

Triggering Process Chains ter Successful Completion of a Particular Job in R/3



Applies to:

SAP BW 3.5 and SAP BI 7.0. For more information, visit the [Business Intelligence homepage](#).

Summary

This Document discusses how to trigger the process chain after successful completion of a particular job in R/3.

Author: Mihir L. Kiri

Company: L & T Infotech

Created on: 21 May 2009

Author Bio



Mihir Kiri is a SAP Certified Application Associate- Business Intelligence with SAP NetWeaver 7.0. currently working with L & T Infotech India (Powai/Mumbai). He has over 2+ years of experience in various BW/BI Implementation/Support projects.

Table of Contents

Introduction	3
Requirement:.....	3
How to Achieve:	3
Step 1:	3
Step 2:	4
Step 3:	7
Step 4:	11
Step 5:	11
Related Content.....	17
Disclaimer and Liability Notice.....	18

Introduction

This article describes how to automatically trigger a Process Chain after successful completion of a job in R/3 system.

- Using this logic we can also automate execution of ABAP programs, Function modules and Info Packages after completion of a particular job in R/3 or BI.

Requirement:

We have a scenario where we have to trigger a process chain after successful completion of a job in R/3 system. For example:

We need to trigger CO-PA process chain after successful completion of its realignment job in R/3 so that the data from the profitability segment is read at the time of the extraction so that any realignments can be acted upon and the results in the Business Explorer match those in reports in CO-PA that have the same definition.

How to Achieve:

An ABAP program will be added as a step in the R/3 job. This ABAP program will run after successful completion of the original program on which R/3 job is based. This ABAP program will take RFC destination and event name as input parameters and also call a Z function module in this RFC destination and pass event name to this Z function module. This Z function module will be made remote enabled and will call standard function module 'BP_EVENT_RAISE' which will actually raise this event. This event should have been defined in SM64 in the RFC destination.

This event name will be used in scheduling of process chain via event and should also get defined as periodic job.

This has been mentioned in 5 steps explained below.

Step 1:

Go to SM64 and create new event

The screenshot displays the SAP 'Background Events: Overview and Administration' interface. The main window shows a table of background events. An 'Event Definition' dialog box is open, showing the event name 'ztesting' and description 'ztesting'.

Created By	Background Processing Event	Description	Changed by	Date	Time
SAP	SAP_ARCHIVING_DELETE_FINISHED	ADK: All archive files in session: Status "Deletion completed"	SAP	01.12.1999	22:00:53
SAP	SAP_ARCHIVING_WRITE_FINISHED	ADK: All archive files in session: Status "Archiving completed"	SAP	01.12.1999	22:01:17
SAP	SAP_BRANCHE_IMPORT	Industry sector import	SAP	09.02.1995	14:20:07
SAP	SAP_CSM_TRIGGER_CENSYS_DISPATCH		SAP	20.11.2000	14:59:09
SAP	SAP_DBA_ACTION		SAP	03.08.1995	14:10:53
SAP	SAP_END_OF_JOB		SAP	07.09.1993	13:29:38
SAP	SAP_EVTHIST_CRITERIA_ACTIVATED		SAP	18.10.2005	09:13:53
SAP	SAP_IMPORT_START	Change and Transport System: Start of an import	SAP	26.10.1998	08:46:30
SAP	SAP_IMPORT_STOP	Change and Transport System: End of an import	SAP	26.10.1998	08:46:43
SAP	SAP_LANGUAGE_FILL	Language filling	SAP	16.01.1995	16:58:07
SAP	SAP_LANGUAGE_IMPORT	Language import	SAP	16.01.1995	16:57:57
SAP	SAP_MONITORING_STARTUP_DISPATCH		SAP	01.07.2003	10:01:33
SAP	SAP_OPMODE_SWITCH		SAP	26.01.1994	17:36:36
SAP	SAP_POSDW_TASK_PROCESS		JEBENS	03.05.2004	15:42:15
SAP	SAP_QEVENT	Event at the start of a background job via QAPI	SAP	27.06.1994	11:45:18
SAP	SAP_RSCSMNCDATA		SAP	05.08.1999	10:24:11
SAP	SAP_SYSTEM_START		SAP	07.09.1993	13:31:34
			SAP	07.09.1993	13:30:13
			SAP	07.07.1993	10:59:37
			ment SAP	02.11.1998	12:51:11
			SAP	20.07.1994	21:46:51
			SAP	16.05.1995	21:09:29
			DEVELOPER	06.02.2009	10:21:27
			DEVELOPER	06.05.2009	12:34:13

