

How To... Implement a Transformation End Routine

Applicable Releases:

SAP NetWeaver BI 3.x and 7.0

IT Practice:

Business Information Management

IT Scenario:

Enterprise Data Warehouse

Version 1.2

November 2008

© Copyright 2008 SAP AG. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.

Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.

Microsoft, Windows, Outlook, and PowerPoint are registered trademarks of Microsoft Corporation.

IBM, DB2, DB2 Universal Database, OS/2, Parallel Sysplex, MVS/ESA, AIX, S/390, AS/400, OS/390, OS/400, iSeries, pSeries, xSeries, zSeries, z/OS, AFP, Intelligent Miner, WebSphere, Netfinity, Tivoli, Informix, i5/OS, POWER, POWER5, OpenPower and PowerPC are trademarks or registered trademarks of IBM Corporation.

Adobe, the Adobe logo, Acrobat, PostScript, and Reader are either trademarks or registered trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Oracle is a registered trademark of Oracle Corporation.

UNIX, X/Open, OSF/1, and Motif are registered trademarks of the Open Group.

Citrix, ICA, Program Neighborhood, MetaFrame, WinFrame, VideoFrame, and MultiWin are trademarks or registered trademarks of Citrix Systems, Inc.

HTML, XML, XHTML and W3C are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.

Java is a registered trademark of Sun Microsystems, Inc.

JavaScript is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

MaxDB is a trademark of MySQL AB, Sweden.

SAP, R/3, mySAP, mySAP.com, xApps, xApp, SAP NetWeaver, and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only. National product specifications may vary.

These materials are subject to change without notice.

These materials are provided by SAP AG and its affiliated companies ("SAP Group") for informational purposes only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

These materials are provided "as is" without a warranty of any kind, either express or implied, including but not limited to, the implied warranties of merchantability, fitness for a particular purpose, or non-infringement.

SAP shall not be liable for damages of any kind including without limitation direct, special, indirect, or consequential damages that may result from the use of these materials.

SAP does not warrant the accuracy or completeness of the information, text, graphics, links or other items contained within these materials. SAP has no control over the information that you may access through the use of hot links contained in these materials and does not endorse your use of third party web pages nor provide any warranty whatsoever relating to third party web pages.

SAP NetWeaver "How-to" Guides are intended to simplify the product implementation. While specific product features and procedures typically are explained in a practical business context, it is not implied that those features and procedures are the only approach in solving a specific business problem using SAP NetWeaver. Should you wish to receive additional information, clarification or support, please refer to SAP Consulting.

Any software coding and/or code lines / strings ("Code") included in this documentation are only examples and are not intended to be used in a productive system environment. The Code is only intended better explain and visualize the syntax and phrasing rules of certain coding. SAP does not warrant the correctness and completeness of the Code given herein, and SAP shall not be liable for errors or damages caused by the usage of the Code, except if such damages were caused by SAP intentionally or grossly negligent.

Disclaimer

Some components of this product are based on Java™. Any code change in these components may cause unpredictable and severe malfunctions and is therefore expressly prohibited, as is any decompilation of these components.

Any Java™ Source Code delivered with this product is only to be used by SAP's Support Services and may not be modified or altered in any way.

Document History

Document Version	Description
1.20	Addition of SAP Note #1223532 as prerequisite.
1.10	Reference to HowTo paper about disaggregation on upload (Return Table feature).
1.00	First official release of this guide

Typographic Conventions

Type Style	Description
<i>Example Text</i>	Words or characters quoted from the screen. These include field names, screen titles, pushbuttons labels, menu names, menu paths, and menu options. Cross-references to other documentation
Example text	Emphasized words or phrases in body text, graphic titles, and table titles
Example text	File and directory names and their paths, messages, names of variables and parameters, source text, and names of installation, upgrade and database tools.
Example text	User entry texts. These are words or characters that you enter in the system exactly as they appear in the documentation.
< Example text >	Variable user entry. Angle brackets indicate that you replace these words and characters with appropriate entries to make entries in the system.
EXAMPLE TEXT	Keys on the keyboard, for example, F2 or ENTER.

