

How To... Disaggregate on Upload

Applicable Releases:

SAP NetWeaver BI 7.0 and SAP BW 3.x

IT Practice:

Business Information Management

IT Scenario:

Enterprise Data Warehouse

Version 1.1

September 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 |
|-------------------------|---|
| 1.10 | Conversion / Adaption of Document from 3.x to 7.0 |
| 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**..... 2
 - 3.1 Proceeding for SAP BW 3.x 2
 - 3.1.1 Create update routine with result table 2
 - 3.1.2 Alternative options for the creation of multiple records..... 5
 - 3.1.3 Populate your InfoCube 6
 - 3.2 Proceeding for SAP NetWeaver 7.0..... 7
- 4. **Appendix** 8

1. Business Scenario

You want distribute the costs per cost center to the employees of this cost center. The source system can only provide the cost information for the entire cost center, hence a single record per cost center. In this case, however, you want to analyze the data on employee level, having as many records in the InfoCube as there are employees of the cost center. Therefore, you have to use the cost information for the entire cost center and calculate the cost per employee during the update process to your InfoCube. You also have to create a new record in the InfoCube for every employee of the cost center (distribute the costs).

2. Background Information

The InfoCube contains one record per employee of the cost center, with the distributed costs of the cost center. In general, the data is aggregated while being updated to the InfoCube, e.g. from line item level to lower granular (less detailed) level like customer, customer group, etc. After the implementation of proceeding described in this paper, the data is disaggregated during the update to the InfoCube, from a lower granularity (less detailed), the costs for the entire cost center, to a higher granularity (more detailed), the cost per employee of the cost center.

InfoCube data records

Change run ID 00000000000000
 Request ID REQU_0GXHGSVAKN4MXP1LRL23EYJ75

| Cost center | Employee | Amount | Distributed | Currency |
|-------------|----------|----------|-------------|----------|
| 3105 | 1702 | 1.500,00 | 500,00 | USD |
| 3105 | 1704 | 1.500,00 | 500,00 | USD |
| 3105 | 1263 | 1.500,00 | 500,00 | USD |
| 3200 | 10270 | 2.000,00 | 333,33 | DEM |
| 3200 | 10271 | 2.000,00 | 333,33 | DEM |
| 3200 | 10580 | 2.000,00 | 333,33 | DEM |
| 3200 | 10200 | 2.000,00 | 333,33 | DEM |
| 3200 | 10201 | 2.000,00 | 333,33 | DEM |
| 3200 | 10202 | 2.000,00 | 333,33 | DEM |

Callouts:

- Amount per employee (points to the Amount column)
- Total amount for cost center (points to the Amount column for cost center 3105)
- Employee number (points to the Employee column)

3. Step-by-Step Procedure

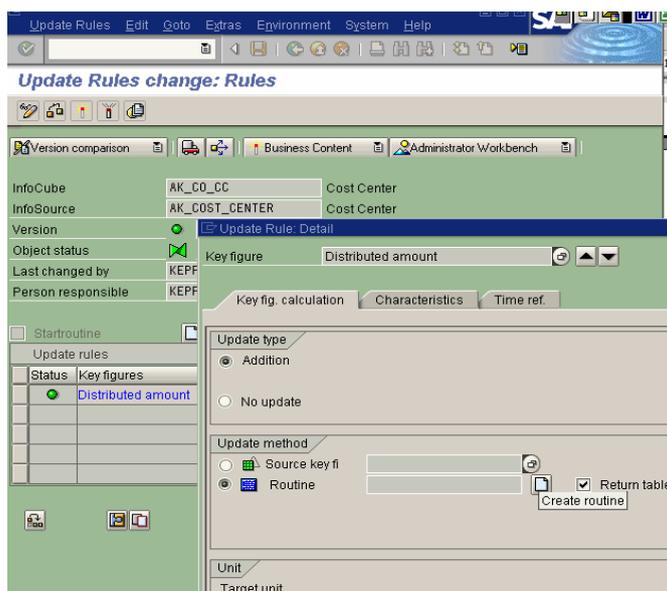
Important

The underneath description is intended for the release SAP BW 3.x. If you want to use the solution for release SAP NetWeaver 7.0, please refer to chapter 3.2.