The 'Event Definition' dialog box shows:

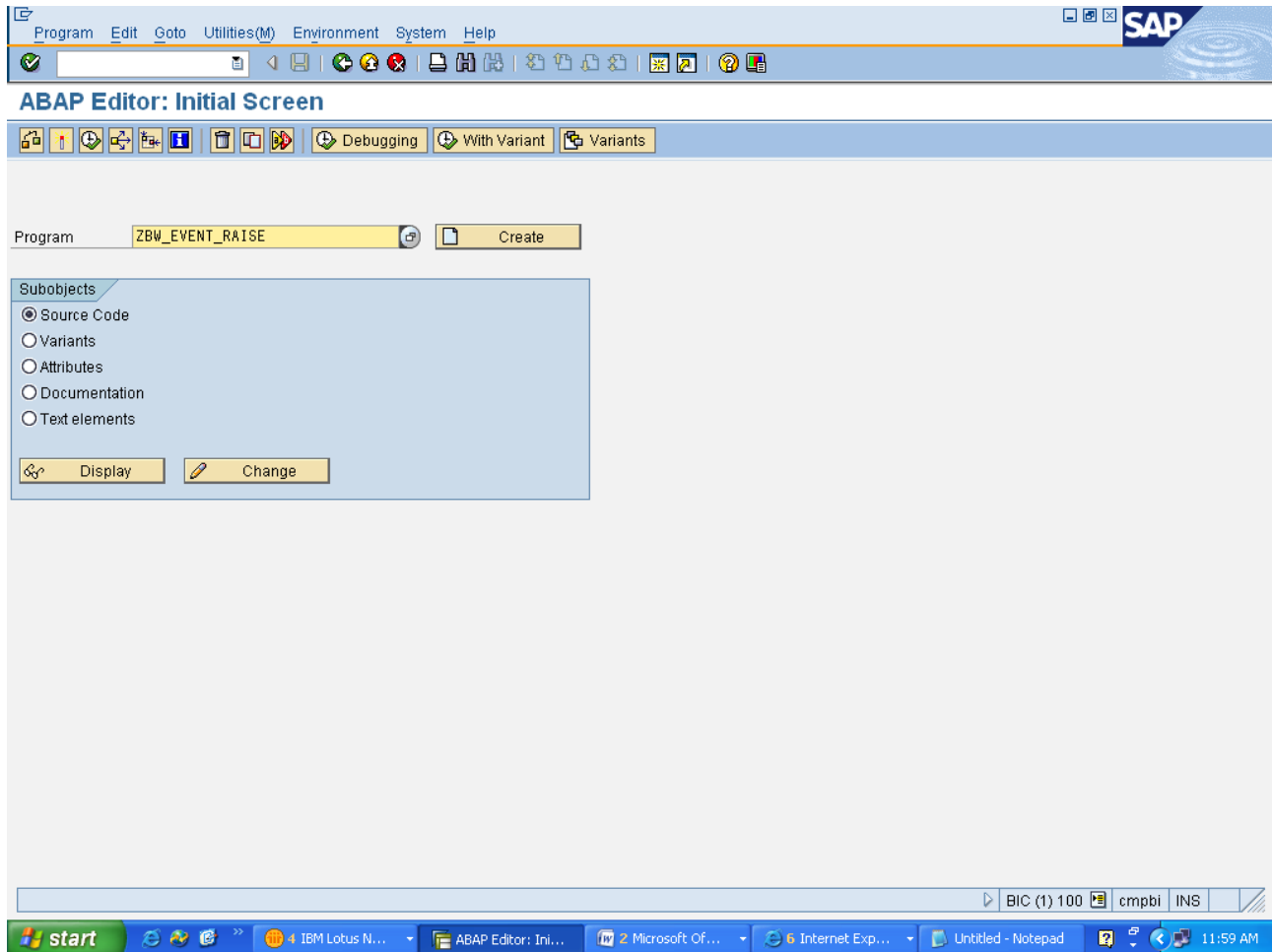
- Event: ztesting
- Description: ztesting
- System
- Buttons: Save, X

We will use this 'ZTESTING' event in process chain.