Icons

Icon	Description
	Caution
	Note or Important
	Example
	Recommendation or Tip

Table of Contents

- 1. **Business Scenario**..... 1
- 2. **Background Information**..... 1
- 3. **Step-by-Step Procedure**..... 3
 - 3.1 Prerequisite..... 3
 - 3.2 Create End Routine 6
- 4. **Appendix** 7

1. Business Scenario

Your Source delivers the sales figures for the individual sales organization. However, the sales figures need to be displayed for each employee. This means the incoming data needs to distribute according to the employees belonging to the sales organization. The information which employee belongs to a sales organization is stored in the employee master data.

Important

In scenarios with insertion of new records into the result table (RESULT_TABLE), and you want to use error handling, please incorporate the information included in the SAP note #1223532.

Note

The approach with a routine in the Transformations can also be used as a replacement of the 3.x feature of *Return Tables*. For an example of how to use the *Return Table* feature, please refer to the online documentation or the HowTo guide "How To... Disaggregate on Upload". This can be found in the SDN at SAP NetWeaver Capabilities → SAP How-to Guides → SAP NetWeaver 7.0 How-to Guides → Business Information Management (<https://www.sdn.sap.com/irj/sdn/howtoguides>)

2. Background Information

The DataSource delivers the sales figures amount and quantity for a sales organization, material and month. The following processing steps are required in the End Routine of the Transformation:

- Read, from the master data table the employee IDs that belong to the sales organization
- Read, from master data table the number of employees belong to a sales organization
- Divide the key figures the number of employees that have been read from the master data table
- Create an entry in the return table for each employee

Endroutine

End Routine

Month	Sales Org.	Material	Currency	Amount	Unit	Quantity
200601	OST	0815	EUR	300000	PC	600

Incoming
Change Parameter
of End Routine
<RESULT_FIELDS>

Call read of master data
and number of
employees at runtime

Master data table
of employees

Employee	Sales Org.
25671	OST
25672	OST
25673	OST

The number of entries in the result
table corresponds to the number of
employees read from the table

Month	Sales Org.	Material	Employee	Currency	Amount	Unit	Quantity
200601	OST	0815	25671	EUR	100000	PC	200
200601	OST	0815	25672	EUR	100000	PC	200
200601	OST	0815	25673	EUR	100000	PC	200

Outgoing
Change Parameter
of End Routine
<RESULT_FIELDS>

© SAP AG 2004

THE BEST-RUN BUSINESSES RUN SAP

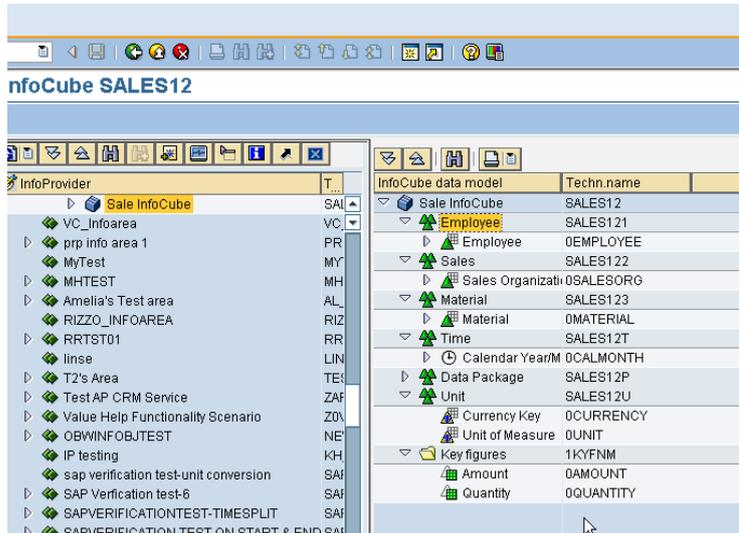


Please have a look at the How-to paper “How To... Routines within Transformations“ to get familiar with the routines concept within Transformations. It can be found in the “SAP Business Intelligence How-To Guides for SAP NetWeaver 7.0” in the SDN (<http://www.sdn.sap.com/irj/sdn/howtoguides>).

