Automation of SAP PI Queue Monitoring using SAP-ABAP Program

Shashikant R. Chandratre,
SAP Consultant

***Abstract* – Objective of this journal is creating an automation for manual background job monitoring of SAP PI Queue. SAP NetWeaver Process Integration (SAP PI) is SAP enterprise application integration (EAI) software, a component of the NetWeaver product group used to facilitate the exchange of information among a company's internal systems and external parties. Connection to external system is via SAP interfaces. Currently if we need to check inbound and outbound SAP PI queue from multiple SAP system (We are explicitly talking about SAP 4.7. version here) then someone needs to manually get into the system and check inbound and outbound queue. IF they are stuck then SAP PI consultant will manually release them. Most of the time there are no queues stuck and person just needs to go in to the system and check and come out of the system. That time this become time consuming activity for no reason. This article is mainly focusing on monitoring those interfaces – inbound and outbound; Sending email notification to concerns parties without manual intervention. ABAP (Advance Business Application Programming) program will perform this activity that will be scheduled as background job in SAP system will monitor the stuck queue in SAP PI system. Program will also send notification to relevant parties about stuck queue to take the appropriate action. This automation will save manual efforts of monitoring which ultimately will save money for business. Also monitoring here will be more precise in time and over weekends/holidays. Total savings hours of savings from this automation is estimated around 30 hours per month if resource is doing automation every day and takes 1 hour.**

***Keywords-Component* - SAP-PI; ABAP; SAP PI Queue Monitoring; ERP**

INTRODUCTION:

SAP stands for Systems, Applications, and Products in Data Processing is one of the most highly integrated software is being used in many industries now. With highly integration capabilities SAP PI is mainly used to interact with external system for exchanging business-related data. SAP has interfaces which interact with external system via middleware as SAP PI. This middle has both inbound and outbound queues monitored in SMQ1 transaction. Someone always needs to monitor those interfaces for any failures and queues were getting stuck. In this automation ABAP Program will be created which is going to monitor those interfaces and will trigger an email to notify relevant person when it fails. This ABAP program (refer [1] and [2]) will be scheduled as background job for execution at certain frequency on daily basis. This article will cover code written in ABAP to read inbound/outbound interface status. Then to trigger an email notification for the failed interface so that respective team can take an action on it.

Problem Statement – Manual monitoring is time consuming and involved lots manual efforts and resulted financial cost to an organization. To handle this issue, we are proposing this solution of automation.

Normal time of monitoring for 1 person per day is 3- 4 hours which include preparing report and sending an email to respective teams or leadership in an organization. This time is based on experiential narrative.

Also, this monitoring is not precise as person who is going to do monitoring and report generation may be absent or may not come to office. Also Monitoring may not happen on holidays and weekend. Organization needs to pay extras for this amount.

This could be resulted of 60 hours per months and money can be calculated based on that as per the rates.

Architectural Diagram of Automation Process – Fig.1. are the overall system data flow diagram shows flow of data. This diagram mainly consists multiple system systems namely SAP1, SAP2 and SAP3 (for explanation of concepts we have taken three systems). SAP PI – This is the middleware between external system and SAP system. It connects interfaces between SAP and External systems. Here SAP PI connects external systems – ES1, ES2 and ES3 to their respective SAP systems SAP1 or SAP2 or SAP3. Those are nothing but interfaces being used. Inbound and or outbound interfaces which needs to be monitored in each SAP systems SAP1. SAP2 and SAP3. ABAP program ZSAP_PI_QUEUE which is being used for monitoring of stuck queues from SAP1, SAP2 and SAP3 is written in system SAP1. Since this program uses RFC enable function module that can read the stuck queues from SAP2 and SAP 3 system as well hence program in SAP1 system will give you overall monitoring results from all the systems. This is one of the advantages of this automation that we can monitor multiple system at once.

Fig.1. shows bidirectional data flow arrows from external system ES1, ES2 and ES3 to SAP1, SAP2 and SAP 3 via SAP PI middleware.

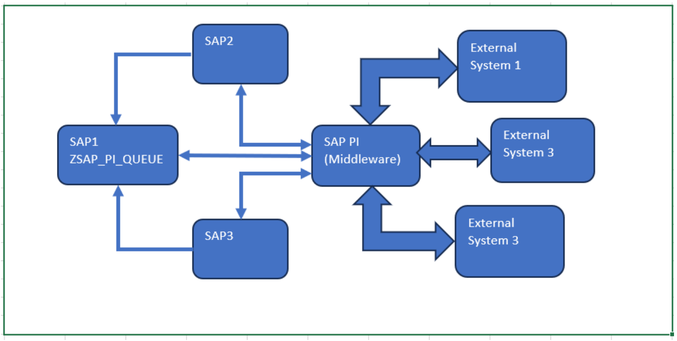


Fig.1. Systems data flow diagram

To implement this solution, one has to have expertise in SAP -ABAP code and have background of SAP PI monitoring. expert

Program code and process of execution – As mentioned in the Fig. 1. Create the custom ABAP Program in SAP1 system with naming ZSAP_PI_QUEUE to be scheduled in background mode. ABAP code can be referenced from Ref [1] , Ref [2] and Ref[3]. This program will use the standard function

module 'TRFC_QOUT_GET_CURRENT_QUEUES' to fetch current outbound queue details from the executable systems - SAP1, SAP2 and SAP3 store them in internal table LT_QUEUE_OUT with fields structure like TRFCQVIEW.