Step 2:

Create a program in R/3 system which will take RFC destination where process chain will run and event name as input and call a RFC enabled function module in this RFC destination and pass this event name as parameter.

Go to SE38

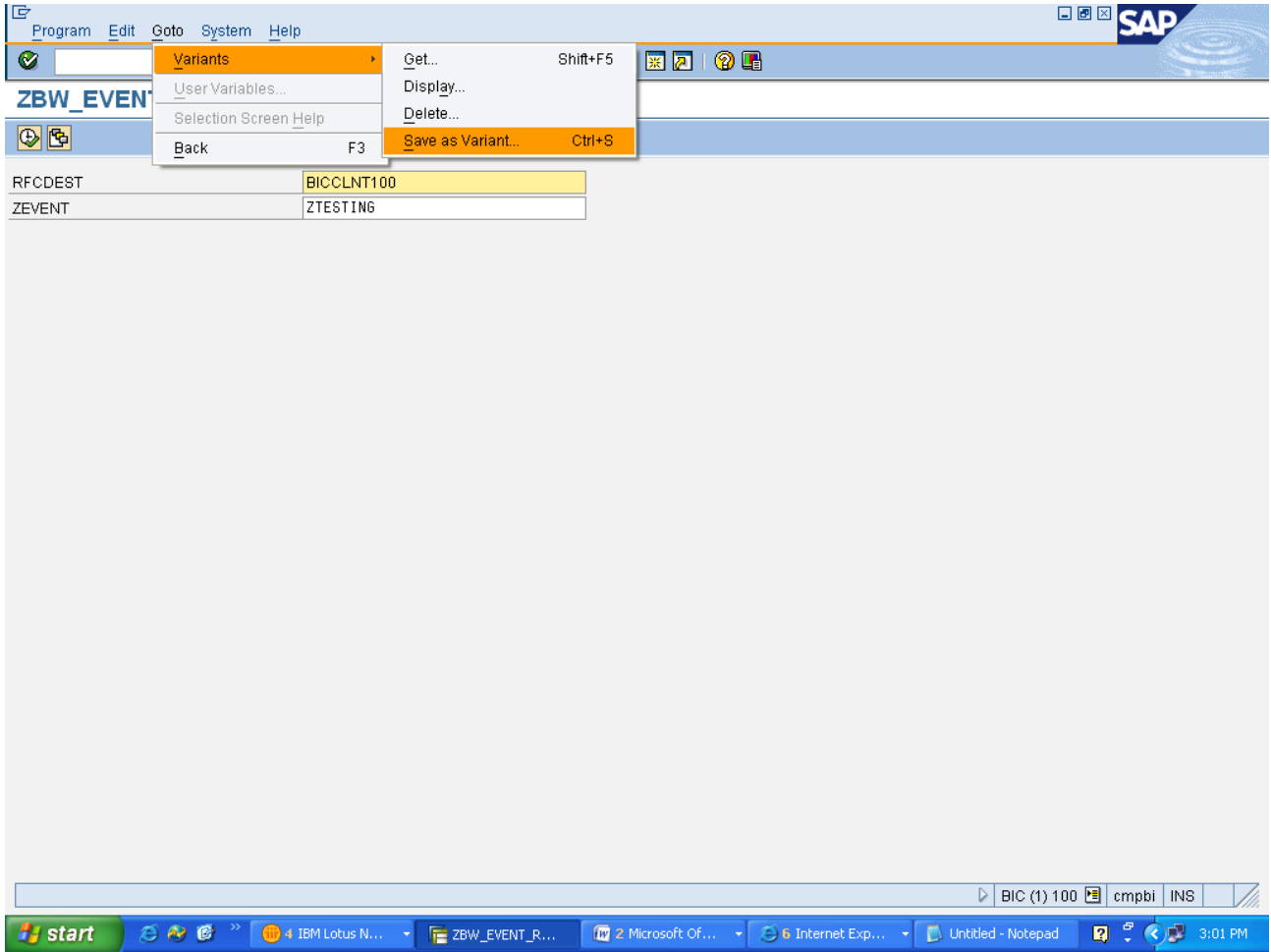


Source code as below

```
REPORT ZBW_EVENT_RAISE.
parameters: rfcdst like rfcdisplay-rfcdst,
zevent type char32.
```

```
call function 'Z_BW_EVENT_RAISE'
destination rfcdst
exporting
eventid = zevent.
```

Execute the ABAP program and enter the RFC destination of the process chain and the event name and save as variant.