3.1 Proceeding for SAP BW 3.x

3.1.1 Create update routine with result table

1. Create an update rule for your InfoCube. Within the maintenance of the update rules double-click on the particular key figure which should capture the information, in this example the key figure "Distributed amount".
Mark the radio button "Return table", since you want to return more than one record, and press the push button "Create routine". Enter a descriptive text for the routine.



- Enter the source code (see Appendix for source code example).
Please note that the field names of the sample coding probably differ from the fields in your own InfoCubes and Communication Structures. To determine the fields of an InfoCube, display the structure for the RESULT_TABLE per double-click. For the fields of the respective communication structure, double-click on the structure for the COMM_STRUCTURE.

```

15 TABLES MONITOR STRUCTURE RSMONITOR "user defined monitoring
16 RESULT_TABLE STRUCTURE /BIC/VAK_CO_ACTT
17 USING COMM_STRUCTURE LIKE /BIC/CS0CO_OM_CCA_1
18 RECORD_NO LIKE SY-TABIX
19 RECORD_ALL LIKE SY-TABIX
20 SOURCE_SYSTEM LIKE RSUPDSIMULH-LOGSYS
21 ICUBE_VALUES LIKE /BIC/VAK_CO_ACTT
22 CHANGING RETURNCODE LIKE SY-SUBRC
23 ABORT LIKE SY-SUBRC. "set ABORT <=> 0 to cancel update
24 *
25 *$$$ begin of routine - insert your code only below this line *.*
26 * fill the internal table "MONITOR", to make monitor entries
27
28 * Insert source coding as per Appendix of How To ...
29
30 * if the returncode is not equal zero, the result will not be updated
31 RETURNCODE = 0.
32 * if abort is not equal zero, the update process will be canceled
33 ABORT = 0.
    
```

Note

To improve performance it might be feasible to read the employee information for all cost centers of the data package in a start routine into an internal table. Subsequently, the update routine could read directly from this internal table.

- After checking and saving your routine, also create a routine for every other key figure. Even if you only want to perform a 1:1 mapping of the key figure, you have to populate the characteristic values for the "Employee". When you create this routine for the particular key figure that you want to distribute (in this example "Distributed amount"), the missing derivation of the characteristic values for the "Employee" will lead to the creation of an additional record with the original value.

Update Rules change: Rules

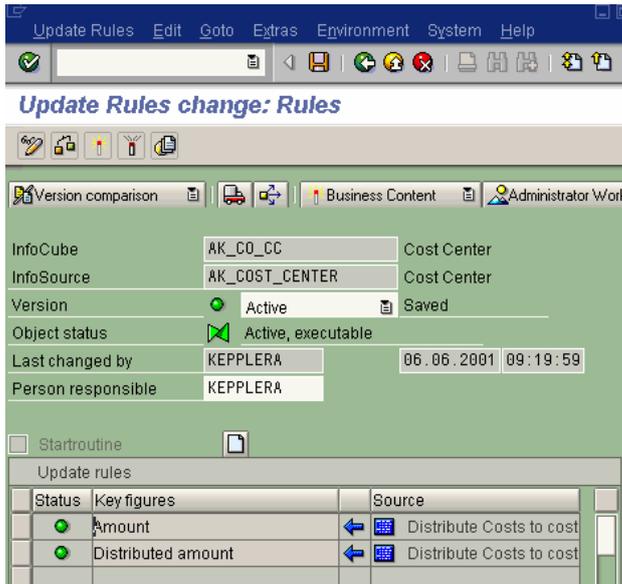
Key figure: Amount

Update type: Addition

Update method: Routine (Distribute Costs to cost ...)

