT on the required day.

Using RSPC go to the Parent chain
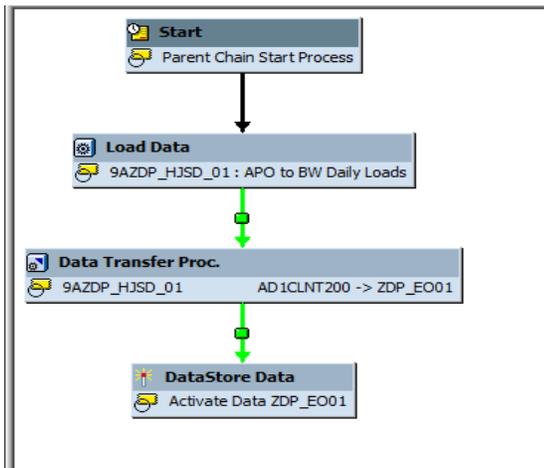
Now select the 'Process Types' from the menu and then select 'ABAP program'
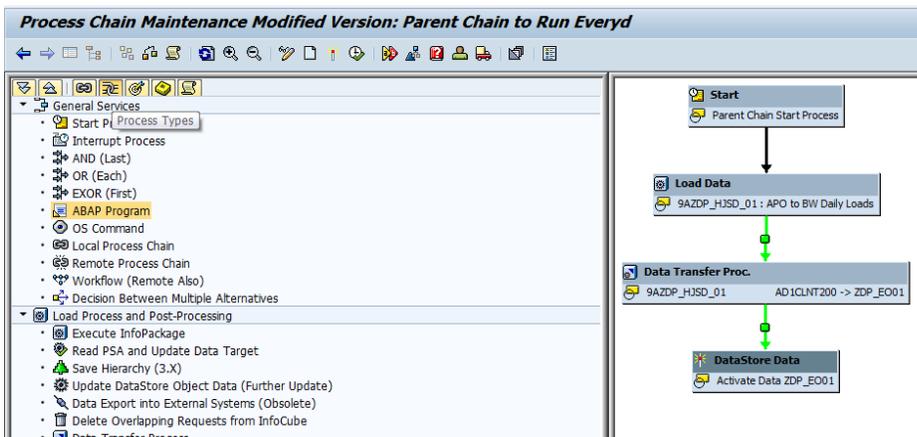
Create' a variant for the process
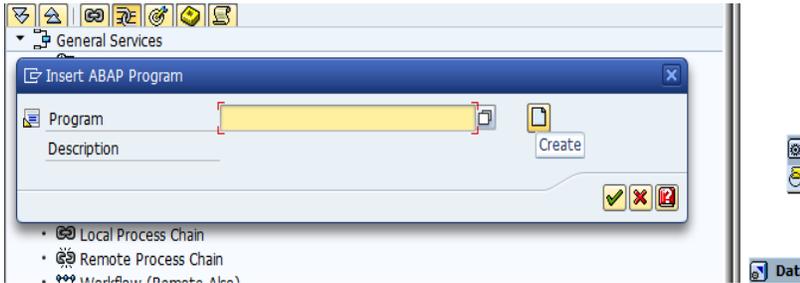
Enter the variant details and select 'Continue'.
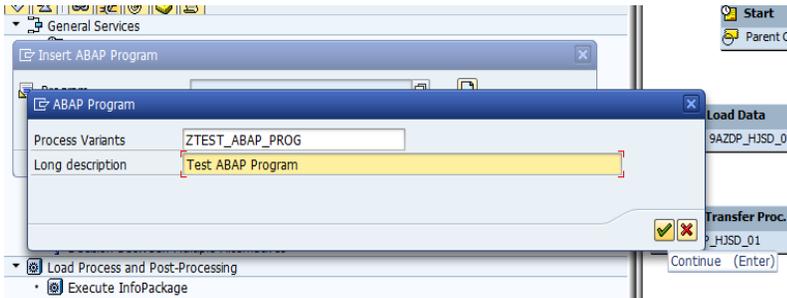
The process maintenance screen is displayed
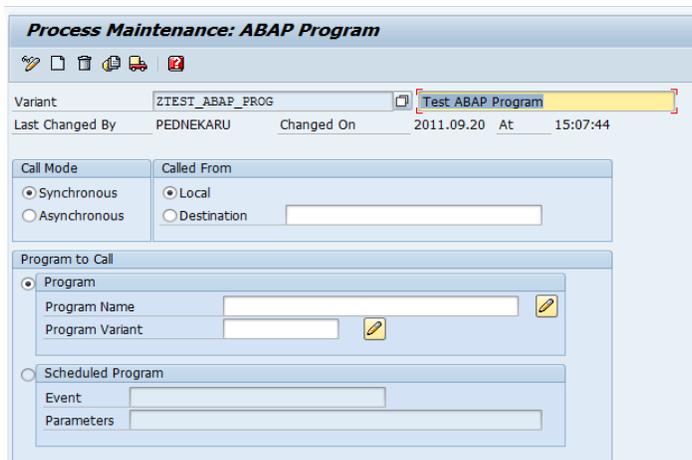
At the program name Tab enter the ABAP program name earlier created i.e. 'Z_EVENT_RAISE_DP_FRIDAY'
