

SAP BUSINESS TECHNOLOGY PLATFORM | EXTERNAL

Serialization of IDocs Configuration Guide

SAP Responsible Design and Production Integration with SAP ERP

Table of Contents

OVERVIEW.....	3
DISABLE IMMEDIATE PROCESSING OF IDOCS FOR PARTNER.....	4
CREATE A NEW VARIANT FOR PROGRAM RBDAPP01.....	6
CHOOSE A PROGRAM TO SCHEDULE	9
OPTION A: CREATE AND SCHEDULE A NEW PROGRAM TO WRAP RBDAPP01 - ZRDP_RUN_AND_RESCHEDULE (RECOMMENDED)	10
OPTION B: SCHEDULE THE IDOC PROCESSING PROGRAM – RBDAPP01.....	19

Overview

Please refer to the known issue described in the following SAP Note: <https://me.sap.com/notes/3192109>

While all the solutions in the SAP Note are valid, only the ones described in this document are verified to be completely compatible with the Cloud Integration Package “**SAP Responsible Design and Production Integration with SAP ERP**”.

Disable immediate processing of IDocs for partner

Execute Transaction: **WE20**

Find the partner you have configured in Cloud Integration.

Example Values:

- Type: LS (EDI_DC40/SNDPRT)
- Partner Number: RDPCPI (EDI_DC40/SNDPRN)

Partner No.

RDPCPI

RDPCPI

Type

LS

Logical system

Post Processing: Valid Processors

Classification

Telephony

Ty.

US

User

Processor

ZTEST_USER

ZTEST_USER

Lang.

EN

English

Outbound

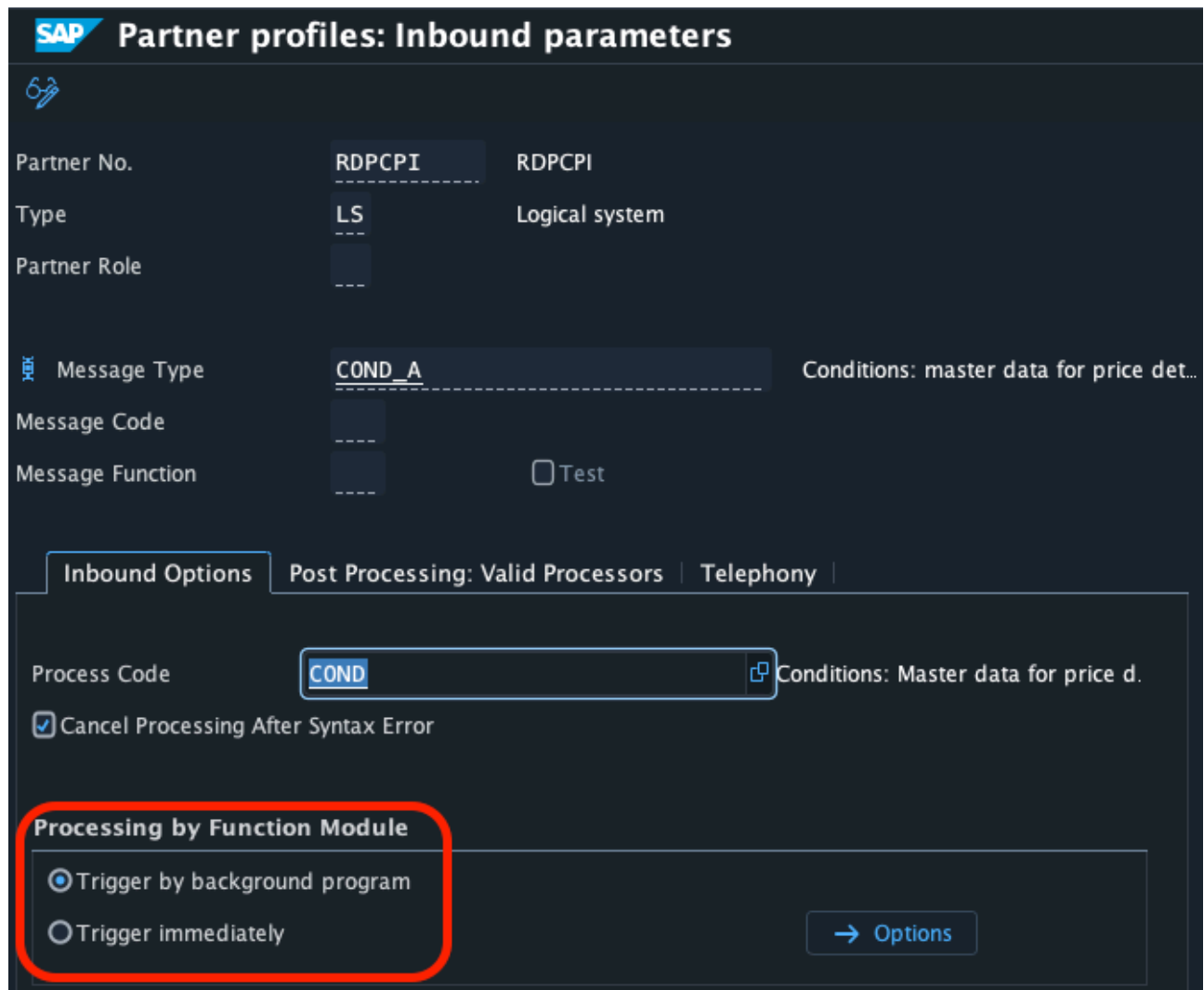
	Partner Role	Message Type	Message Vari.	Function	Test	Receiver Port	Id.	Pac...
					☐			
					☐			
					☐			
					☐			
					☐			
					☐			
					☐			
					☐			
					☐			
					☐			