Use below piece of code to get outbound queue and store them in internal table LT_QUEUE_OUT

```
CALL
FUNCTION 'TRFC_QOUT_GET_CURRENT_QUEUES'
      EXPORTING
           Dest          = '*'
           Client        = sy-mandt
      TABLES
           Qview         = lt_queue_out
```

Similarly use standard function module 'TRFC_QIN_GET_CURRENT_QUEUES' to fetch current inbound queue details from the executable system - SAP1, SAP2 and SAP3 store them in internal table LT_QUEUE_IN which has fields structure like TRFCQVIEW. Use below piece of code to get inbound queue and store them in internal table LT_QUEUE_IN

```
CALL FUNCTION 'TRFC_QIN_GET_CURRENT_QUEUES'
      EXPORTING
           Dest          = '*'
           Client        = sy-mandt
      TABLES
           Qview         = lt_queue_in
```

These inbound queues and outbound queues are nothing but these are the queues which got stuck SAP system SAP 1, SAP2 and SAP3. We have to write the code with logical system of SAP2 and SAP3 to get their stuck inbound and outbound queues in SAP1 system. With this program we have all the stuck queues from current system SAP1 and remote system SAP2 and SAP3.

To trigger the notification from the program we need email addresses for that create one custom table ZEMAIL_ADDR_MAINT shown in TABLE 1 to maintain the email addresses of peoples who should get notification on stuck queues. Create table maintenance generator for that table so the email addresses can be maintained (Add or Delete or Change) based on the requirement via SM30. Program will the pick the e-mail id from table: ZEMAIL_ADDR_MAINT and triggered the queue report to list of recipients maintained in the table.

First Name (CHAR 50)	Last Name (CHAR 50)	Email Address (CHAR200)	Email CC
ABC	XYZ	ABC.XYZ@test.com	X

TABLE I*:* STRUCTURE OF ZEMAIL_ADDR_MAINT

To write code for email triggering in program ZSAP_PI_QUEUE create one internal table LT_EMAIL_ADD having structure like ZEMAIL_ADDR_MAINT to fetch list of email addresses to send the stuck queue status. If the field Email CC in the table have value as 'X' for email address, then that email addressed should be in CC for email sending out functionality and wherever its blank that email address should be in TO list. Write an ABAP code email triggering functionality. There are multiple ways to write email triggering code e.g., Use standard function module or use standard ABAP class. As mentioned, to write this program one has to have deep ABAP coding knowledge.

When Program code execute it will pick up complete list inbound queues from LT_QUEUE_IN and outbound queues from LT_QUEUE_OUT. These queues contain numbers means that systems have failed jobs. With the help of ABAP code, we can explore and create HTML report and send as an attachment in addition to summary as below in table format.

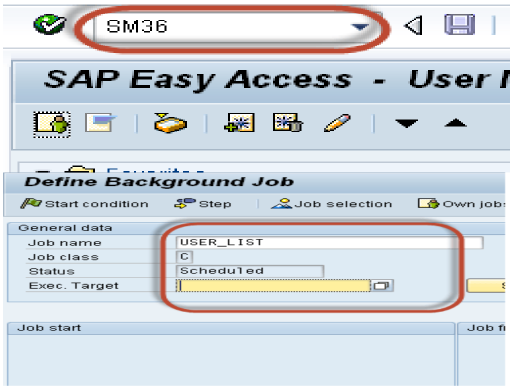
SAP System	Cancelled Job Status
SAP1	XXX number of jobs cancelled.
SAP2	XXX number of jobs cancelled.

TABLE II*:* STRUCTURE OF EMAIL CONTENT

Above process completes the ABAP code and its execution. To automate this process of monitoring stuck queues once has to schedule this as background job with certain frequency of execution. Below are the details on background job scheduling.

Background Job Scheduling – ABAP program ZSAP_PI_QUEUE need to be scheduled as background job with frequency of execution as many times as we want. Below are detailed steps as below.
You can schedule the background job using SAP transaction SM36. Planned or immediate jobs can be scheduled. In this case its planned.

Execute SM37 as shown in **Fig.2** Fill the job name, priority(A/B/C) and the target server. Background jobs once scheduled on a target server run on that server. Main purpose of defining target server is the workload balancing.


