

Making Error DTP Work for DSO's



Applies to:

SAP BI 7..

Summary

Prior to SAP BI7, erroneous request were handled at Info package level by error handling, now it is done at DTP level by creating Error DTP's. Error DTP works differently for CUBE and DSO. For Cube it works directly, but for DSO we have to make customization by carrying out validations at start routine and updating to ERROR stack. This article explains how to implement it based on simple scenario with sample code and results.

Author: Sejal Sonchhatra

Company: IBM India Pvt. Ltd

Created on: 22 October 2009

Author Bio



Sejal is a BI7 Certified solution consultant, having 5 years of experience in BW/BI. She has worked on various end to end BW/BI implementation projects, BI Upgrade projects, Performance tuning projects. Got SAP Training on Business Objects. Sejal is working with IBM India Pvt. Ltd as a BW/BI consultant

Table of Contents

Introduction	3
Scenario	3
How to Implement	4
1. Create an Error DTP at the Update Tab of your normal DTP.	4
2. Created a Transformation from source to target with start routine.	6
Working of Start Routine:.....	10
3. Loaded the source data containing error with date format & having special character.	10
4. Execute the Normal DTP	11
5. Check the DSO Manage screen	11
6. Check the ERROR STACK data	12
7. Correct the ERROR STACK data.....	12
8. Execute ERROR DTP.....	13
9. Activate the Request.....	13
Related Content	14
https://forums.sdn.sap.com/thread.jspa?threadID=1003243	14
Disclaimer and Liability Notice.....	15

Introduction

Error DTP is a concept given in BI7 for Error handling which was done at Info package level in BW3.5.

Creation of Error DTP in Cube is direct as the error DTP works at the time of loading the data, so it does all the checks like invalid characters, lowercase, date plausibility, if no SID exists etc. But in case of DSO this is not direct as all these checks happen at the time of activation of DSO so error DTP doesn't work directly. This Article describes in detail about how to implement ERROR DTP for DSO.

Scenario

Many a times due to incorrect records from R/3, file, the data load (to cube) or activation (in DSO) fails in BW. Because of N number of reasons the data load is erroneous and the data is not available to end user. So there is a need of Error handling similar to BW3.5 info package level where we can process the correct records and block the error records. In BI7 we do it by using a concept of ERROR DTP. This would be an advantage as every time we don't have to load all the records but only the incorrect records needs to be corrected and loaded later.

At Cube level system is smart enough to block the records and save it to ERROR STACK , later we can include the ERROR DTP in process chain or execute it after correcting the records in ERROR STACK . It works same as PSA Correction and loading to target.

But to accomplish it at DSO level we have common step of creating ERROR DTP, including it in process chain, correcting the records of ERROR STACK and load again to target DSO.

The extra things we need to carry is to write a code in Start routine at transformation level carrying out all the possible checks and then push the error records to ERROR STACK.

How to Implement

1. Create an Error DTP at the Update Tab of your normal DTP.

The screenshot shows the configuration interface for a Data Transfer Process (DTP) in SAP. The 'Update' tab is selected. The 'Data Target' is set to 'DataStore Object' with the name 'VP_DE004' and description 'BL-Indiv input data'. The 'Error Handling' section is highlighted with a red circle and contains the following settings:

- Error Handling:** Valid Records Update, Reporting Possible (Request Green)
- Maximum Number of Errors per Pack:** 6
- Displaying Error DTPs:** Checked (indicated by a mouse cursor icon)

At the bottom, the 'Type of Data Update to Data Targets' section has two radio buttons:

- Further Processing Without Master Data
- No Further Processing Without Master Data

- **Error Handling:** Valid Records Update, Reporting Possible (Request Green)
- **Maximum Number of Error per Pack:** You can set your maximum limit.
- If Error DTP is not created then instead of Displaying Error DTP's you would get as Creating Error DTP's