Inbound

	Partner Role	Message Type	Message Vari.	Function	Test	P.	Process Code
		CLFMAS			☐	CLFM	
		COND_A			☐	COND	
		DEBMAS			☐	DEBM	

In the inbound section, double click on the Message type you want to replicate.

Make sure **“Trigger by background program”** is selected.



The screenshot shows the SAP S/4HANA 'Partner profiles: Inbound parameters' configuration screen. The 'Inbound Options' tab is active. The 'Process Code' field is set to 'COND' and is highlighted with a red box. Below it, the 'Processing by Function Module' section is also highlighted with a red box, showing two radio button options: 'Trigger by background program' (which is selected) and 'Trigger immediately'. Other fields include 'Partner No.' (RDPCPI), 'Type' (LS), 'Message Type' (COND_A), and 'Message Code' (---). A 'Test' checkbox is visible next to the 'Message Function' field.

SAP Partner profiles: Inbound parameters

Partner No. RDPCPI RDPCPI

Type LS Logical system

Partner Role ---

Message Type COND_A Conditions: master data for price det...

Message Code ---

Message Function --- ☐ Test

Inbound Options | Post Processing: Valid Processors | Telephony

Process Code COND Conditions: Master data for price d.

☒ Cancel Processing After Syntax Error

Processing by Function Module

☒ Trigger by background program

☐ Trigger immediately

→ Options

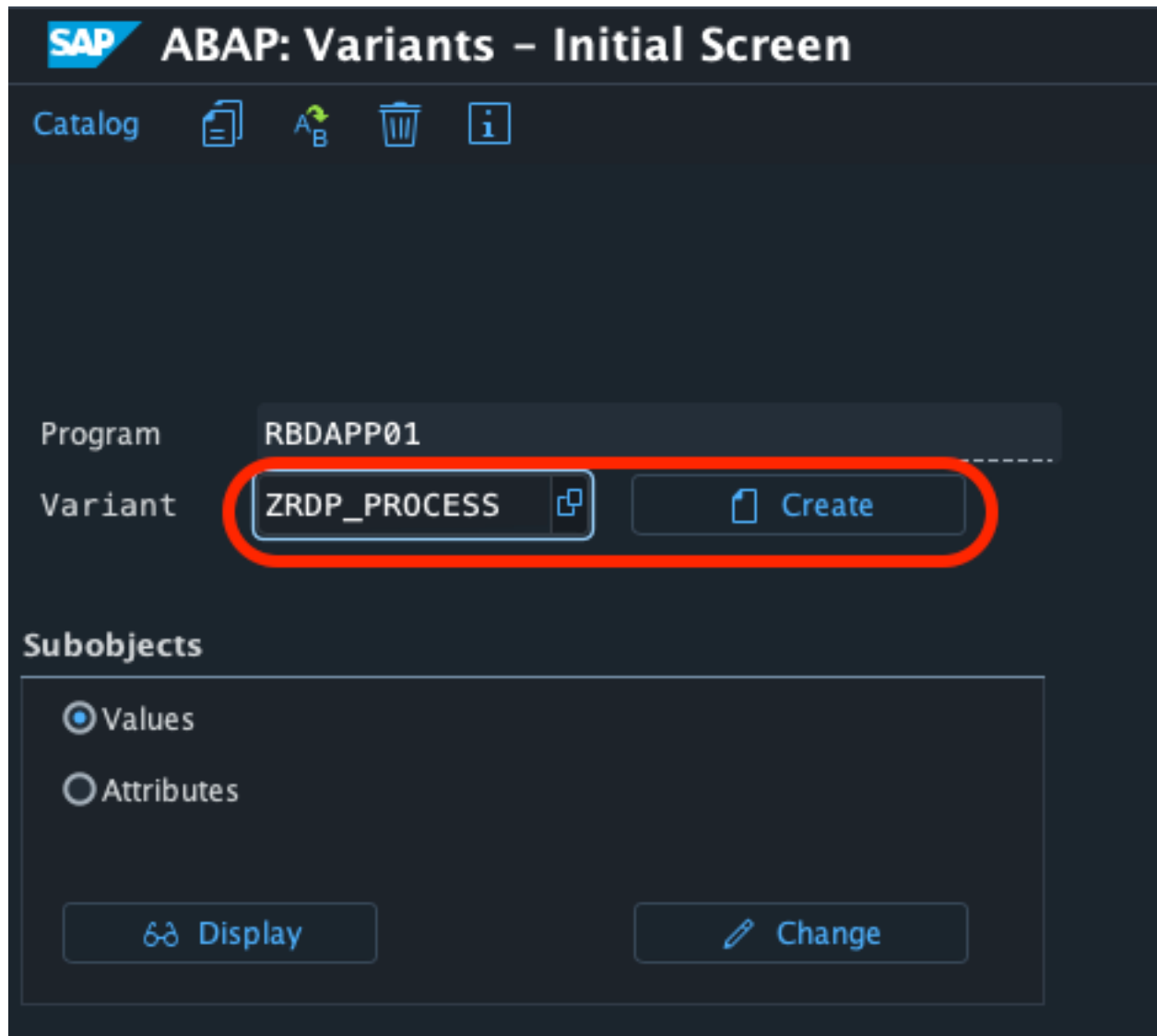
Save and go back.

If you want to replicate other message types as well, repeat this step. All relevant IDocs must be set to **“Trigger by background program”**.

Create a new variant for program RBDAPP01

Execute transaction: **RBDAPP01**

Enter a Variant name of choice then click create.