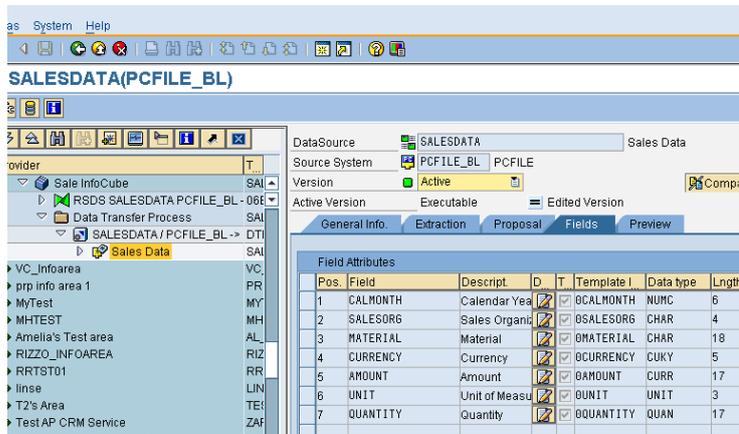
3. Step-by-Step Procedure

3.1 Prerequisite

1. Create InfoCube: SALES12



2. Create DataSource: SALESDATA

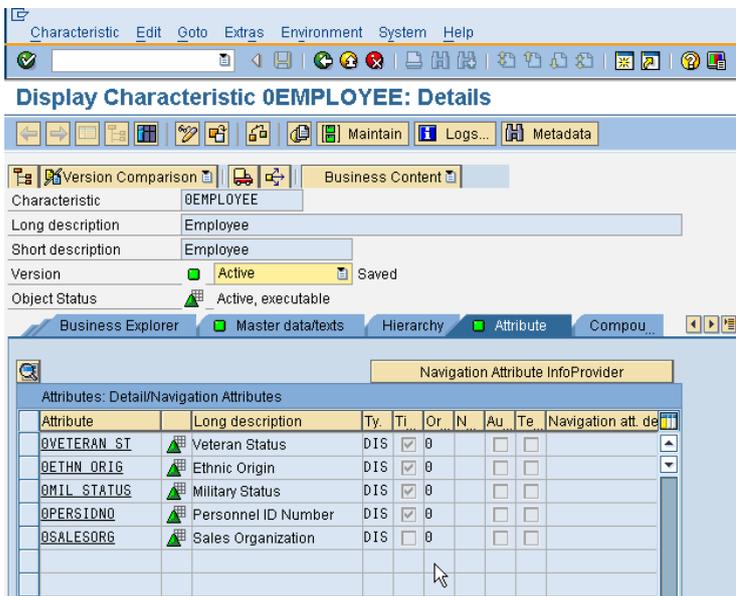


Structure of the DataSource:

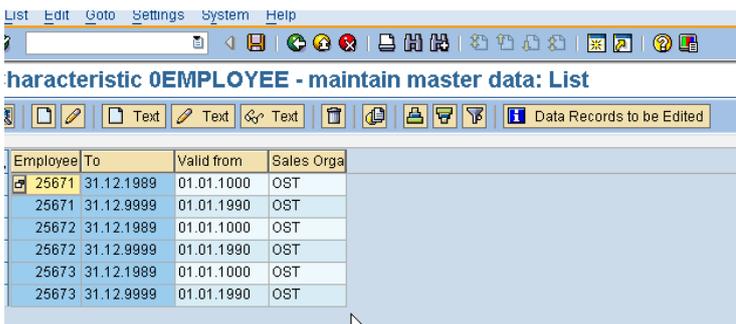
- 0CALMONTH
- 0SALESORG
- 0MATERIAL
- 0CURRENCY
- 0AMOUNT
- 0UNIT
- 0QUANTITY

3. Enhance Employee Master Data (0EMPLOYEE) with Sales Organization (0SALES_ORG)

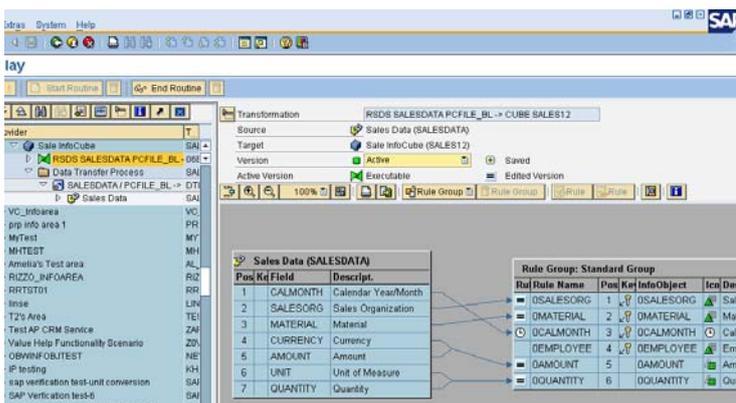
The sales organization is a Attribute of the Employee master data



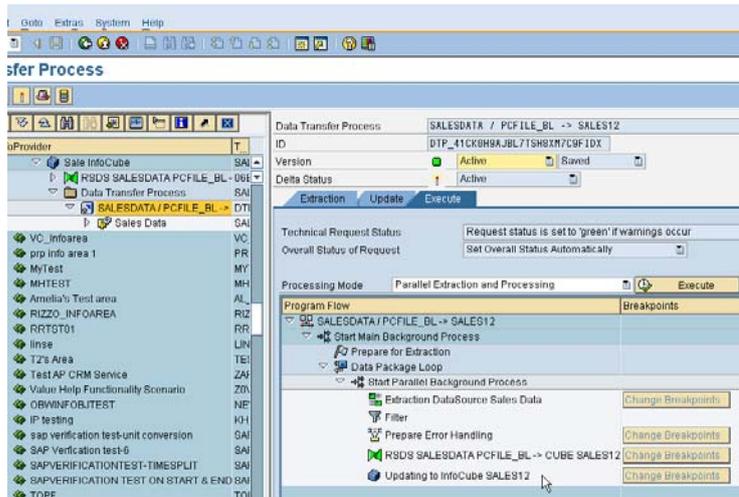
4. Maintain the Sales Organization for the Employees



5. Create Transformation between the DataSource SALESDATA and the InfoCube SALES123

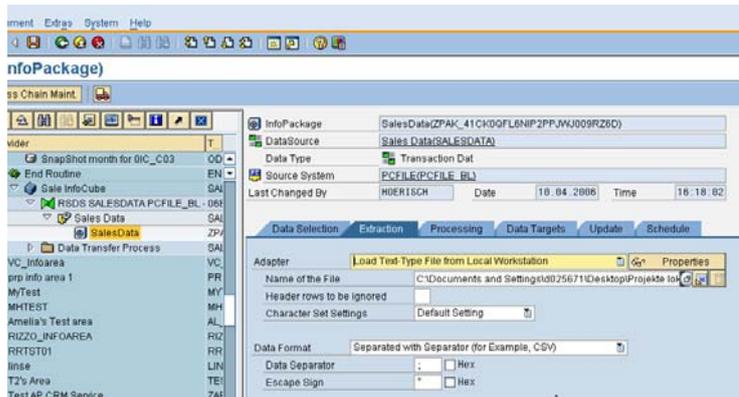


6. Create Data Transfer Process between the DataSource SALESDATA and InfoCube SALES12



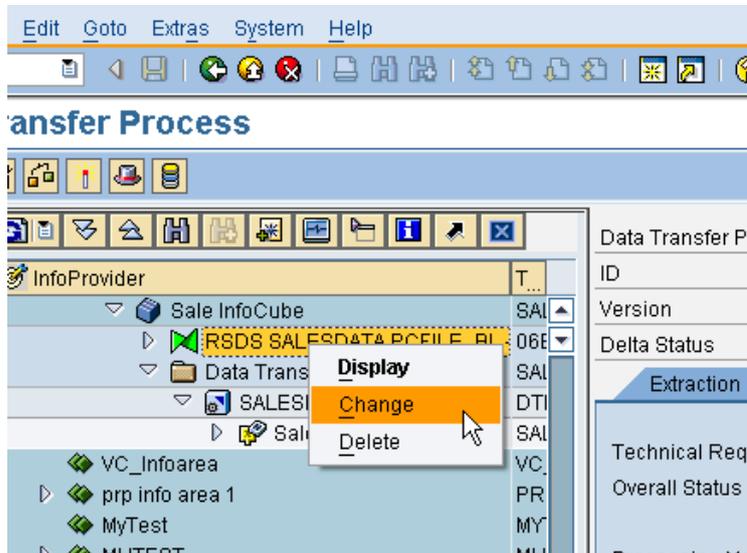
7. Create a Flat File (csv) corresponding to the DataSource structure in 2.