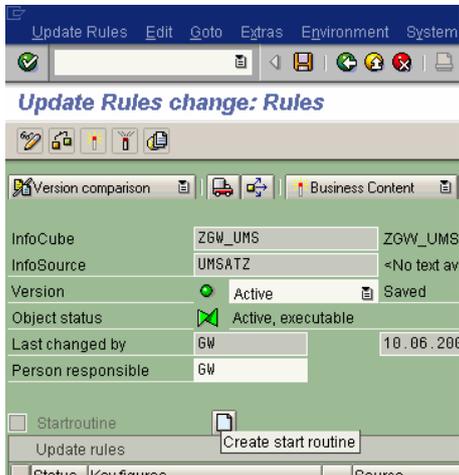
Return table:

4. Activate your update rule

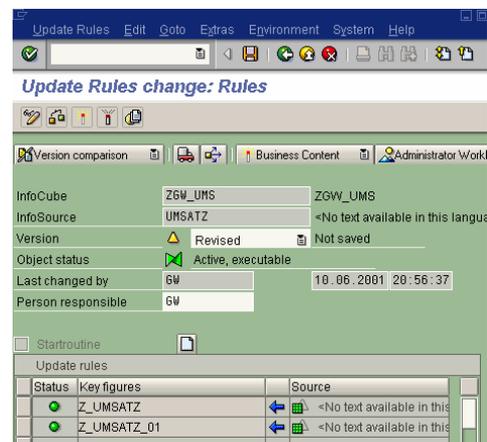
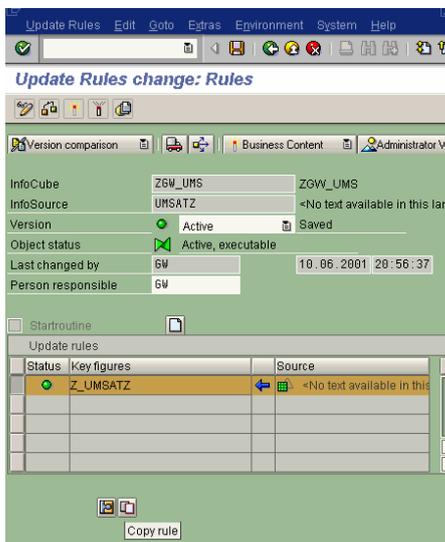


3.1.2 Alternative options for the creation of multiple records

1. Create a start routine where you have access to all data records of a data package. Using this technique, you can modify, delete or add records before the InfoCube update rules are processed.



2. Copy the update rule for a key figure. This method can create additional entries in the InfoCube depending on the derivation of the characteristic values. Yet it is not feasible for the scenario used in this paper, since the number of records to be created cannot be determined beforehand.



3.1.3 Populate your InfoCube

1. Schedule an InfoPackage for the upload of the data from your source system into your InfoCube. Trigger the InfoCube data load from the Persistent Staging Area (PSA), or update the data from an ODS object to the InfoCube.

After a successful data load to the InfoCube, you can check whether the distribution of the cost has been performed correctly.

- a. First look at the data in form of the communication structure, e.g. within the PSA maintenance (PSA → right mouse button → display data)

The screenshot shows the 'Communication structure data records' window in SAP. The table displays data for two cost centers: 3105 and 3200. Cost center 3105 has a total amount of 1,500.00 USD for calendar year 2001. Cost center 3200 has a total amount of 2,000.00 DEM for calendar year 2000. Summary rows are highlighted in yellow.

| Cost center | Calenda... | Amount | Currency |
|-------------|------------|-----------------|------------|
| 3105 | 2001 | 1.500,00 | USD |
| 3200 | 2000 | 2.000,00 | DEM |
| | | 1.500,00 | USD |
| | | 2.000,00 | DEM |

- b. When you compare this data with data in the InfoCube (InfoCube → right mouse button → Manage → Content tab strip), the costs should add up to the same total (except for rounding differences), yet distributed to the different employees of the cost center.