The screenshot shows the 'SAP ABAP: Variants - Initial Screen'. At the top, there is a 'Catalog' link and several icons. Below this, the 'Program' field is set to 'RBDAPP01'. The 'Variant' field contains 'ZRDP_PROCESS' and is highlighted with a red circle. To the right of the variant field is a 'Create' button, also within the red circle. Below the variant field, there is a 'Subobjects' section with two radio buttons: 'Values' (selected) and 'Attributes'. At the bottom of the 'Subobjects' section are two buttons: 'Display' and 'Change'.

SAP ABAP: Variants – Initial Screen

Catalog [Icons]

Program: RBDAPP01

Variant: ZRDP_PROCESS [Copy Icon] [Create Button]

Subobjects

☒ Values
☐ Attributes

[Display Button] [Change Button]

Make sure to select screens number **1000** and **1100**.

Variants: Change Screen Assignment

☐ For all selection screens

☒ For individual screens

Created for Selection Screens

Created	Number	Ty.
☒	1000	Full screen
☒	1100	Subscreen
☐	1200	Subscreen

☒ Continue ☐

Then click Continue.

Set the following:

- Created On: <CURRENT_DATE> to 31.12.9999
- Sender Partner Type: LS (If you want to affect only this Partner)
- Sender Partner No.: RDPCPI (If you want to affect only this Partner)
- IDoc Status: 64
- Packet Size: 200

SAP Edit Variants: Report RBDAPP01, Variant ZRDP_PROCESS, Screen 1 of 2

← → ⓘ

IDoc selection Parallel Processing

IDoc Number		to		↗
Created On	26.05.2026	to	31.12.9999	↗
Created At	00:00:00	to	23:59:59	↗
IDoc Status		to		↗
Message type		to		↗
Message Variant		to		↗
Message Function		to		↗
Sender Partner Type	LS	to		↗
Sender Partner No.	RDPCPI	to		↗
Partner Function		to		↗
Pack. Size	200			
Test flag		to		

if object channel serializer active

Object Type		to		↗
-------------	--	----	--	---

Output Control

☒ Output list

Then save the variant.

