

Broadcasting eMails at Custom Intervals after the Completion of Data Loads



Applies to:

SAP NetWeaver BW 3.x. For more information, visit the [Business Intelligence homepage](#).

Summary

The objective of this article is to explain the business requirement of automating the process of triggering the data loads status mails to teams on specific days of a week. This is in continuation to my previous article "[Scheduling of Process Chains at Custom Intervals Using Function Modules and Events](#)."

Author: Rudra Pradeep Reddy Neelapu

Company: Mahindra Satyam

Created on: 16 February 2010

Author Bio



Working as a SAP Technical consultant with Mahindra Satyam. Skill set includes SAP Business Intelligence, ABAP and Business Objects.

Table of Contents

Introduction	3
Business Scenario	3
Background Information.....	3
Procedure	3
Code	8
Related Content.....	16
Disclaimer and Liability Notice.....	17

Introduction

The objective of this article is to explain the business requirement of triggering the alert mails when the data loads process is completed along with the Process Chain Load Status at Custom time intervals.

Business Scenario

Need to trigger Data Loads Status mails to team after the data from R/3 system to our BW systems gets refreshed on Specific days of a week. Mails need to be triggered on every Monday, Wednesday, Thursday and Friday. Because of Custom time intervals we cannot go out with any of the SAP delivered periodic scheduling options because of the customized requirement.

Background Information

Create the flow to capture the Process Chain attribute data from tables RSPCPROCESSLOG (Logs for the Chain Runs) and RSPCLOGCHAIN (Cross-Table Log ID / Chain ID) at BW end.

Create the Report Program which triggers the mail content to the Team.

Once the loads are completed a variant need to be fired to start the loading of Process Chain attributes data. The last variant of this chain is your Report Program which can be embedded using Process Type → General Services → ABAP Program.

Prerequisite

Schedule your Process Chains to trigger at Custom Intervals Using Function Modules and Events (explained in Article **Scheduling of Process Chains at Custom Intervals Using Function Modules and Events**

<http://www.sdn.sap.com/irj/scn/go/portal/prtroot/docs/library/uuid/8068a6b6-40e3-2c10-22ab-d52c5ab48f90?QuickLink=index&overridelayout=true>)

Your Process chain with attribute data should be fired at the end of the data load Process.

Procedure

Tables RSPCPROCESSLOG (Logs for the Chain Runs) and RSPCLOGCHAIN (Cross-Table Log ID / Chain ID) contains the Chains attribute data. We are creating a View on both these tables. SE11 → View → ZPC_TIME (Create) with required fields.

ABAP Dictionary: Initial Screen

The screenshot shows the 'ABAP Dictionary: Initial Screen' for creating a new object. The 'View' radio button is selected, and the object name 'ZPC_TIME' is entered in the text field. Below the text field is a small icon of a document with a plus sign. The other radio buttons are 'Database table', 'Data type', 'Type Group', 'Domain', 'Search help', and 'Lock object', all of which are unselected. At the bottom, there are three buttons: 'Display' (with a magnifying glass icon), 'Change' (with a pencil icon), and 'Create' (with a document icon).

Maintain the Table/Join Conditions on table RSPCPROCESSLOG and RSPCLOGCHAIN

Database view: ZPC_TIME Active
Short Text: PC FOR LOAD TIME

Attributes | **Table/Join Conditions** | View Flds | Selection Conditions | Maint.Status

Table	Field name	=	Table	Field name
RSPCPROCESSLOG	LOG_ID	=	RSPCLOGCHAIN	LOG_ID

Maintain the Fields in the View

Database view: ZPC_TIME Active
Short Text: PC FOR LOAD TIME

Attributes | Table/Join Conditions | **View Flds** | Selection Conditions | Maint.Status

View field	Mod	DTyp	Length	Short description
LOG_ID	<input type="checkbox"/>	CHAR	25	Log-ID of a Process Chain Run
BATCHDATE	<input type="checkbox"/>	DATS	8	Release Date for Background Scheduling
BATCHTIME	<input type="checkbox"/>	TIMS	6	Release time of scheduled background job
ACTUAL_STATE	<input checked="" type="checkbox"/>	CHAR	1	Status of Process
CHAIN_ID	<input type="checkbox"/>	CHAR	25	Process Chain

Create the Data Source ZDS_PCLOADTIME based on the View (ZPC_TIME) at BW end.