8. Create an InfoPackage corresponding to the files created in 7.

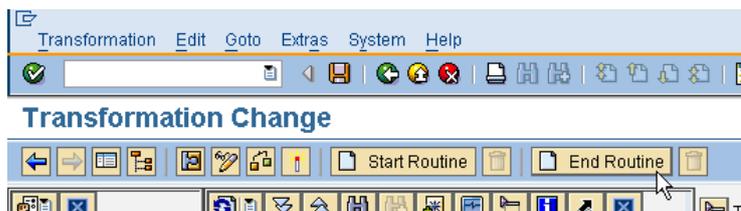


3.2 Create End Routine

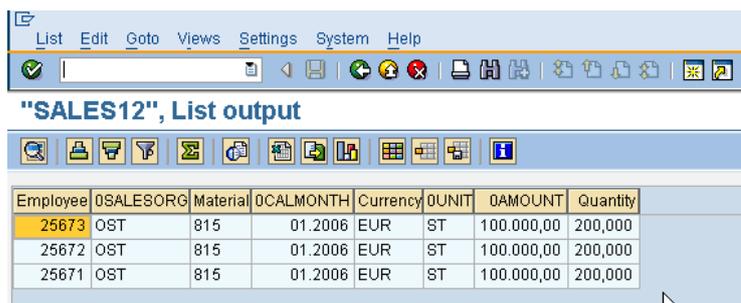
1. Change Transformation between DataSource SALESDATA and InfoCube SALES12



2. Create End Routine



3. Copy the Coding into the End Routine. Please find the coding in the Appendix
4. Result



4. Appendix

Appendix A – Source Coding

In scenarios with insertion of new records into the result table (RESULT_TABLE), be aware of SAP note #1223532 for correct error handling.

Code	Description
End Routine	Sample End Routine in the Transformation

```

PROGRAM trans_routine.

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

  TYPES:
    BEGIN OF _ty_s_TG_1,
*     InfoObject: @SALESORG Sales Organization.
      SALESORG      TYPE /BI0/OISALESORG,
*     InfoObject: @MATERIAL Material.
      MATERIAL      TYPE /BI0/OIMATERIAL,
*     InfoObject: @CALMONTH Calendar Year/Month.
      CALMONTH      TYPE /BI0/OICALMONTH,
*     InfoObject: @EMPLOYEE Employee.
      EMPLOYEE      TYPE /BI0/OIEMPLOYEE,
*     InfoObject: @AMOUNT Amount.
      AMOUNT        TYPE /BI0/OIAMOUNT,
*     InfoObject: @QUANTITY Quantity.
      QUANTITY      TYPE /BI0/OIQUANTITY,
*     InfoObject: @CURRENCY Currency Key.
      CURRENCY      TYPE /BI0/OICURRENCY,
*     InfoObject: @UNIT Unit of Measure.
      UNIT          TYPE /BI0/OIUNIT,
*     Field: RECORD.
      RECORD        TYPE RSARECORD,
    END OF _ty_s_TG_1.
  TYPES:
    _ty_t_TG_1      TYPE STANDARD TABLE OF _ty_s_TG_1
                   WITH NON-UNIQUE DEFAULT KEY.

  PRIVATE SECTION.

  TYPE-POOLS: rsd, rstr.

*$$$ begin of global - insert your declaration only below this line *-*
*****
* Global data declaration
*****
* List of all Employees and corresponding sales organisation
DATA: BEGIN OF I_S_EMPLOYEE,
      SALESORG TYPE /BI0/MEEMPLOYEE-SALESORG,
      EMPLOYEE TYPE /BI0/MEEMPLOYEE-EMPLOYEE,
    END OF I_S_EMPLOYEE.
DATA: i_t_employee like table of I_S_EMPLOYEE.
* List of the all sales organisations and nb. of employees in