Choose a Program to schedule

For serialization of very high quantities of IDocs, it is recommended to create a small wrapper program to synchronize the processing of RBDAPP01.

The alternative is to schedule RBDAPP01 directly. While this approach is simpler than the wrapper program, it has a significant downside. The time to process all IDocs in a load (initial or delta) should not exceed the specified period, otherwise parallel instances of RBDAPP01 would start running and serialization errors will occur.

For example, if RBDAPP01 is scheduled to execute every 5 minutes, all IDocs in a specific load must be able to process in those 5 minutes to avoid errors. The only workaround is increasing the period between executions. The wrapper ZRDP_RUN_AND_RESCHEDULE does not have this downside.

Option A: Create and Schedule a new program to wrap RBDAPP01 - ZRDP_RUN_AND_RESCHEDULE (Recommended)

Execute transaction: **SE38**

For Program enter: **ZRDP_RUN_AND_RESCHEDULE**

SAP ABAP Editor: Initial Screen

Program [Create](#)

Subobjects

- ☒ Source code
- ☐ Variants
- ☐ Attributes
- ☐ Text elements
- ☐ Documentation

[Display](#) [Change](#)

Then click Create.

Configure:

- Title: Periodic IDoc Inbound Processing (RBDAPP01)
- ABAP Language Version: Standard ABAP

Title	Periodic IDoc Inbound Processing (RBDAPP01)	
Original language	EN	
Created		26.05.2026
Last Changed		
Status	new(revised)	
Attributes		
Type	Executable program	
Status	Unclassified	
Authorization Group		
Application		
LDB Name		
Selection Screen		
ABAP Language Version	Standard ABAP	
☒ Fixed point arithmetic		
☐ Editor lock		
☐ Start using variant		

 Save   

Then click Save.

Choose the Package to use on the next screen then proceed.

An ABAP Editor should appear.

