

## **SAP First Guidance**

SAP NetWeaver BW 7.30

SAP HANA™ Appliance

# **SAP First Guidance – SAP Netweaver BW 7.30 on HANA Inventory InfoCubes**

## **Applicable Releases:**

### **SAP BW 7.30\* powered by SAP HANA**

This First Guidance Document is the beginning of a series of documents that should help to better understand the various concepts of SAP BW powered by SAP HANA.

The documents are still “work in progress”, so these guides are not intended to be exhaustive so far, and might never be. The purpose of these documents is to deliver additional information besides SAP Help and Blogs to get a better understanding of the concepts of SAP BW on HANA.

For more information or feedback please contact: [Stefan.Hoffmann@sap.com](mailto:Stefan.Hoffmann@sap.com).

**Version 1.6**

**November 2013**



The Best-Run Businesses Run SAP™

© Copyright 2013 SAP AG. All rights reserved.

Business Objects and the Business Objects logo, BusinessObjects, Crystal Reports, Crystal Decisions, Web Intelligence, Xcelsius, and other Business Objects products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of Business Objects Software Ltd. Business Objects is an SAP company.

SAP "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.



The Best-Run Businesses Run SAP™

## Document History





Document Version	Description
1.00	First draft of this guide
1.10	Removed 'Internal Only'
1.20	Removed reference to note 584158
1.30	Added guideline for migration and conversion (notes 1766577, 1780575)
1.31	Changed description of picture 22
1.5	Reworked section extraction from ECC
1.6	Added in section extraction



## 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.
<b>Example text</b>	User entry texts. These are words or characters that you enter in the system exactly as they appear in the documentation.
<b>&lt;Example text&gt;</b>	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.	Introduction .....	2
2.	Scenario .....	2
2.1	Data model Inventory InfoCube .....	3
2.2	Transformation .....	3
2.3	Source table and extractor .....	4
2.4	Database structures and their initial views .....	5
3.	Data Staging .....	6
3.1	Initialization .....	6
3.2	Historical Transactions .....	8
3.3	Delta transactions .....	11
4.	Contents of the fact table .....	13
5.	Differences between HANA based version and non-Hana based version of Inventory data staging .....	13
5.1	Logic of 'No marker update' pushed down .....	13
5.2	No 'E fact table' any longer .....	14
6.	Query .....	14
7.	Influence of InfoCube Compression .....	18
8.	Extraction from SAP ERP .....	21
9.	Considerations regarding conversion of a non In Memory Optimized Inventory InfoCube into a In Memory optimized Inventory InfoCube .....	38
9.1	Single Node Systems .....	38
9.2	Multi Node System .....	39
10.	Conclusions / Recommendations .....	39
11.	Appendix .....	40
11.1	Query logic .....	40
11.2	Query examples .....	42
11.3	Notes .....	44

## 1. Introduction

As the handling of Inventory InfoCubes changed within SAP BW 7.30 based on SAP HANA, this document wants to briefly describe the differences.

In a first section the document describes the data model of the InfoCube. As there is already a How to Guide based on previous release in SDN (<http://scn.sap.com/docs/DOC-15987>), the reader is referred to the 'old' How To Guide, as the majority of the content is still valid.

Furthermore we recommend to read through note 1548125

(<http://service.sap.com/sap/support/notes/1548125>) as this gives explains very well the handling of non-cumulative InfoCubes in SAP BW.

## 2. Scenario

The basis for the explanation in this First Guidance document is an In Memory optimized InfoCube (technical name ZDGRNCUM1), which gets populated by extracting data from a flat table.

In non-cumulative InfoCubes SAP BW on HANA distinguishes between Initialization, Historic and Delta records. To explain you the concepts what happens behind the scenes, we are using a transparent table from which we extract data and run the various DTPs to get transactional data into the In Memory optimized InfoCube.

In the flat table (see Picture 1) we use a field called 'Recordtype' to differentiate the record types. Recordtype = 'I' is used to identify records valid for initialization, Recordtype = 'H' stands for records valid for historic movements, Recordtype = 'D' stands for movements of type delta. This table is the basis for the whole explanation how inventory values are handled within SAP BW on HANA.