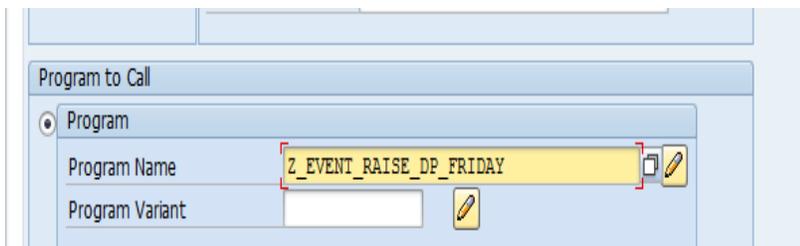
At the program variant tab, enter the variant name earlier created i.e. ZFRIDAY_RUN'
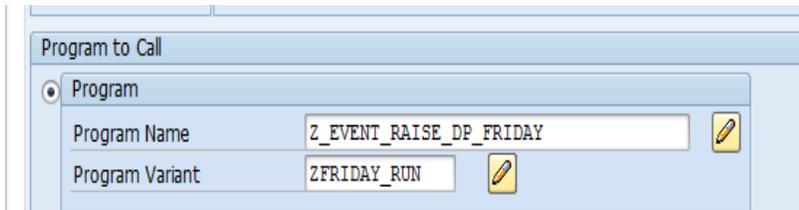
'Save' and Go back
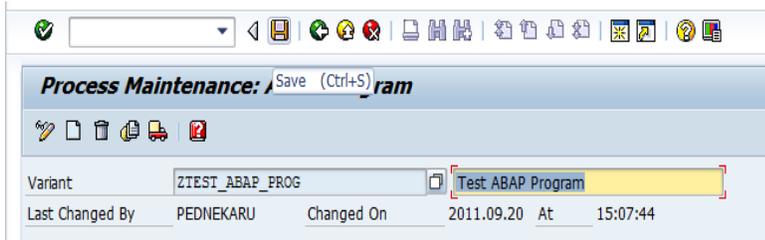
Select 'Continue' and you will see the new process included in the chain.
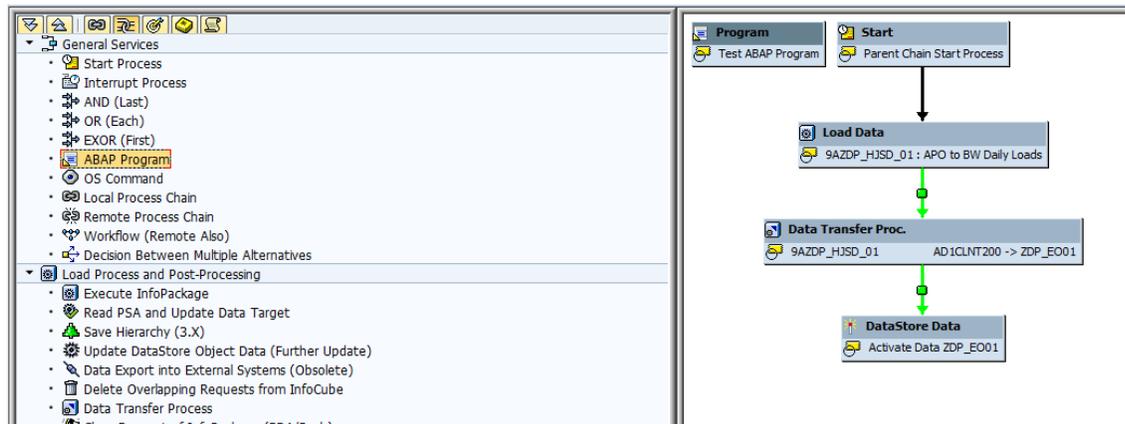
Add the process at the end of the chain , Save, Activate and reschedule the chain.
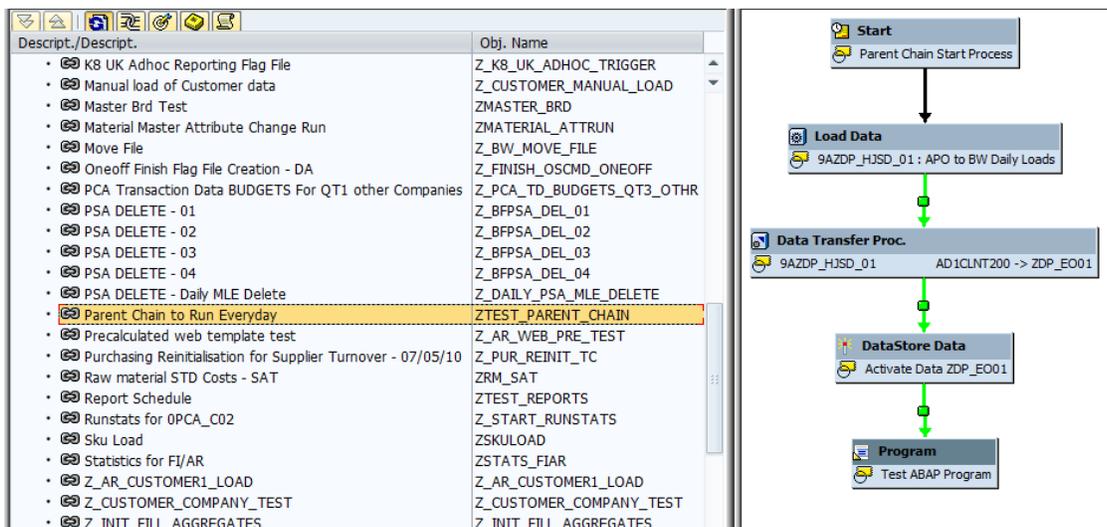
The new process added to the process does the following activities :