Error DTP looks like below

Data Transfer Process	Error DTP: TEST DTP	
ID	DTP_CAW2VHZG7GTP1L4CUGCW43V4R	
DTP Type	Error DTP	
Version	<input checked="" type="checkbox"/> Active	<input type="checkbox"/> Saved
Extraction Update Execute		
Data Target	DataStore Object	
	VP_DE004	
	BL-Indiv input data	
Error Handling	Valid Records Update, Reporting Possible (Request Green)	
	Maximum Number of Errors per Pack.	1.000
Type of Data Update to Data Targets		
<input checked="" type="radio"/> Further Processing Without Master Data		
<input type="radio"/> No Further Processing Without Master Data		

2. Created a Transformation from source to target with start routine.

The screenshot shows the SAP Data Services Transformation Designer interface. At the top, the 'Start Routine' button is highlighted with a red circle. Below it, the transformation configuration is shown: Source is ZTEST4CRMT (ZTEST4CRMT) and Target is BL-Indiv input data (VP_DE004). The transformation is active and executable. A table on the left lists source fields, and a table on the right lists target fields. Arrows indicate mappings between source and target fields.

Pos	Ke	Field	Descript.
1		/BIC/VP_DEAPPC	Application
2		/BIC/VP_DEORDN	INDIV Ord Number
3		/BIC/VP_DEMSGV	Vers. No.
4		/BIC/VP_DEKPSN	KPS-No
5		/BIC/VP_DEGFY	GF Type
6		/BIC/VP_DEORDD	Order created date

VP_DEORDN	2	VP_DEORDN	INDIV Ord Number
VP_DEMSGV	3	VP_DEMSGV	Vers. No.
VP_DEKPSN	4	VP_DEKPSN	KPS-No
VP_DECREA	5	VP_DECREA	UTC Time Stamp in Short Fo
VP_DECHGD	6	VP_DECHGD	Date on Which Object was
VP_DEORDS	7	VP_DEORDS	Status
VP_DBUID	8	VP_DBUID	Business Transaction ID
VP_DEGFY	9	VP_DEGFY	GF Type
VP_DEGIDP	10	VP_DEGIDP	EG Id P
VP_DEGFBM	11	VP_DEGFBM	BMP Bus Transaction
VP_DEXCAN	12	VP_DEXCAN	Is cancelled
VP_DENOCH	14	VP_DENOCH	No Recovery
VP_DETRFC	15	VP_DETRFC	Trafo case id
VP_DEBSNR	16	VP_DEBSNR	VP_DEBSNR
VP_DEZPID	17	VP_DEZPID	Target Product
VP_DERULE	18	VP_DERULE	Rule ID
VP_DECORD	19	VP_DECORD	CRMT Order No.
VP_DEORDP	20	VP_DEORDP	CRMT Ord Pos No
VP_DEORDD	21	VP_DEORDD	Order created date

Below is the code of Transformation for checking the Date Plausibility and Special Characters and then pushing into Error Stack.

```
PROGRAM trans_routine.
```

```
* -----*
*      CLASS routine DEFINITION
* -----*
*
* -----*
CLASS lcl_transform DEFINITION.
  PUBLIC SECTION.

*  Attributs
  DATA:
    p_check_master_data_exist
      TYPE RSODSOCHECKONLY READ-ONLY,
* - Instance for getting request runtime attributs;
* Available information: Refer to methods of
* interface 'if_rsbk_request_admintab_view'
    p_r_request
      TYPE REF TO if_rsbk_request_admintab_view READ-ONLY.
```

PRIVATE SECTION.

TYPE-POOLS: rsd, rstr.

* Rule specific types

TYPES:

BEGIN OF _ty_s_SC_1,

* Field: /BIC/VP_DEAPPC Application.

/BIC/VP_DEAPPC TYPE C LENGTH 6,

* Field: /BIC/VP_DEORDN INDIV Ord Number.

/BIC/VP_DEORDN TYPE C LENGTH 10,

* Field: /BIC/VP_DEMSGV Vers. No..

/BIC/VP_DEMSGV TYPE N LENGTH 10,

* Field: /BIC/VP_DEKPSN KPS-No.

/BIC/VP_DEKPSN TYPE C LENGTH 18,

* Field: /BIC/VP_DEGFTY GF Type.

/BIC/VP_DEGFTY TYPE C LENGTH 3,

* Field: /BIC/VP_DEORDD Order created date.