SM36

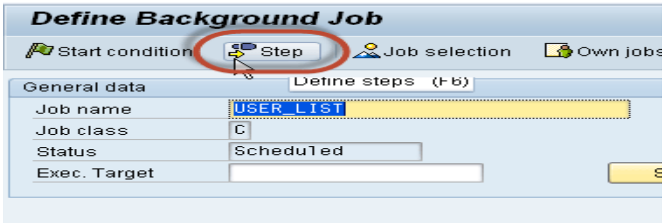
SAP Easy Access - User I

Define Background Job

General data
Job name: USER_LIST
Job class: C
Status: Scheduled
Exec. Target:

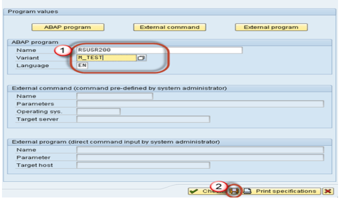
Fig.2. Background Job Scheduling

Click Step button to define ABAP program, variant's details, etc. as shown in **Fig 3**.


Define Background Job

General data
Job name: USER_LIST
Job class: C
Status: Scheduled
Exec. Target:

Define program name, variant details. Enter your program name, Variant name in the field. If you have not created variant as per your requirement, then leave it blank and then press save button. **Fig. 4.**


Program values

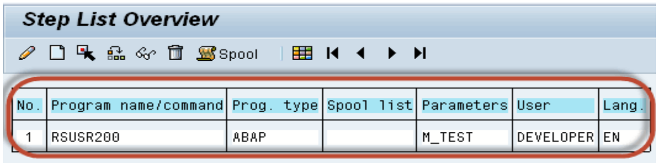
ABAP program
Name: RSUSR200
Variant: M_TEST
Language: EN

External command (command pre-defined by system administrator)
Name:
Parameters:
Operating sys.:
Target server:

External program (direct command input by system administrator)
Name:
Parameter:
Target host:

Fig. 4.

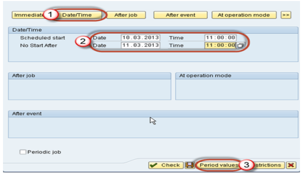
Once you scheduled the job you will get the following screen


Step List Overview

No.	Program name/command	Prog. type	Spool list	Parameters	User	Lang.
1	RSUSR200	ABAP		M_TEST	DEVELOPER	EN

Click Start conditions to fill start date, end date, frequency, etc. for job. If you do not specify start condition then job will always remain in scheduled status. A job in scheduled status will never run. Click on Date/Time (For periodic jobs). If you click "Immediate" then job will start running right away. But it will not be set as periodic job. It's like "press and run." Define job's start date/time, end date/time. The job will be released only once it meets its Scheduled start date/time. Press periodic values as shown in **Fig. 5**

then Click on Hourly/Daily/Weekly period to define the frequency of the job as per your requirement. We will select Other Period as shown in **Fig. 6**. and **Fig. 7** Here you specify the recurring criteria of the job. For example, you can have the Job run after every 5 days from the Start Date. Here we select job to run every 10 minutes after clicking on save as per figure **Fig. 8.**


Date/Time
Scheduled start: Date 10.03.2013 Time 11:00:00
No Start After: Date 11.03.2013 Time 11:00:00

□ Periodic job

Fig. 5

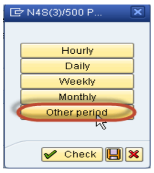


Fig.6.

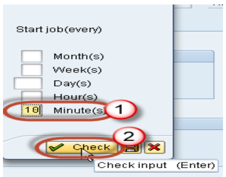

Start job(every)
□ Month(s)
□ Week(s)
□ Day(s)
□ Hour(s)
10 Minute(s)

Fig. 7

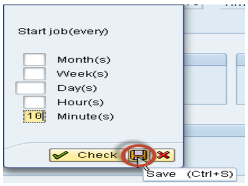

Start job(every)
□ Month(s)
□ Week(s)
□ Day(s)
□ Hour(s)
10 Minute(s)

Fig. 8

All above are the steps to schedule the job and ABAP expert is aware on it.

BENEFITS:

1. This automation will have zero dependencies with manual intervention.
2. Monitoring happens automatically for multiple SAP systems.
3. It can save many hours of manual monitoring and leads to financial benefits to the business.
4. Since there are no dependencies of human intervention on monitoring hence it can be done over holidays, weekends midnights and anytime.

CONCLUSION: This is just solution for one of the issues and can be handled at large scale with more advanced features of SAP latest version.

REFERENCES:

[1] Complete ABAP: The Comprehensive Guide to SAP ABAP by Kiran Bandari.
[2] ABAP to the Future written by Paul Hardy.
[3] Function Modules in ABAP Quick Reference Guide Book by Tanmaya Gupta.