Maintain Generic DataSources

DataSource

Transaction data ZDS_PCLOADTIME

Master Data Attributes

Texts

Create Change Display

Display DataSource for Transactn data: ZDS_PCLOADTIM

Generic Delta

DataSource	ZDS_PCLOADTIME
Applic. Component	DM-IO
Data Reconciliation	<input type="checkbox"/>
Obj. status	Saved

Texts

Short description	DS FOR PC LOADTIME
Medium description	DS FOR PC LOADTIME
Long description	DS FOR PC LOADTIME

Extraction from DB View

View/Table	ZPC_TIME
ExtractStruct.	Z0XDEV0135

Extraction frm SAP Query

InfoSet	
---------	--

Extraction by Function Module

Function Module	
Extract.Struct.	

Replicate and assign the Data Source to the ODS. Create a ODS to store process chain attribute data with structure as that of Source.

ODS FOR PC'S LOAD TIME			ZPC_TIME
INFOSOURCE FOR PCLOADTIME			ZIS_PCLOADTIME

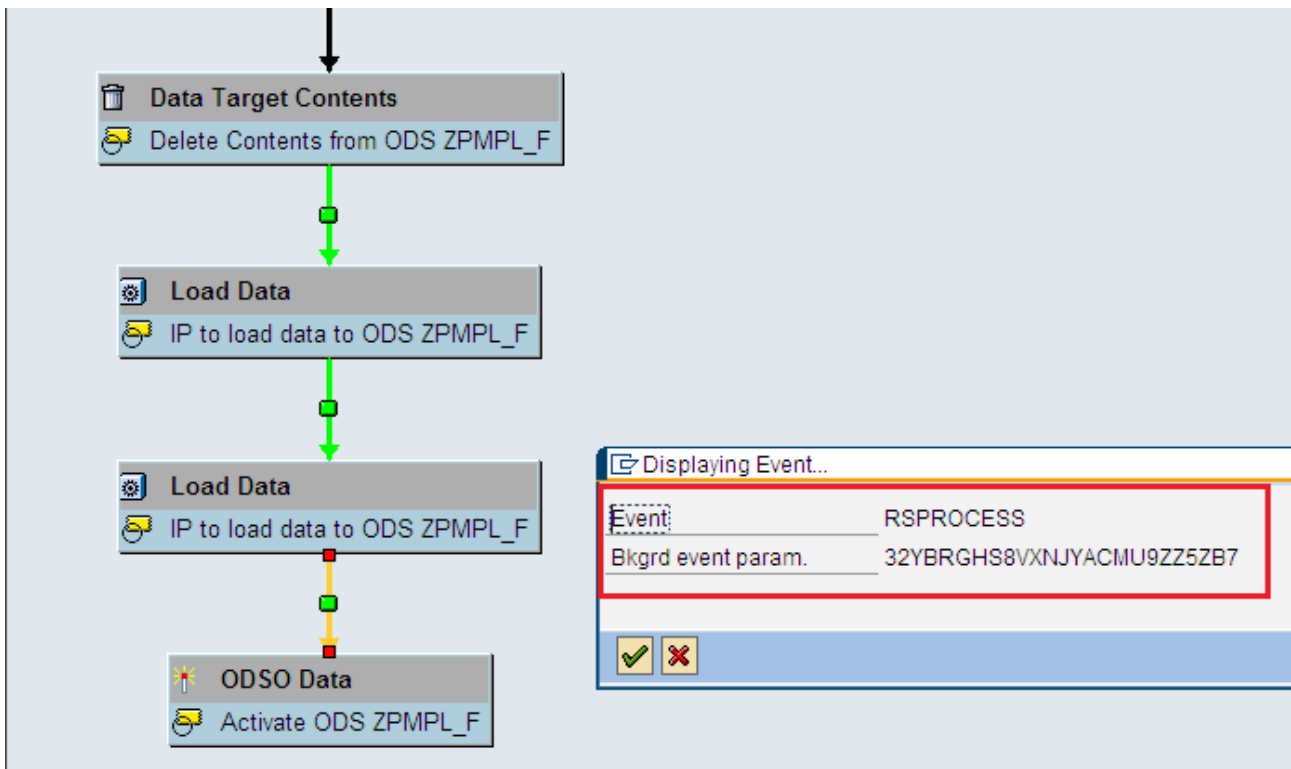
Transfer Structure of the same:

The screenshot shows the SAP Transfer Structure configuration for Transfer Rules. The Source System is BIDCLN100 - Logical system for BW I, and the DataSource is ZDS_PCLOADTIME - DS FOR PC L... The Status is Active(Saved). The Transfer Method is set to PSA. The configuration is divided into two main sections: Communication str./Transfer rules and Assign.InfObject-field.