/BIC/VP_DEORDD TYPE D,

* Field: RECORD Record Number.

RECORD TYPE RSARECORD,

END OF _ty_s_SC_1.

TYPES:

_ty_t_SC_1 TYPE STANDARD TABLE OF _ty_s_SC_1
WITH NON-UNIQUE DEFAULT KEY.

***\$ begin of global - insert your declaration only below this line *-*

***\$ end of global - insert your declaration only before this line *-*

METHODS

start_routine

IMPORTING

request type rsrequest

datapackid type rsdatapid

EXPORTING

monitor type rstr_ty_t_monitors

CHANGING

SOURCE_PACKAGE type _ty_t_SC_1

RAISING

cx_rsrount_abort.

METHODS

inverse_start_routine

IMPORTING

i_th_fields_outbound TYPE rstran_t_field_inv

i_r_selset_outbound TYPE REF TO cl_rsmds_set

i_is_main_selection TYPE rs_bool

i_r_selset_outbound_complete TYPE REF TO cl_rsmds_set

i_r_universe_inbound TYPE REF TO cl_rsmds_universe

CHANGING

c_th_fields_inbound TYPE rstran_t_field_inv

c_r_selset_inbound TYPE REF TO cl_rsmds_set

c_exact TYPE rs_bool.

ENDCLASS. "routine DEFINITION

***\$ begin of 2nd part global - insert your code only below this line *

```

... "insert your code here

*$$$ end of 2nd part global - insert your code only before this line  *
*-----*
*      CLASS routine IMPLEMENTATION
*-----*
*
*-----*
CLASS lcl_transform IMPLEMENTATION.
*-----*
*      Method start_routine
*-----*
*      Calculation of source package via start routine
*-----*
*      <-> source package
*-----*
METHOD start_routine.
*=== Segments ===

FIELD-SYMBOLS:
  <SOURCE_FIELDS>    TYPE _ty_s_SC_1.

DATA:
  MONITOR_REC        TYPE rstmonitor.

*$$$ begin of routine - insert your code only below this line  *-*
... "insert your code here

* BEGIN of Code for pushing Error entries in error stack.
* IBM
  BREAK-POINT.

DATA : wa_source_package TYPE _ty_s_sc_1.
DATA : t(2) VALUE '#!'.
DATA : flag TYPE c.

BREAK-POINT.
LOOP AT SOURCE_PACKAGE INTO wa_SOURCE_PACKAGE.
  CLEAR monitor_rec.
  CLEAR flag.

* Check for Date plausibility

  CALL FUNCTION 'DATE_CHECK_PLAUSIBILITY'
    EXPORTING
      date = wa_source_package-/bic/vp_deordd

  EXCEPTIONS
    plausibility_check_failed      = 1
.

* Pushing data to ERROR STACK if check failed.

IF sy-subrc <> 0.

```

```

    flag = 1.
    monitor_rec-msgid = 'ZBW'.
    monitor_rec-msgno = 201.
    monitor_rec-msgty = 'E'.
    monitor_rec-recno = wa_source_package-record.
    monitor_rec-skipped = 'X'.
    APPEND monitor_rec TO MONITOR.
    DELETE TABLE SOURCE_PACKAGE FROM wa_SOURCE_PACKAGE.

ENDIF.

** Check for special characters.
    IF wa_source_package-/bic/vp_dekpsn CA t.

** Pushing data to ERROR STACK if contains special character
    monitor_rec-msgid = 'ZBW'.
    monitor_rec-msgty = 'E'.
    monitor_rec-recno = wa_source_package-record.
    monitor_rec-skipped = 'X'.
    IF flag = 1.
        monitor_rec-msgno = 203. "Maintained separate custom messages for
monitoring
    ELSE.
        monitor_rec-msgno = 202. "Maintained separate custom messages for
monitoring
        APPEND monitor_rec TO MONITOR.
        DELETE TABLE SOURCE_PACKAGE FROM wa_SOURCE_PACKAGE.
    ENDIF.

ENDIF.
ENDLOOP.

* END of Code for pushing Error entries in error stack.

*-- fill table "MONITOR" with values of structure "MONITOR_REC"
*- to make monitor entries
... "to cancel the update process
* raise exception type CX_RSR0UT_ABORT.

*$$ end of routine - insert your code only before this line *-*
    ENDMETHOD.                "start_routine
*-----*
*      Method inverse_start_routine
*-----*
*
*      This subroutine needs to be implemented only for direct access
*      (for better performance) and for the Report/Report Interface
*      (drill through).
*      The inverse routine should transform a projection and
*      a selection for the target to a projection and a selection
*      for the source, respectively.
*      If the implementation remains empty all fields are filled and