Replace the contents of the Program with the following code:

```

REPORT zrdp_run_and_reschedule.

CONSTANTS:
  c_jobname      TYPE tbtctjob-jobname VALUE 'ZRDp_RUN_AND_RESCHEDULE';
  c_variant      TYPE raldb_vari   VALUE '<CONFIGURE_YOUR_VARIANT_HERE>';
  c_min_period_secs TYPE i          VALUE 60.

DATA:
  lv_curr_jobname TYPE tbtctjob-jobname,
  lv_curr_jobcount TYPE tbtctjob-jobcount,
  lv_other_jobcount TYPE tbtctjob-jobcount.

IF sy-batch IS INITIAL.
  MESSAGE 'Must run as background job' TYPE 'A'.
ENDIF.

SELECT COUNT(*) FROM varid WHERE report = 'RBDAPP01' AND variant = c_variant.
IF sy-subrc <> 0.
  MESSAGE |Variant { c_variant } missing| TYPE 'A'.
ENDIF.

PERFORM schedule_next.

IF lv_curr_jobcount IS NOT INITIAL.
  SELECT SINGLE jobcount FROM tbtco INTO lv_other_jobcount
    WHERE jobname = c_jobname AND jobcount <> lv_curr_jobcount AND status = 'R'.
  IF sy-subrc = 0.
    RETURN.
  ENDIF.
ENDIF.

SUBMIT rbdapp01 USING SELECTION-SET c_variant AND RETURN.

FORM schedule_next.
  DATA: lv_jobcount TYPE tbtctjob-jobcount,
        lv_ts      TYPE timestamp,
        lv_sys_tz   TYPE ttzz-tzone,
        lv_start_dt TYPE sy-datum,
        lv_start_tm TYPE sy-zeit.

  CALL FUNCTION 'GET_JOB_RUNTIME_INFO'
    IMPORTING jobname = lv_curr_jobname jobcount = lv_curr_jobcount
    EXCEPTIONS OTHERS = 1.
  IF sy-subrc <> 0 OR lv_curr_jobname IS INITIAL OR lv_curr_jobcount IS INITIAL.
    WRITE: / 'GET_JOB_RUNTIME_INFO failed, sy-subrc=', sy-subrc,
           'jobname=', lv_curr_jobname, 'jobcount=', lv_curr_jobcount.
    RETURN.
  ENDIF.

  CALL FUNCTION 'GET_SYSTEM_TIMEZONE'
    IMPORTING timezone = lv_sys_tz
    EXCEPTIONS OTHERS = 1.
  IF sy-subrc <> 0 OR lv_sys_tz IS INITIAL.
    WRITE: / 'GET_SYSTEM_TIMEZONE failed, falling back to UTC, sy-subrc=', sy-subrc.
    lv_sys_tz = 'UTC'.
  ENDIF.

  TRY.
    GET TIME STAMP FIELD lv_ts.
    lv_ts = cl_abap_tstmp=>add( tstmp = lv_ts secs = c_min_period_secs ).
    CONVERT TIME STAMP lv_ts TIME ZONE lv_sys_tz INTO DATE lv_start_dt TIME lv_start_tm.
  CATCH cx_root INTO DATA(lo_ts_err).
  WRITE: / 'Timestamp/timezone conversion failed, tz=', lv_sys_tz,
        'msg=', lo_ts_err->get_text( ).
  RETURN.
ENDTRY.

CALL FUNCTION 'JOB_OPEN'
  EXPORTING jobname = c_jobname
  IMPORTING jobcount = lv_jobcount
  EXCEPTIONS OTHERS = 1.
IF sy-subrc <> 0.
  WRITE: / 'JOB_OPEN failed, sy-subrc=', sy-subrc.
  RETURN.
ENDIF.

SUBMIT zrdp_run_and_reschedule VIA JOB c_jobname NUMBER lv_jobcount AND RETURN.

CALL FUNCTION 'JOB_CLOSE'
  EXPORTING
    jobname      = c_jobname
    jobcount     = lv_jobcount
    sdlstrtdt    = lv_start_dt
    sdlstrttm    = lv_start_tm
    pred_jobname = lv_curr_jobname
    pred_jobcount = lv_curr_jobcount
  EXCEPTIONS OTHERS = 1.
IF sy-subrc = 0.
  COMMIT WORK AND WAIT.
ELSE.
  WRITE: / 'JOB_CLOSE failed, sy-subrc=', sy-subrc,
        'jobcount=', lv_jobcount,
        'pred=', lv_curr_jobname, lv_curr_jobcount.
ENDIF.
ENDFORM.

```

IMPORTANT: After you paste the code, make sure to replace the **<CONFIGURE_YOUR_VARIANT_HERE>** with the actual name of the Variant that you created for RBDAPP01 in the previous step.

Another option that can be changed is how often the program reschedules itself to check for more IDocs. The default is 60 seconds but can be increased if preferred.

The screenshot shows the SAP ABAP Editor interface for the report 'ZRDP_RUN_AND_RESCHEDULE'. The report is currently in 'Active' status. The code is as follows:

```
1 REPORT zrdp_run_and_reschedule.
2
3 CONSTANTS:
4   c_jobname      TYPE tbtctjob-jobname VALUE 'ZRDP_RUN_AND_RESCHEDULE',
5   c_variant      TYPE raldb_vari      VALUE 'ZRDP_PROCESS',
6   c_min_period_secs TYPE i            VALUE 60.
7
8 DATA:
9   lv_curr_jobname TYPE tbtctjob-jobname,
10  lv_curr_jobcount TYPE tbtctjob-jobcount,
11  lv_other_jobcount TYPE tbtctjob-jobcount.
12
13 IF sy-batch IS INITIAL.
14   MESSAGE 'Must run as background job' TYPE 'A'.
15 ENDIF.
16
17 SELECT COUNT(*) FROM varid WHERE report = 'RBDAPP01' AND variant = c_variant.
18 IF sy-subrc <> 0.
19   MESSAGE |Variant { c_variant } missing| TYPE 'A'.
20 ENDIF.
21
22 PERFORM schedule_next.
23
24 IF lv_curr_jobcount IS NOT INITIAL.
25   SELECT SINGLE jobcount FROM tbtco INTO lv_other_jobcount
26     WHERE jobname = c_jobname AND jobcount <> lv_curr_jobcount AND status = 'R'.
27   IF sy-subrc = 0.
28     RETURN.
29   ENDIF.
30 ENDIF.
31
32 SUBMIT rbdapp01 USING SELECTION-SET c_variant AND RETURN.
33
34 FORM schedule_next.
35   DATA: lv_jobcount TYPE tbtctjob-jobcount,
36         lv_ts        TYPE timestamp,
37         lv_sys_tz    TYPE ttzz-tzone,
38         lv_start_dt  TYPE sy-datum,
39         lv_start_tm  TYPE sy-uzeit.
40
41   CALL FUNCTION 'GET_JOB_RUNTIME_INFO'
42     IMPORTING jobname = lv_curr_jobname jobcount = lv_curr_jobcount
43     EXCEPTIONS OTHERS = 1.
44   IF sy-subrc <> 0 OR lv_curr_jobname IS INITIAL OR lv_curr_jobcount IS INITIAL.
```