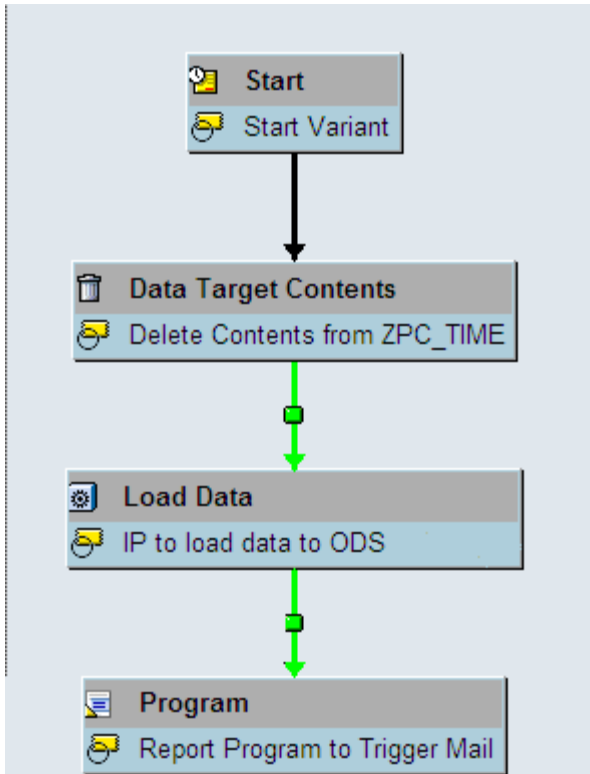
InfoObject	Descript.	Tp	Rule
ZLOGID	PC LOGID	←	ZLOGID
ØDATE_KYF	Date	←	ØDATE_KYF
ØTIME_KYF	Time	←	ØTIME_KYF
ZPC_STATE	STATUS OF ...	←	ZPC_STA
ZCHAIN_ID	PROCESSSC...	←	ZCHAIN_
ØRECORDMODE	Update Mode	←	ØRECORDMODE

InfoObject	Descript.	Field
ZLOGID	Log-ID of a ...	LOG_I
ØDATE_KYF	Release Da...	BATCH
ØTIME_KYF	Release tim...	BATCH
ZPC_STATE	Status of Pr...	ACTU
ZCHAIN_ID	Process Ch...	CHAIN

Create a Process chain to get this attribute data to be loaded into the ODS **ZPC_TIME**. This PC need to fire at event which is obtained at the last Process of your loads chain which was customized to run at user intervals.



At the Start Variant of your Attributes Chain maintain the Event and Event Parameter as shown below.
 Start Variant → Maintain Start Process → Change selection and maintain the Event Parameter.



Start Time

>>

Date/Time

After job

At operation mode

After event

Event:

Parameter:

Periodic job

In the Program Variant embed the code:

Code

```

*****CODE*****
*&-----*
*& Report  ZBW_ITAB_MAIL_NR
*&
*&-----*
*&
*&
*&-----*
REPORT  zbw_itab_mail_nr.
*Declare a types to hold the Process Chain Related data
TYPES:BEGIN OF ty_chain,
        /bic/zlogid TYPE /bic/oizlogid,
        /bic/zchain_id TYPE /bic/oizchain_id,
        date_kyf TYPE /bi0/oidate_kyf,
        /bic/zst_time TYPE /bic/oizst_time,
        /bic/zend_time TYPE /bic/oizend_time,
        time_kyf TYPE /bi0/oitime_kyf,
        /bic/zpc_state TYPE /bic/oizpc_state,
END OF ty_chain.
*Declare types for Process Chain data table
TYPES:tt_chain TYPE STANDARD TABLE OF ty_chain.
*Declare a work area to hold the Process Chain data
DATA: gf_pc TYPE ty_chain.
*Declare a table to hold the Employee data
DATA: gt_chain TYPE tt_chain.
*All the declarations to send the email is same for this also
*doc type specify as HTML
DATA:
*- Structure to hold Output Details in delimited format
        gf_output_soli TYPE soli,
*- Structure to hold Imported Object Components
        gf_objpack TYPE sopcklsti1,
*- Structure to hold data in single line format
        gf_objhead TYPE solisti1,
*- Structure to hold data in single line format
        gf_objtxt TYPE solisti1,
*- Structure to hold data for API Recipient List
        gf_reclist TYPE somlreci1,
*- Structure to hold email data
        gf_doc_chng TYPE sodocchgi1,

```


*- Internal Table to hold Output Details in delimited format

gt_output_soli TYPE TABLE OF soli,

*- Internal Table to hold Imported Object Components

gt_objpack TYPE TABLE OF sopcklsti1,

*- Internal Table to hold data in single line format

gt_objhead TYPE TABLE OF solisti1,

*- Internal Table to hold data in single line format

gt_objtxt TYPE TABLE OF solisti1,

*- Internal Table to hold data for API Recipient List

gt_reclist TYPE TABLE OF somlreci1.