```

```

*      all values are selected.
*
*-----*
*
*-----*
METHOD inverse_start_routine.

*$$$ begin of inverse routine - insert your code only below this line*-*
... "insert your code here
*$$$ end of inverse routine - insert your code only before this line *-*

      ENDMETHOD.                "inverse_start_routine
ENDCLASS.                       "routine IMPLEMENTATION

```

Working of Start Routine:

Start routine pushes the error records with invalid date or special characters to Error STACK by using monitor_rec which in turn pushes to monitor which by default pushes those error records in work area to ERROR STACK. We have to delete the records pushed to ERROR STACK from source_package.

Note : This example shows validation of Special characters and Date Check , there could be more validation done in real code example for small letters, for null values etc.)

3. Loaded the source data containing error with date format & having special character.

Created a PC_FILE type of data source for test purpose. The Text file is attached below which contains invalid date format and special characters in few records marked as red.

BW	1234	1	KP1#	GF1	20080101
APO	4567	2	KP2	GF2	200800
APO1	768	3	KP3	GF3	20080102
APO2	768	4	KP4	GF4	20080103
APO3	768	5	KP5	GF5	20080
APO4	768	6	KP6	GF6	200803

4. Execute the Normal DTP DTP Request 137.197

Request ID: 137.197
 Start Time: 31.10.2009 14:22:41
 Finish Time: 31.10.2009 14:22:48

Request Processing	M...	D...	Time Stamp	Duration
Process Request			31.10.2009 14:22:42	6 Sec.
Data Package 1 (6 Data Records)			31.10.2009 14:22:44	3 Sec.
Extraction DataSource ZTEST4CRMT : 6 Data Records			31.10.2009 14:22:44	3 Sec.
Filter Out New Records with the Same Key : 6 -> 6 Data			31.10.2009 14:22:47	
RSDS ZTEST4CRMT PC_FILE -> ODSO VP_DE004 : 6			31.10.2009 14:22:47	5 Sec.
Transformation Start			31.10.2009 14:22:47	
Start Routine			31.10.2009 14:22:47	1 Sec.
Rules			31.10.2009 14:22:47	
Transformation End			31.10.2009 14:22:47	
Messages saved for 4 data records; request green			31.10.2009 14:22:47	
Update to DataStore Object VP_DE004 : 2 -> 2 Data R			31.10.2009 14:22:47	
No More Data Available			31.10.2009 14:22:48	

From the above screen shot you can check that 4 records having incorrect data went into ERROR STACK and 2 records got updated to DSO.

5. Check the DSO Manage screen

InfoProvider Administration

Name	D...	Technical Name	Table Type
BL-Indiv input data		VP_DE004	DataStore Object

Requests from DataStore Object:BL-Indiv input data(VP_DE004)

Request ID	R...	D...	ID of Requ...	Re...	Loa...	Log...	DTP/InfoPackage	Request D...	Update Date	Selection ...	Transfere...	Added Rec...	Type of Data
137197			0				TEST DTP (DTP_A65XJ...	31.10.2009	31.10.2009		6	2	Added Records update

6. Check the ERROR STACK data

Request ID	137.197	Error Stack (F9)
Start Time	31.10.2009 14:22:41	
Finish Time	31.10.2009 14:22:48	

Below are the 4 erroneous records.