Annotations in the image:

- 1:** Points to the constant `c_variant` and its value `'ZRDP_PROCESS'` in the `CONSTANTS` section.
- 2:** Points to the 'Save' icon in the top toolbar.
- 3:** Points to the 'Activate' icon in the top toolbar.
- 4:** Points to the 'Active' status indicator in the top right of the editor.

After you maintain the necessary code changes, make sure to **Save** and **Activate** the Program.

Execute Transaction: **SM36**

Enter a name for the job and a pick a class.

NOTE: Lower priority jobs may get delayed if there are other higher priority processes. Choose a higher job class if that is not convenient.

Then select Start Condition

The screenshot shows the SAP 'Define Job' (SM36) interface. At the top, there is a menu bar with 'Job', 'Edit', 'Goto', 'System', and 'Help'. Below the menu bar is a toolbar with various icons. The main title is 'SAP Define Job'. Below the title, there are several tabs: 'Start Condition' (highlighted with a red circle and a red '2'), 'Step', 'Job Selection', 'Own Jobs', 'Job Wizard', and 'Job Repository'. The 'General Data' section is visible, containing the following fields and buttons:

- Job Name:** ZRDP_RUN_AND_RESCHEDULE (highlighted with a red circle and a red '1')
- Job Class:** C
- Status:** Scheduled
- Exec. Target:** (empty field)
- E-Mail Notification:** (button)
- Spool List Recipient:** (button)

Select Immediate as Start Condition and Save.

IMPORTANT: Do NOT enable Periodic Job, the Program itself is self-recurring.

The screenshot shows a window titled "Start Time" with a dark theme. At the top, there are five buttons: "Immediate", "Date/Time", "After Job", "After Event", and "At Operation Mode", followed by a right-pointing arrow. The "Immediate" button is circled in red with a red "1" next to it. Below these buttons, the "Date/Time" section is visible, containing a checkbox labeled "Immediate Start" which is checked. The "After Job" and "At Operation Mode" sections are empty. The "After Event" section is also empty. At the bottom left, there is an unchecked checkbox labeled "Periodic Job". The bottom bar contains four buttons: "Check" (with a green checkmark icon), "Period Values" (with a document icon), "Restrictions", and a close button (with a red X icon). The "Period Values" button is circled in red with a red "2" next to it.

Click on the Step button.

Job Edit Goto System Help

✓ << Save << >> X Print Search Search+ Upload Upload Down Down Star ?

SAP Define Job

Start Condition **Step** Job Selection Own Jobs Job Wizard Job Repository

General Data

Job Name	ZRDP_RUN_AND_RESCHEDULE	
Job Class	C	
Status	Scheduled	E-Mail Notification
Exec. Target		Spool List Recipient

Enter Program Name: **ZRDP_RUN_AND_RESCHEDULE**, then Save.

Program values

ABAP program

Name  **1**

Variant

Language

External command (command pre-defined by system administrator)

Name

Parameters

OS

Target server

External program (direct command input by system administrator)

Name

Parameters

Target server

2

 **Check**   **Print Data** 

The Step List should now look like this:

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

Press the Back button and Save the Job.

It should now be displayed that the Job is Released.

General Data

Job Name: ZRDG_RUN_AND_RESCHEDULE

Job Class: C

Status: Scheduled

Exec. Target:

E-Mail Notification

Spool List Recipient

Verify that this is the case using transaction: **SM37**

If the Job was successfully activated, you should see one entry with status Released and subsequent entries a few minutes apart.