DATA:l_date(10),

l_start(10),

l_end(10),

l_endstatus(10),

l_time(10).

DATA: g_lines TYPE sy-tabix, "To hold number of records

g_msg_lines TYPE sy-tabix, "To hold number of records

g_sent_all(1) TYPE c.

**

C O N S T A N T S

**

CONSTANTS : c_int(3) TYPE c VALUE 'INT', "Internet mail address

c_doc_type_htm(3) TYPE c VALUE 'HTM', "Code for document class

c_rec_type TYPE c VALUE 'U', "Recipient type

c_express TYPE c VALUE 'X'. "Send express

START-OF-SELECTION.

*select to get the Process Chain Data

SELECT /bic/zlogid /bic/zchain_id date_kyf

/bic/zst_time /bic/zend_time time_kyf /bic/zpc_state

FROM /bic/azpc_time00 INTO TABLE gt_chain.

*html Body format

*Set the Subject line

gf_doc_chng-obj_name = 'Data Loads Completed'.

gf_doc_chng-obj_descr = 'Data Loads Completed'.

*Set the Body background colour

gf_objtxt-line = '<body bgcolor = "#E6E6FA">'.

*- Append

APPEND gf_objtxt TO gt_objtxt.

*- Clear

CLEAR gf_objtxt.

*Set font color and its type

CONCATENATE '' '' INTO gf_objtxt-line.

*- Append

```

APPEND gf_objtxt TO gt_objtxt.
*- Clear
CLEAR gf_objtxt.
*Pripare mail body
CONCATENATE '<p>' 'Dear Team,' '</p>'
            INTO gf_objtxt-line.
*- Append
APPEND gf_objtxt TO gt_objtxt.
*- Clear
CLEAR gf_objtxt.
gf_objtxt-line = space.
*- Append
APPEND gf_objtxt TO gt_objtxt.
*- Clear
CLEAR gf_objtxt.
CONCATENATE '<p>'
            '    Data Loads are Completed, below are the details.'
            '</p>'
            INTO gf_objtxt-line SEPARATED BY space.
APPEND gf_objtxt TO gt_objtxt.
CLEAR gf_objtxt.
gf_objtxt-line = '<center>'.
APPEND gf_objtxt TO gt_objtxt.
CLEAR gf_objtxt.
* Prepare chain data in table format to dispay in Mail body
gf_objtxt-line = '<TABLE width= "100%" border="1">'.
APPEND gf_objtxt TO gt_objtxt.
CLEAR gf_objtxt.
CONCATENATE '<TR ><td align = "LEFT" BGCOLOR = "#708090">'
            '<FONT COLOR = "BLUE"><B>Process Chain</B> </FONT>'
            '</td>' INTO gf_objtxt-line.
APPEND gf_objtxt TO gt_objtxt.
CLEAR gf_objtxt.
CONCATENATE '<td align = "LEFT" BGCOLOR = "#708090">'
            '<FONT COLOR = "BLUE"> <B>Date</B> </FONT>'
            '</td>' INTO gf_objtxt-line.
APPEND gf_objtxt TO gt_objtxt.
CLEAR gf_objtxt.
CONCATENATE '<td align = "LEFT" BGCOLOR = "#708090">'
            '<FONT COLOR = "BLUE"><B>Start Time</B> </FONT>'
            '</td>' INTO gf_objtxt-line.