- Program Z_EVENT_RAISE_DP_FRIDAY is executed for the variant ZFRIDAY_RUN.
- The program finds the 'Day of the week' based on the system date.
- If the 'Day of the week' corresponds to the Day Number parameter in the Variant then the program raises an event called 'ZDP_FRIDAY_BROADCAST'.

Editing Dependent Process chain

The dependent process chain will now undergo a change. This chain has to get triggered following the event raised by its parent/preceding chain.

In the dependent process chain the scheduler settings are changed from
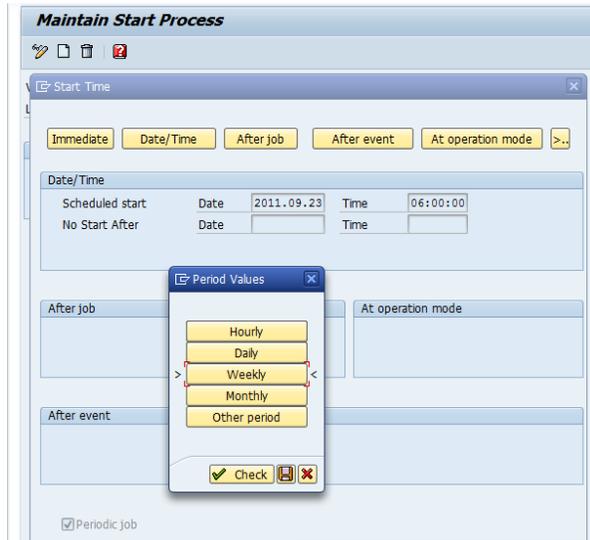
To