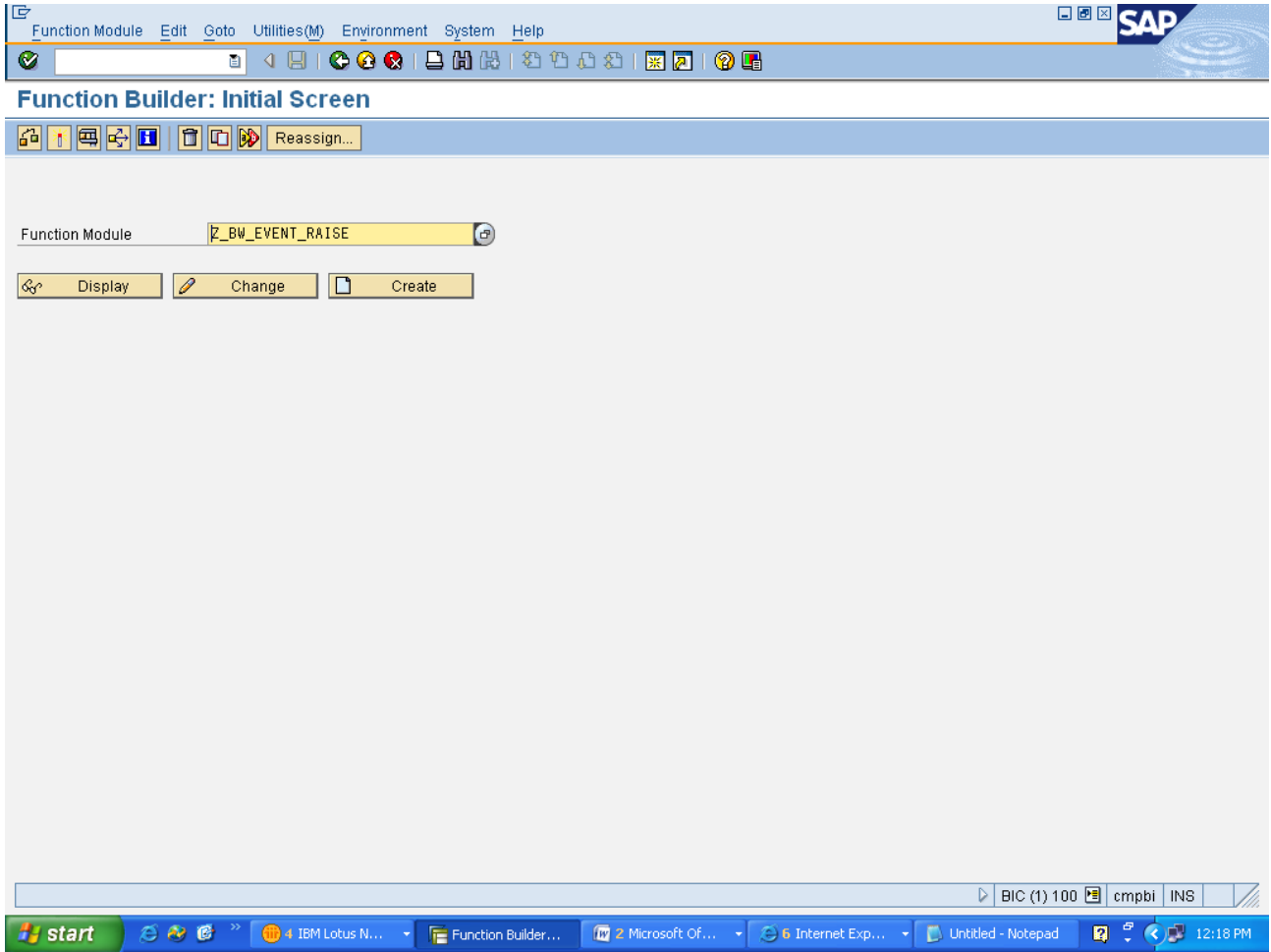
The screenshot displays the SAP 'Variant Attributes' configuration screen for variant 'VAR_MP'. The variant name is 'VAR_MP' and its meaning is 'Var_MP'. There are several checkboxes for variant properties: 'Only for Background Processing', 'Protect Variant', 'Only Display in Catalog', and 'System Variant (Automatic Transport)'. A 'Scrn Assignm.' table shows a screen assignment for '1000'. Below this is a table of 'Objects for selection screen' with columns for Selection Scr., Field na., Ty., Protect field, Hide field, Hide field 'BIS', Save field without values, Switch GPA off, Required field, Selection variab., Option, and Name of Variable (Input C).