The screenshot shows the 'InfoCube data records' window in SAP. It displays the 'Change run ID' (0000000000000000) and 'Request ID' (REQU_0GXHGSVAKN4MXP1LRL23EYJ75). The table below shows the disaggregation of costs by employee for cost centers 3105 and 3200. Each employee's share is 500.00 USD for 3105 and 333.33 DEM for 3200. Summary rows are highlighted in yellow.

| Change run ID | Request ID | | | |
|------------------|--------------------------------|-----------------|------------|----------|
| 0000000000000000 | REQU_0GXHGSVAKN4MXP1LRL23EYJ75 | | | |
| Cost center | Employee | Amount | Distribute | Currency |
| 3105 | 1702 | 1.500,00 | 500,00 | USD |
| 3105 | 1704 | 1.500,00 | 500,00 | USD |
| 3105 | 1263 | 1.500,00 | 500,00 | USD |
| 3200 | 10270 | 2.000,00 | 333,33 | DEM |
| 3200 | 10271 | 2.000,00 | 333,33 | DEM |
| 3200 | 10580 | 2.000,00 | 333,33 | DEM |
| 3200 | 10200 | 2.000,00 | 333,33 | DEM |
| 3200 | 10201 | 2.000,00 | 333,33 | DEM |
| 3200 | 10202 | 2.000,00 | 333,33 | DEM |
| | | 1.500,00 | USD | |
| | | 1.999,98 | DEM | |

3.2 Proceeding for SAP NetWeaver 7.0

If you want to use the solution also for the release SAP NetWeaver 7.0, the feature of Return Tables is not implemented in the Transformation infrastructure (replacement of Transfer / Update Rules).

In order to achieve the same result, you can either:

1. use the existing routines, namely Start / End or Expert Routine within the Transformation infrastructure. A detailed description of the procedure for a End Routine is described in the existing HowTo paper [Howto Implement a Transformation End Routine](#). 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. or keep your dataflow on 3.x, where you can use update and transfer rules.

4. Appendix

Appendix A – Source Coding

Sample source code for the update routine for the key figure “Distributed amount” (for “Amount” and all other key figures refer to changes in the lines marked with →):

```

PROGRAM update_routine.
*$$$ begin of global - insert your declaration only below this line  *-*
* TABLES: ...

DATA:   l_t_employee TYPE TABLE OF /bi0/memployee,
        l_s_employee LIKE /bi0/memployee.

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

FORM compute_key_figure
    TABLES   monitor STRUCTURE rsmonitor "user defined monitoring
              result_table STRUCTURE /bic/vak_co_cct
    USING     comm_structure LIKE /bic/csak_cost_center
              record_no LIKE sy-tabix
              record_all LIKE sy-tabix
              source_system LIKE rsupdsimulh-logsys
              icube_values LIKE /bic/vak_co_cct
    CHANGING  returncode LIKE sy-subrc
              abort LIKE sy-subrc. "set ABORT <> 0 to cancel update
*
*$$$ begin of routine - insert your code only below this line          *-*
* fill the internal table "MONITOR", to make monitor entries
DATA: l_amount LIKE result_table-amount.

* Get cost center employees
SELECT * INTO TABLE l_t_employee
      FROM /bi0/memployee
      WHERE co_mst_ar   = comm_structure-co_area
      AND   mast_ctr    = comm_structure-costcenter
      AND   datefrom    < sy-datum

```

```
        AND      dateto      > sy-datum
        AND      objvers     = 'A'.

* Distribute over all employees of cost center
l_amount = comm_structure-amount / sy-dbcnt.      "→ Delete
LOOP AT l_t_employee
        INTO l_s_employee.
        MOVE-CORRESPONDING icube_values TO result_table.
        result_table-employee      = l_s_employee-employee.
        result_table-/bic/ak_amnt = l_amount.      "→ Delete
* RESULT_TABLE-amount      = COMM_STRUCTURE-amount. "→ Insert & un-comment
        result_table-currency      = comm_structure-currency.
        APPEND result_table.
ENDLOOP.

* if the returncode is not equal zero, the result will not be updated
returncode = 0.

* if abort is not equal zero, the update process will be canceled
abort = 0.

*$$$ end of routine - insert your code only before this line      *-*
*
ENDFORM.
```

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