Click on 'After Event'
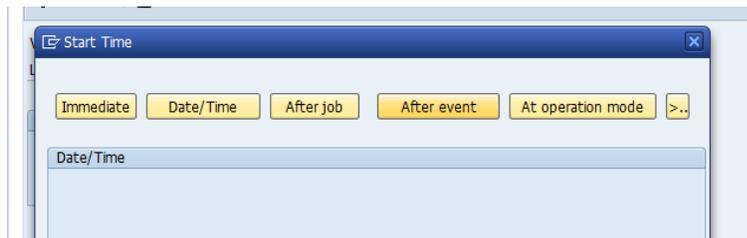
At the Event name tab enter the name of the event created in SM62 (this is the event included in the variant of the progam). Check the Periodic job tab.
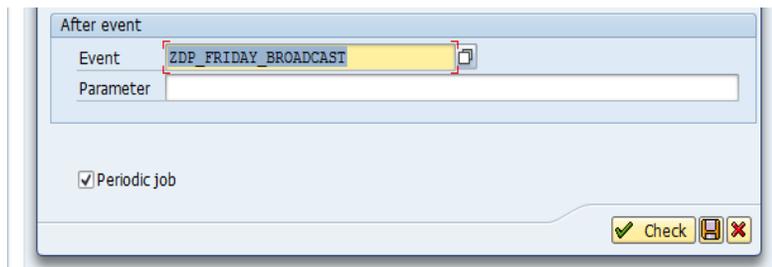
Check and Save the settings. Save the process , Go Back, Save, activate and reschedule the chain.