Selection Scr.	Field na.	Ty.	Protect field	Hide field	Hide field 'BIS'	Save field without values	Switch GPA off	Required field	Selection variab.	Option	Name of Variable (Input C)
1.000	RFCDEST	P	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
1.000	ZEVENT	P	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			

Step 3:

Create a RFC enable function module. This will take event ID as input from the SE38 program in step 2.

Go to SE37.



Enter the definitions as below

The screenshot shows the SAP Function Builder interface for the function module Z_BW_EVENT_RAISE. The 'Processing Type' section is highlighted, with 'Remote-Enabled Module' selected and circled. The 'General Data' section contains the following information:

General Data	
Person Responsible	DEVELOPER
Last Changed By	DEVELOPER
Changed on	07.05.2009
Package	\$TMP
Program Name	SAPLZ_GROUP
INCLUDE Name	LZ_GROUPU01
Original Language	EN
Not released	
<input type="checkbox"/> Edit Lock	
<input type="checkbox"/> Global	

Insert code as below

```

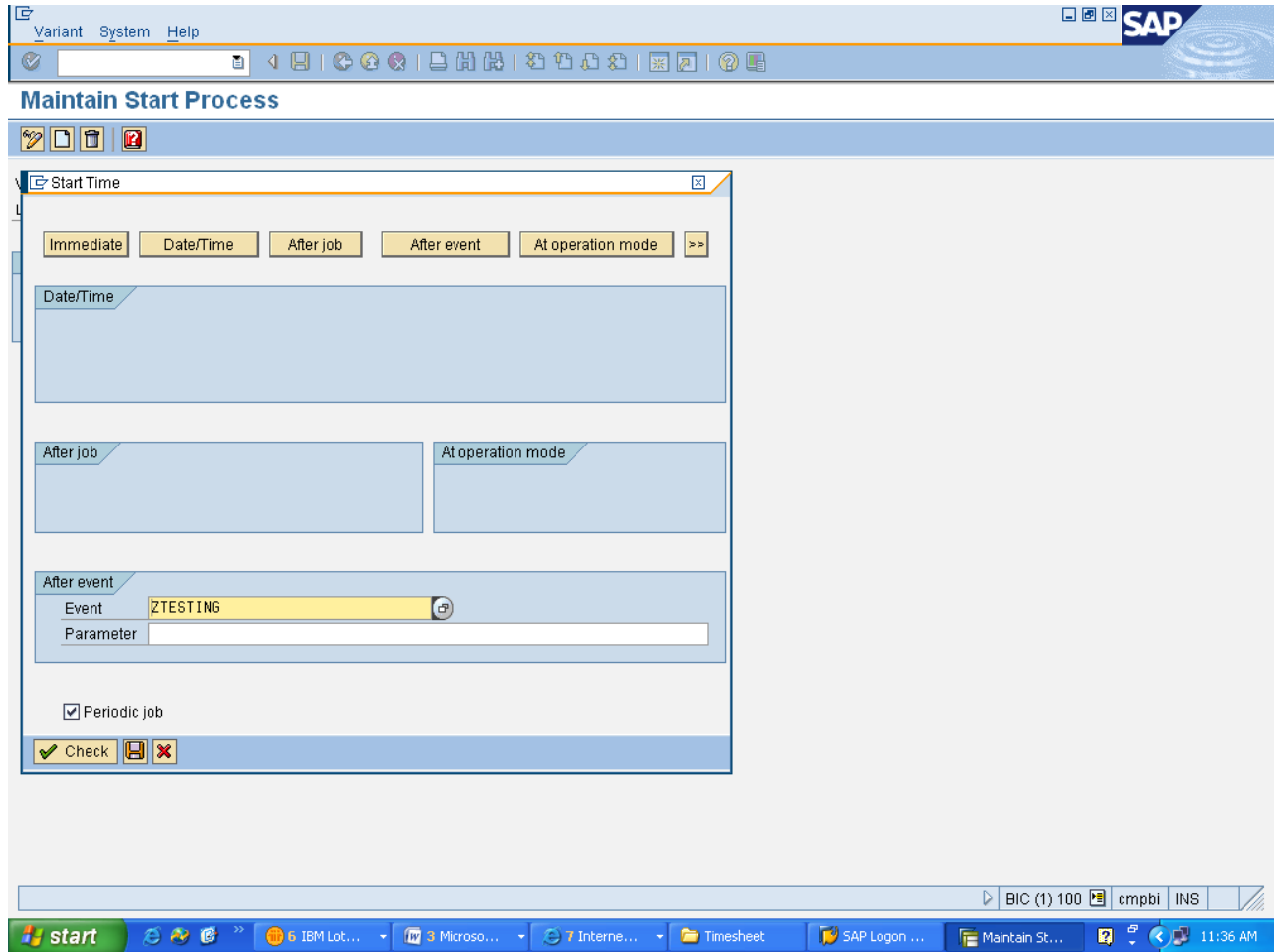
FUNCTION z_bw_event_raise.
*-----
* " * "Local Interface:
* "   IMPORTING
* "     VALUE(EVENTID) TYPE  TBTCO-EVENTID
* "   EXPORTING
* "     VALUE(RESULT) TYPE  CHAR2
*-----
CALL FUNCTION 'BP_EVENT_RAISE'
EXPORTING
  eventid          = eventid
EXCEPTIONS
  bad_eventid      = 1
  eventid_does_not_exist = 2
  eventid_missing  = 3
  raise_failed     = 4
  OTHERS           = 5.

IF sy-subrc EQ 0.
  result = 'OK'.
ENDIF.
ENDFUNCTION.