```

```

APPEND gf_objtxt TO gt_objtxt.
CLEAR gf_objtxt.
CONCATENATE '<td align = "LEFT" BGCOLOR = "#708090">'
            '<FONT COLOR = "BLUE"><B>End Time</B> </FONT>'
            '</td>' INTO gf_objtxt-line.
APPEND gf_objtxt TO gt_objtxt.
CLEAR gf_objtxt.
CONCATENATE '<td align = "LEFT" BGCOLOR = "#708090">'
            '<FONT COLOR = "BLUE"><B>Actual Load Time</B> </FONT>'
            '</td>' INTO gf_objtxt-line.
APPEND gf_objtxt TO gt_objtxt.
CLEAR gf_objtxt.
CONCATENATE '<td align = "LEFT" BGCOLOR = "#708090">'
            '<FONT COLOR = "BLUE"><B>Status</B> </FONT>'
            '</td></tr>' INTO gf_objtxt-line.
APPEND gf_objtxt TO gt_objtxt.
CLEAR gf_objtxt.
LOOP AT gt_chain INTO gf_pc.
  WRITE: gf_pc-date_kyf TO l_date.
  WRITE: gf_pc-/bic/zst_time TO l_start.
  WRITE: gf_pc-/bic/zend_time TO l_end(10).
  WRITE: gf_pc-time_kyf TO l_time(10).
  WRITE: gf_pc-/bic/zpc_state TO l_endstatus(10).
  CONCATENATE '<TR><td align = "LEFT">'
              '<FONT COLOR = "BLUE">' gf_pc-/bic/zchain_id '</FONT>'
              '</td>' INTO gf_objtxt-line.
  APPEND gf_objtxt TO gt_objtxt.
  CLEAR gf_objtxt.
  CONCATENATE '<td align = "LEFT">'
              '<FONT COLOR = "BLUE">' l_date '</FONT>'
              '</td>' INTO gf_objtxt-line.
  APPEND gf_objtxt TO gt_objtxt.
  CLEAR gf_objtxt.
  CONCATENATE '<td align = "LEFT">'
              '<FONT COLOR = "BLUE">' l_start '</FONT>'
              '</td>' INTO gf_objtxt-line.
  APPEND gf_objtxt TO gt_objtxt.
  CLEAR gf_objtxt.
  CONCATENATE '<td align = "LEFT">'
              '<FONT COLOR = "BLUE">' l_end '</FONT>'
              '</td>' INTO gf_objtxt-line.

```

```

APPEND gf_objtxt TO gt_objtxt.
CLEAR gf_objtxt.
CONCATENATE '<td align = "LEFT">'
            '<FONT COLOR = "BLUE">' l_time '</FONT>'
            '</td>' INTO gf_objtxt-line.
APPEND gf_objtxt TO gt_objtxt.
CLEAR gf_objtxt.
CONCATENATE '<td align = "LEFT">'
            '<FONT COLOR = "BLUE">' l_endstatus '</FONT>'
            '</td></tr>' INTO gf_objtxt-line.
APPEND gf_objtxt TO gt_objtxt.
CLEAR gf_objtxt.
ENDLOOP.
gf_objtxt-line = '</TABLE>'.
APPEND gf_objtxt TO gt_objtxt.
CLEAR gf_objtxt.
gf_objtxt-line = '</center>'.
APPEND gf_objtxt TO gt_objtxt.
CLEAR gf_objtxt.
CONCATENATE '<p>'
            space
            '</p>'
            INTO gf_objtxt-line SEPARATED BY space.
APPEND gf_objtxt TO gt_objtxt.
CLEAR gf_objtxt.
gf_objtxt-line = space.
*- Append
APPEND gf_objtxt TO gt_objtxt.
gf_objtxt-line = space.
*- Append
APPEND gf_objtxt TO gt_objtxt.
CLEAR gf_objtxt.
CONCATENATE '<p>'
            'Regards,<br />'
            '</p>'
            INTO gf_objtxt-line SEPARATED BY space.
APPEND gf_objtxt TO gt_objtxt.
CLEAR gf_objtxt.
gf_objtxt-line = 'INSIGHT SAP BW TEAM<br />'.
APPEND gf_objtxt TO gt_objtxt.
CLEAR gf_objtxt.