You can check the scheduled process for this chain. Right click on the 'Start process' and select 'Displaying Scheduled jobs'.
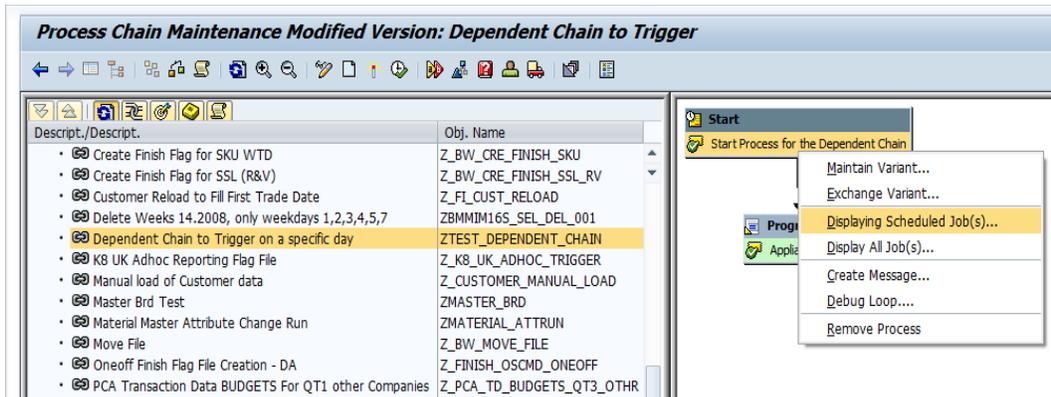
At the Job Overview screen double click on the entry.
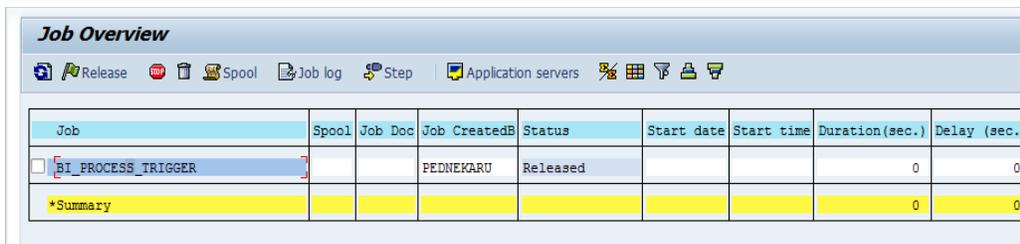
Process trigger for the chain shows the Trigger event.
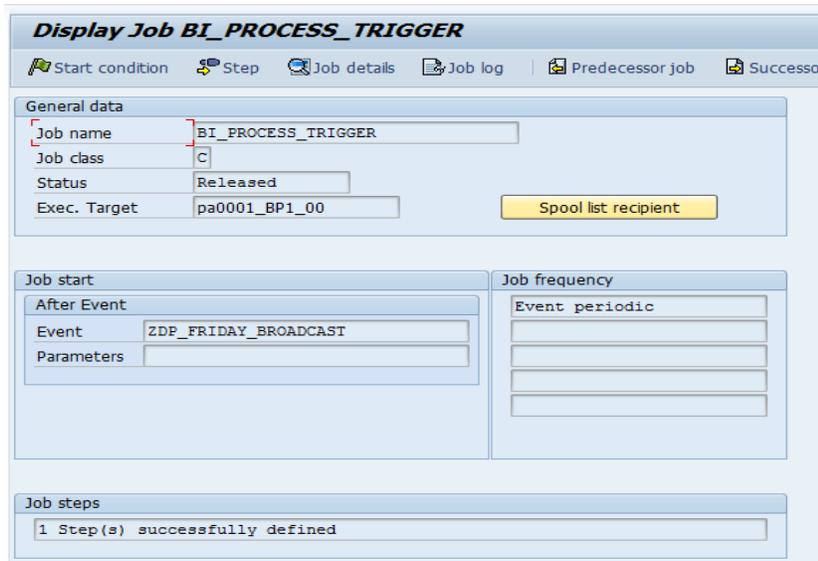
Appendix

```
*&---------------------------------------------------------------------*
*& Report  Z_EVENT_RAISE_DP_FRIDAY
*&
*&---------------------------------------------------------------------*
*&
*&
*&---------------------------------------------------------------------*
REPORT  Z_EVENT_RAISE_DP_FRIDAY.
PARAMETER : evt_id LIKE tbtco-eventid,      "Event ID
            evt_parm LIKE tbtco-eventparm,  "Parameter ID
            tgt_srv LIKE msxxlist-name,     "Target system
            lst_sch LIKE sy-uzeit,          "Last schedule time
            z_daynum type i,                "Day number
            num_sch TYPE i.                 "Number of schedules per hour
DATA : a,
       sys_min TYPE i.
DATA:TODAY LIKE SCAL-INDICATOR.
*  TO RUN ONLY ON THE DAY WHOSE DAY # IS IN THE VARIANT
 CALL FUNCTION 'DATE_COMPUTE_DAY'
    EXPORTING
      DATE          = SY-DATUM
    IMPORTING
      DAY           = TODAY.
*  IF DATE OF ABAP PROGRAM RUN IS EQUIVALENT TO THE VALUE
*  SET IN THE VARIANT PARAMETER THEN RAISE EVENT
  IF TODAY = z_daynum.
    PERFORM event_raise.
  ENDIF.
*&---------------------------------------------------------------------*
*&      Form  event_raise
*&---------------------------------------------------------------------*
*       text
*----------------------------------------------------------------------*
FORM event_raise.
  CALL FUNCTION 'BP_EVENT_RAISE'
    EXPORTING
      eventid               = evt_id
      eventparm             = evt_parm
      target_instance       = tgt_srv
    EXCEPTIONS
      bad_eventid           = 1
      eventid_does_not_exist = 2
      eventid_missing       = 3
      raise_failed          = 4
      OTHERS                = 5.
  CASE sy-subrc.
    WHEN 0.
      MESSAGE s250(bt) WITH evt_id.
    WHEN 1.
      MESSAGE e042(bt) WITH evt_id.
    WHEN 2.
      MESSAGE e042(bt) WITH evt_id.
    WHEN 3.
      MESSAGE e038(bt).
    WHEN 4.
      MESSAGE e249(bt) WITH evt_id.
  ENDCASE.
  WRITE: sy-uzeit, ': Event has been triggered'.
ENDFORM.                    "event_raise
```

Related Content

SCN Article : Trigger an Event Using Program and then Trigger a Process Chain by Eventn
http://www.sdn.sap.com/irj/scn/go/portal/prtroot/docs/library/uuid/4027834a-81e5-2c10-1692-866e137fe2a3?QuickLink=index&overridelayout=true

Blog : Scheduling process chains : http://www.sdn.sap.com/irj/scn/weblogs?blog=/pub/wlg/13787

SDN forum : http://forums.sdn.sap.com/thread.jspa?threadID=1991110

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.