```

```

*   the sales organisation
DATA: BEGIN OF I_S_SALESORG_COUNT,
      SALESORG TYPE /BI0/MEMPLOYEE-SALESORG,
      counter TYPE i,
      END OF I_S_SALESORG_COUNT.
DATA: i_t_SALESORG_COUNT like table of I_S_SALESORG_COUNT.

*$$$ end of global - insert your declaration only before this line  *-*
METHODS
  end_routine
  IMPORTING
    request           type rsrequest
    datapackid       type rsdatapid
  EXPORTING
    monitor           type rstr_ty_t_monitors
  CHANGING
    RESULT_PACKAGE   type _ty_t_TG_1
  RAISING
    cx_rsrout_abort.
ENDCLASS.           "routine DEFINITION

* ----- *
*   CLASS routine IMPLEMENTATION
* ----- *
* ----- *
CLASS routine IMPLEMENTATION.

* ----- *
*   Method end_routine
* ----- *
*   Calculation of result package via end routine
* ----- *
*   <-> result package
* ----- *
METHOD end_routine.
*=== Segments ===

FIELD-SYMBOLS:
  <RESULT_FIELDS>   TYPE _ty_s_TG_1.

DATA:
  MONITOR_REC       TYPE rstmonitor.

*$$$ begin of routine - insert your code only below this line  *-*
*****
* local data declaration
*****

data: e_s_result type _ty_s_TG_1.
data: e_t_result type _ty_t_TG_1.

data: amount   like e_s_result-amount.
data: quantity like e_s_result-quantity.

*****
* read master data in local table once
*****
  read table i_t_employee index 1 transporting no fields.

  if sy-subrc = 4.

* get master data for all employees of the RESULT_PACKAGE
  select employee salesorg from /BI0/MEMPLOYEE
    into corresponding fields of table i_t_employee

```

```

        where OBJVERS = 'A'
        and DATETO   >= sy-datum
        and DATEFROM <= sy-datum
        and salesorg <> ''.
    if sy-subrc = 4.
        RAISE EXCEPTION TYPE CX_RSROUT_ABORT.
    endif.

    sort i_t_employee by salesorg.

*   select nb. of employees of a sales organisation
    select salesorg COUNT( * ) as counter
        from /BI0/EMPLOYEE
        into corresponding fields of table i_t_salesorg_count
        where OBJVERS = 'A'
          and DATETO >= sy-datum
          and DATEFROM <= sy-datum
          and salesorg <> ''
        group by salesorg
        order by salesorg.
    if sy-subrc = 4.
        RAISE EXCEPTION TYPE CX_RSROUT_ABORT.
    endif.
endif.

*****
*   Do the calculation and add the new rows
*****

*   loop over the input result data package
loop at RESULT_PACKAGE into e_s_result.
*   get Counter
    read table i_t_salesorg_count into i_s_salesorg_count with key
        salesorg = e_s_result-salesorg.
*   if no counter is available input = output
    if sy-subrc <> 0.
        append e_s_result to e_t_result.
*   add the employees and divide the amount/quantity by counter
    else.
        " calculate quantity and amount
        if i_s_salesorg_count-counter <> 0.
            amount   = e_s_result-amount   /
                i_s_salesorg_count-counter.
            quantity = e_s_result-quantity /
                i_s_salesorg_count-counter.
        else.
            " counter is 0
            RAISE EXCEPTION TYPE CX_RSROUT_ABORT.
        endif.
        loop at i_t_employee into i_s_employee where
            salesorg = e_s_result-salesorg.
            move i_s_employee-employee to e_s_result-employee.
            move amount to e_s_result-amount.
            move quantity to e_s_result-quantity.
*   append the added lines to result
            append e_s_result to e_t_result.
        endloop.
    endif.
endloop.

*   add the lines to the output package
refresh RESULT_PACKAGE.
move e_t_result[] to RESULT_PACKAGE[].

*_*$ end of routine - insert your code only before this line      *_*
    ENDMETHOD.                                                    "end_routine
ENDCLASS.                                                         "routine IMPLEMENTATION

```

www.sdn.sap.com/irj/sdn/howtoguides