```

```

gf_objtxt-line = '<br><br><b><center><i><font color = "RED">This is an auto generated
Email.'.
  APPEND gf_objtxt TO gt_objtxt.
  CLEAR gf_objtxt.
gf_objtxt-line = '</FONT></body>'.
*- Append
  APPEND gf_objtxt TO gt_objtxt.
*- Clear
  CLEAR gf_objtxt.
END-OF-SELECTION.
  DESCRIBE TABLE gt_objtxt LINES g_msg_lines.
  READ TABLE gt_objtxt INTO gf_objtxt INDEX g_msg_lines.
  gf_doc_chng-doc_size = ( g_msg_lines - 1 ) * 255 + STRLEN( gf_objtxt ).
*- Creation of the entry for the compressed document
  gf_objpack-transf_bin = ' '.
  gf_objpack-head_start = 1.
  gf_objpack-head_num   = 0.
  gf_objpack-body_start = 1.
  gf_objpack-body_num   = g_msg_lines.
  gf_objpack-doc_type   = c_doc_type_htm.
*- Append
  APPEND gf_objpack TO gt_objpack.
*- Clear
  CLEAR gf_objpack.
*- Creation of the document attachment
*- Describe
  DESCRIBE TABLE gt_output_soli LINES g_lines.
*- Don't create attachment if no data is present
  IF g_lines <> 0.
    LOOP AT gt_output_soli INTO gf_output_soli.
      gf_objtxt = gf_output_soli.
*- Append
      APPEND gf_objtxt TO gt_objtxt.
*- Clear
      CLEAR gf_objtxt.
    ENDLOOP.
  ENDIF.
*- Completing the recipient list
  TYPES:BEGIN OF ty_mail,
    /bic/z_email TYPE /bic/oiz_email,
  END OF ty_mail.
  TYPES:tt_mail TYPE STANDARD TABLE OF ty_mail.

```

```

DATA: it_mail TYPE tt_mail,
      wa_mail TYPE ty_mail.
SELECT /bic/z_email FROM /bic/tz_email INTO TABLE
      it_mail WHERE /bic/z_email NE ''.
LOOP AT it_mail INTO wa_mail.
  gf_reclist-receiver = wa_mail-/bic/z_email.
  gf_reclist-rec_type = c_rec_type.
  gf_reclist-express = c_express.
  APPEND gf_reclist TO gt_reclist.
ENDLOOP.
*- Append
* APPEND gf_reclist TO gt_reclist.
*- Clear
* CLEAR gf_reclist.

CALL FUNCTION 'SO_DOCUMENT_SEND_API1'
  EXPORTING
    document_data      = gf_doc_chng
    put_in_outbox      = 'X'
    sender_address     = '49037' " give the User to trigger the Job
    commit_work        = 'X'
  IMPORTING
    sent_to_all        = g_sent_all
  TABLES
    packing_list       = gt_objpack
    object_header      = gt_objhead
    contents_txt       = gt_objtxt
    receivers          = gt_reclist
  EXCEPTIONS
    too_many_receivers = 1
    document_not_sent  = 2
    document_type_not_exist = 3
    operation_no_authorization = 4
    parameter_error    = 5
    x_error            = 6
    enqueue_error      = 7
    OTHERS              = 8.
IF sy-subrc = 0.
  cl_os_transaction_end_notifier=>raise_commit_requested( ).
  CALL FUNCTION 'DB_COMMIT'.
  cl_os_transaction_end_notifier=>raise_commit_finished( ).

```

ENDIF.

*****End of Code *****

Maintained the Mail Ids in an Info Object or DSO for Look Up in Program

Change Characteristic Z_EMAIL: Detail

Characteristic: Z_EMAIL

Long description: email id

Short description: email id

Version: Active Saved

Object Status: Active, executable

Dictionary

Data Element: /BIC/OIZ_EMAIL

Data Type: CHAR - Character String

Length: 60

Lowercase Letters:

Convers. Rout.: ALPHA

Output Length: 60

SID table: /BIC/SZ_EMAIL

Miscellaneous

Attribute Only

Person Respons.:

Content release:

Characteristic is document attrib.

Constant:

Maintaining Mail Ids

/BIC/Z_EMAIL	LAN...	TXTSH
E		
pradeep_neelapu@satyam.com	E	
pavani_potharju@satyam.com	E	

Output mail with Process Chain Status:

From: Rudra Pradeep Reddy Neelapu [mailto:PRADEEP_NEELAPU@SATYAM.COM]
Sent: Monday, December 28, 2009 5:49 PM
To: Pradeep_Neelapu
Subject: Data Loads Completed

Dear Team,

Data Loads are Completed, below are the details.

Process Chain	Date	Start Time	End Time	Actual Load Time	Status
FB4	28.12.2009	13:18:07	13:18:08	00:00:01	F

Regards,
INSIGHT SAP BW TEAM

This is an auto generated Email.

Related Content

Taken most of the inputs from **sap help** in preparing this Article.

<http://forums.sdn.sap.com/thread.jspx?messageID=8747990#8747990>

<http://forums.sdn.sap.com/thread.jspx?messageID=8556918#8556918>

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.