JobName	Spool	Job Doc	Scheduler	Status	Start date	Start Time	Duration(sec.)	Delay (sec.)	Cl	Reason for Delay
ZRDG_RUN_AND_RESCHEDULE				Released			0	41	002	
ZRDG_RUN_AND_RESCHEDULE				Finished	26.05.2026	10:22:38	1	25	002	
ZRDG_RUN_AND_RESCHEDULE				Finished	26.05.2026	10:23:43	0	4	002	
ZRDG_RUN_AND_RESCHEDULE				Finished	26.05.2026	10:24:43	1	0	002	
ZRDG_RUN_AND_RESCHEDULE				Finished	26.05.2026	10:26:03	428	19	002	
ZRDG_RUN_AND_RESCHEDULE				Finished	26.05.2026	10:34:37	0	26	002	
ZRDG_RUN_AND_RESCHEDULE				Finished	26.05.2026	10:36:10	0	33	002	
ZRDG_RUN_AND_RESCHEDULE				Finished	26.05.2026	10:38:23	1	73	002	
ZRDG_RUN_AND_RESCHEDULE				Finished	26.05.2026	10:40:23	1	60	002	
*Summary							432	281		

After this point IDocs should be processing automatically even in large quantities without serialization issues.

Option B: Schedule the IDoc Processing Program – RBDAPP01

Execute Transaction: **SM36**

Enter a name for the job and a choose a class.

NOTE: Lower priority jobs may get delayed if there are other higher priority processes. Choose a higher job class if that is not convenient.

Then select Start Condition

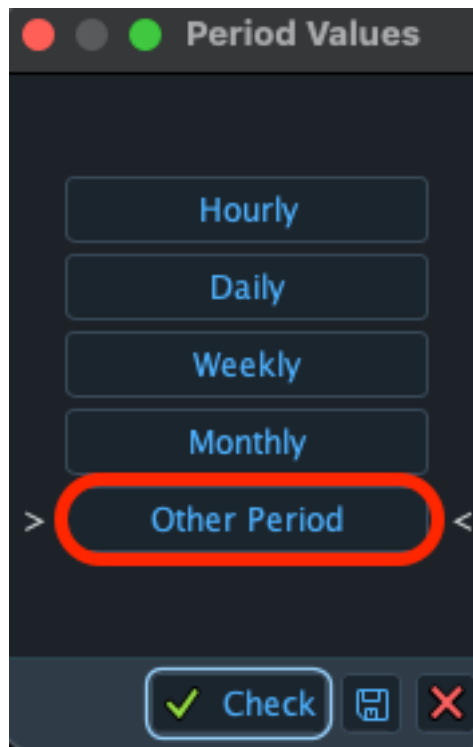
The screenshot displays the SAP 'Define Job' (SM36) interface. The 'Start Condition' tab is selected and highlighted with a red circle and a red number '2'. The 'General Data' section is visible, with the 'Job Name' field containing 'ZRDP_PROCESS_IDOCS' and the 'Job Class' field containing 'C'. This section is also highlighted with a red circle and a red number '1'. The 'Status' is set to 'Scheduled' and the 'Exec. Target' is empty. The 'E-Mail Notification' and 'Spool List Recipient' buttons are visible on the right.

Field	Value
Job Name	ZRDP_PROCESS_IDOCS
Job Class	C
Status	Scheduled
Exec. Target	

Select Immediate as Start Condition, enable Periodic Job and choose Period Values.