Error Stack

Status	DataPacket	Data Rec.	Applicatio	INDIV O...	Vers. No.	KPS-No	GF Type	Order crea
	1	1	APO	4567	2	KP2	GF2	.00.2008
	1	4	APO3	768	5	KP5	GF5	.0.2008
	1	5	APO4	768	6	KP6	GF6	.03.2008
	1	6	BW	1234	1	KP1#	GF1	01.01.2008

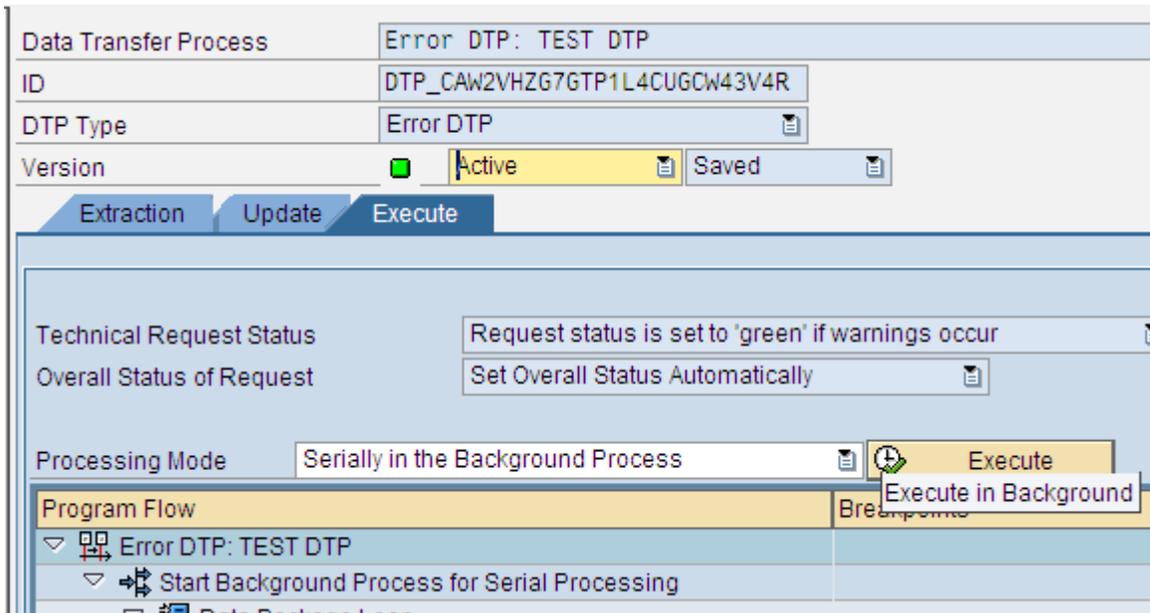
7. Correct the ERROR STACK data.

Error Stack

Status	DataPacket	Data Rec.	Applicatio	INDIV O...	Vers. No.	KPS-No	GF Type	Order crea
	1	1	APO	4567	2	KP2	GF2	06.01.2009
	1	4	APO3	768	5	KP5	GF5	04.09.2009
	1	5	APO4	768	6	KP6	GF6	01.03.2008
	1	6	BW	1234	1	KP1	GF1	01.01.2008

8. Execute ERROR DTP

Load the corrected data from ERROT STACK to Data Target DSO through executing ERROR DTP.



9. Activate the Request.

The two requests are available in manage of DSO one from Normal DTP, other with ERROR DTP.

Selectable Data Targets for Administration

Name	D...	Technical Name	Table Type
BL-Indiv input data		VP_DE004	DataStore Object

Contents Requests Reconstruction

Requests from DataStore Object:BL-Indiv input data(VP_DE004)

Request ID	R...	D...	ID of Requ...	Re...	Loa...	Log...	Re...	Up...	Sele...	Tran...	Adde...	Type of Data U...	Source/InfoSource
137198			137199	○○○			31 ... 31 ...		4	4		Full update	DTP
137197			137199	○○○			31 ... 31 ...		6	2		Full update	DTASRC

Related Content

[Reference 1](#)

<http://forums.sdn.sap.com/thread.jspa?threadID=1323191>

[Reference 2](#)

<http://forums.sdn.sap.com/thread.jspa?threadID=1448130>

[Reference 3](#)

<https://forums.sdn.sap.com/thread.jspa?threadID=1003243>

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.