```

Step 4:

Insert the defined event id in start variant of the process chain and make it a periodic job.

**Step 5:**

Add the SE38 program as a step in the R/3 job for which we need to trigger process chain.

For example BBP_GET_STATUS_2 is the job whose completion should trigger process chain.

Got SM37

SAP

Job Edit Goto Extras Settings System Help

Job Overview

Release Spool Job log Step Application servers

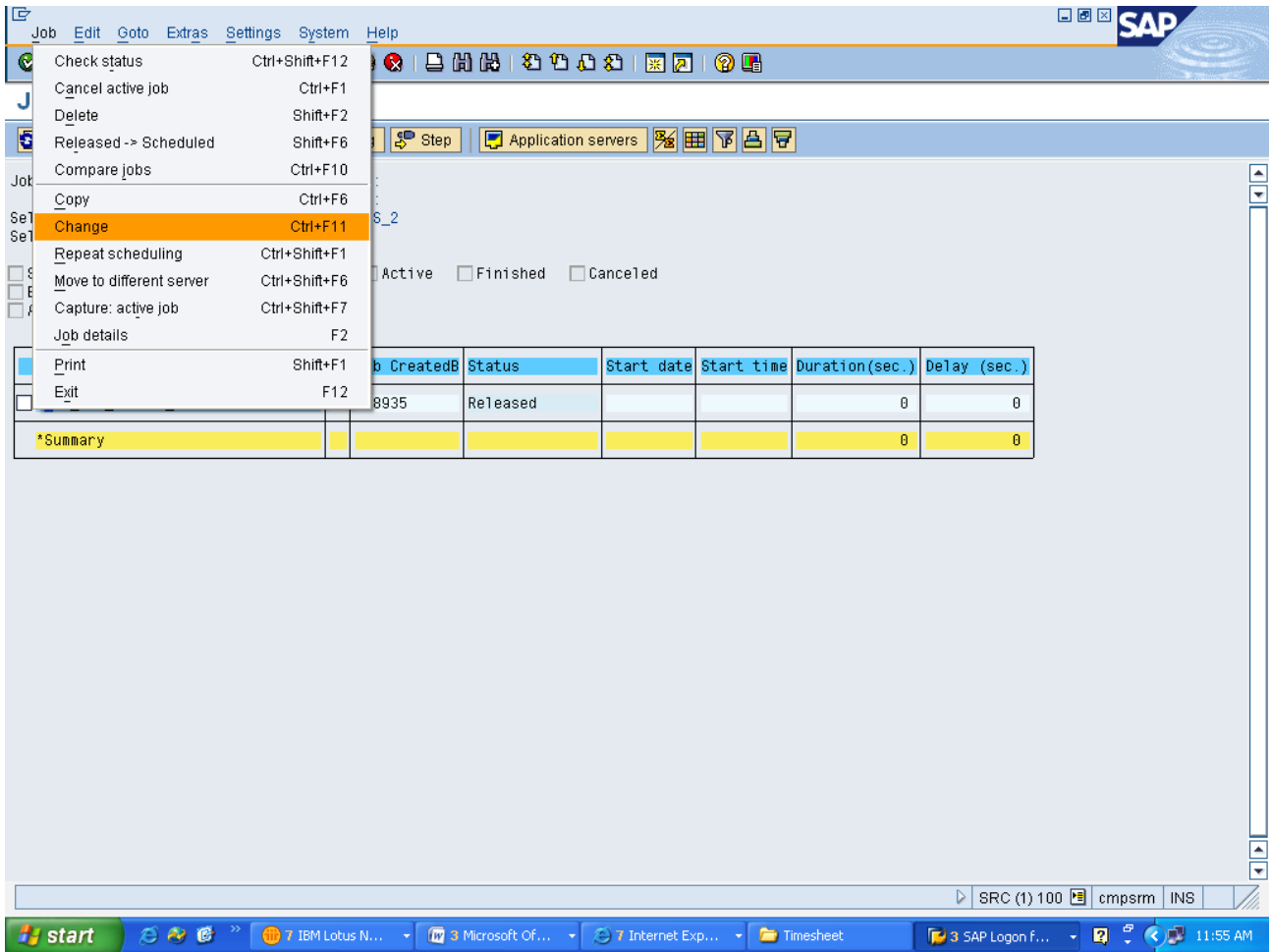
Job overview from: 26.05.2009 at: : :
 to: 26.05.2009 at: : :
 Selected job names: BBP_GET_STATUS_2
 Selected user names: *