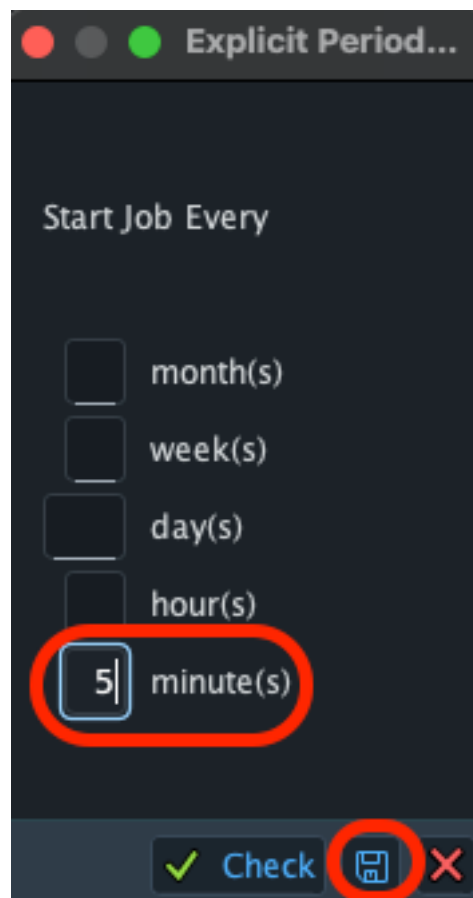
The screenshot shows a 'Start Time' configuration window with a dark theme. At the top, there are five buttons: 'Immediate', 'Date/Time', 'After Job', 'After Event', and 'At Operation Mode', followed by a right-pointing arrow. The 'Immediate' button is circled in red and labeled with a red '1'. Below these buttons, the 'Date/Time' section contains a checkbox labeled 'Immediate Start' which is checked. The 'After Job' and 'At Operation Mode' sections each have a large empty rectangular box. The 'After Event' section also has a large empty rectangular box. At the bottom left, there is a checkbox labeled 'Periodic Job' which is checked and circled in red, labeled with a red '2'. At the bottom right, there is a row of buttons: a green checkmark icon followed by 'Check', a floppy disk icon followed by 'Period Values' (circled in red and labeled with a red '3'), and a red 'X' icon followed by 'Restrictions'.

Specify Other Period.



The 'Period Values' dialog box features a dark background with five buttons: 'Hourly', 'Daily', 'Weekly', 'Monthly', and 'Other Period'. The 'Other Period' button is highlighted with a red oval. At the bottom, there is a 'Check' button with a green checkmark, a save icon, and a red 'X' icon.

Enter the desired frequency. In this example the job will be executed every 5 minutes.



The 'Explicit Period...' dialog box has a dark background. Under the heading 'Start Job Every', there are five input fields with labels: 'month(s)', 'week(s)', 'day(s)', 'hour(s)', and 'minute(s)'. The 'minute(s)' field contains the number '5' and is highlighted with a red oval. At the bottom, there is a 'Check' button with a green checkmark, a save icon (highlighted with a red oval), and a red 'X' icon.

Check that Job Start time and the frequency are correct, then click on the Step button.

SAP Define Job

Start Condition **Step** Job Selection Own Jobs Job Wizard Job Repository

General Data

Job Name	ZRDP_PROCESS_IDOCS	
Job Class	C	
Status	Scheduled	E-Mail Notification
Exec. Target		Spool List Recipient

Job Start

Date/Time
☒ Immediate start

Job Frequency

5 Minute(s)

Enter:

- Program Name: **RBDAPP01**
- Variant: The name of your variant (**ZRDP_PROCESS** in this example)

Then Save.

Program values

ABAP program External command External program

ABAP program

Name: RBDAPP01

Variant: ZRDP_PROCESS  1

Language: EN

External command (command pre-defined by system administrator)

Name: _____

Parameters: _____

OS: _____

Target server: _____

External program (direct command input by system administrator)

Name: _____

Parameters: _____

Target server: _____

2

✓ Check   Print Data ✕

The Step List should now look like this:

No.	Program name/command	Prog. type	Spool list	Parameters	User	Lang.
1	RBDAPP01	ABAP		ZRDP_PROCESS		EN

Press the Back button and Save the Job.

It should now be displayed that the Job is Released.

General Data

Job Name: ZRDP_PROCESS_IDOCS

Job Class: C

Status: Scheduled

Exec. Target:

Job Start

Date/Time:

☒ Immediate start

Job Frequency

5 Minute(s)

Verify that this is the case using transaction: **SM37**

If the Job was successfully activated, you should see one entry with status Released and subsequent entries a few minutes apart as specified by the period.

JobName	Spool	Job Doc	Scheduler	Status	Start date	Start Time	Duration(sec.)	Delay (sec.)	Cl	Reason for Delay
☐ ZRDP_PROCESS_IDOCS				Released			0	0	002	
☐ ZRDP_PROCESS_IDOCS				Finished	29.05.2026	14:20:45	0	0	002	
*Summary										
							0	0		

After this point IDocs should be processing automatically.