Scheduled Released Ready Active Finished Canceled
 Event controlled Event ID:
 ABAP program Program name :

Job	Ln	Job CreatedB	Status	Start date	Start time	Duration(sec.)	Delay (sec.)
BBP_GET_STATUS_2		718935	Released			0	0
*Summary						0	0

SRC (1) 100 cmpsrm INS

start IBM Lotus N... Microsoft Of... Internet Exp... Timesheet SAP Logon F... 11:53 AM



Click "Step"

Job Edit Goto System Help

SAP

Change Job BBP_GET_STATUS_2

Start condition Step Job details Predecessor job Successor job Job selection

General data

Job name: BBP_GET_STATUS_2
Job class: [icon]
Status: Released
Exec. Target: [empty] Spool list recipient

Job start

Planned Start
Date: 26.05.2009 Time: 12:05:00

Job frequency

2 Minute(s)

Job steps

1 Step(s) successfully defined

SRC (1) 100 cmpsrm INS 11:55 AM

Click "Create"

The screenshot shows the SAP Step List Overview window. The title bar includes 'Step List Overview' and the SAP logo. Below the title bar is a toolbar with various icons. The main area contains a table with the following data:

No.	Program name/command	Prog. type	Spool list	Parameters	User	Lang.
1	BBP_GET_STATUS_2	ABAP		CLNT200_300	718935	EN

The bottom status bar shows 'SRC (1) 100 | cmpsrm | INS' and the Windows taskbar at the bottom displays the start button, several open applications, and the time '11:57 AM'.

Insert the ABAP program and its associated variant which was created in Step 2.

The screenshot shows the SAP 'Step List Overview' dialog box for 'Create Step 1'. The 'User' field is set to 'DEVELOPER'. Under 'Program values', three options are available: 'ABAP program', 'External command', and 'External program'. The 'ABAP program' option is selected, with 'Name' set to 'ZBW_EVENT_RAISE', 'Variant' set to 'Var_MP', and 'Language' set to 'EN'. The 'External command' section includes fields for 'Name', 'Parameters', 'Operating sys.', and 'Target server'. The 'External program' section includes fields for 'Name', 'Parameter', and 'Target host'. At the bottom, there are buttons for 'Check', 'Print specifications', and a close button. The taskbar at the bottom shows the Windows start button, several open applications, and the system clock at 11:59 AM.

Related Content

<https://www.sdn.sap.com/irj/scn/thread?messageID=6986590#6986590>

<https://www.sdn.sap.com/irj/scn/thread?threadID=258434>

<https://www.sdn.sap.com/irj/scn/thread?messageID=2740707#2740707>

For more information, visit the [Business Intelligence 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.