**Dictionary: Display Table**

Transp. Table: YNCUMVALUES ☐ Active

Short Description: Non-cumulative values

Attributes | Delivery and Maintenance | Fields | Entry help/check | Currency/Quantity Fields

1 / 6

Field	Key	In...	Data element	Data Type	Length	Deci...	Short Description
RECORDTYPE	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	CHAR1	CHAR	1		0 Single-Character Flag
CALDAY	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	/BIO/OICALDAY	DATS	8		0 Calendar day
MATERIAL	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	/BIO/OIMATERIAL	CHAR	18		0 Material Number
PLANT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	/BIO/OIPLANT	CHAR	4		0 Plant
RECTOTSTCK	<input type="checkbox"/>	<input type="checkbox"/>	DEC8_2	DEC	14		2 Packed integer: 8 bytes, 2 decimal places, leading sign
ISSTOTSTCK	<input type="checkbox"/>	<input type="checkbox"/>	DEC8_2	DEC	14		2 Packed integer: 8 bytes, 2 decimal places, leading sign

Picture 1: Definition of source of Inventory records

**Data Browser: Table YNCUMVALUES** **5 Hits**

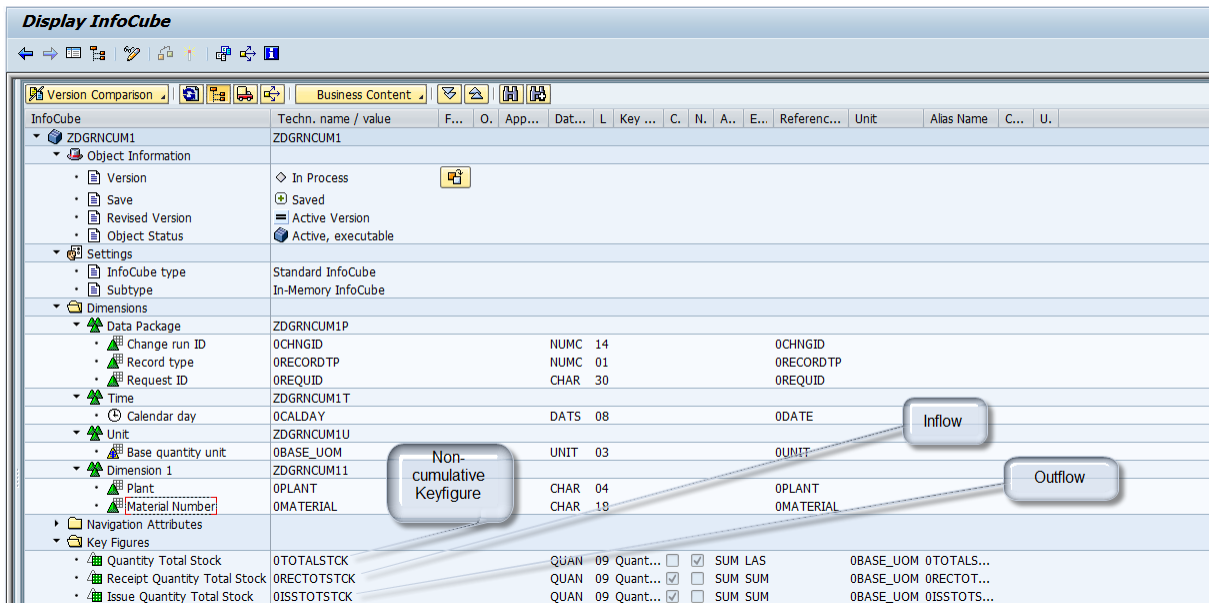
	RECORDTYPE	CALDAY	MATERIAL	PLANT	RECTOTSTCK	ISSTOTSTCK
	D	01.05.2012	A001	S001	75,00	0,00
	D	01.06.2012	A001	S001	25,00	0,00
	H	01.01.2012	A001	S001	100,00	0,00
	H	01.03.2012	A001	S001	50,00	0,00
	I	01.04.2012	A001	S001	150,00	0,00

Picture 2: Table contents of source table

## 2.1 Data model Inventory InfoCube

Picture 3 shows you the data model of the In Memory Optimized InfoCube ZDGRNCUM1. As you can see there is a non-cumulative keyfigure Quantity Total Stock (OTOTALSTCK) which uses exception aggregation 'LAST'.

The value of this keyfigure gets calculated by the sum of Inflows and Outflows. In our example Inflows are stored in the relevant field of Receipt Quantity Total Stock (ORECTOTSTCK), while Outflows are stored in Issue Quantity Total Stock (OISSTOTSTCK). To make the example easy to understand, we have mapped the source fields to the target fields within the transformation without applying complex logic (see Picture 4). In real life, there are in most cases more complex algorithms. In retail for example the inflow and outflow keyfigures are only populated, if certain process keys and movement types are fulfilled.

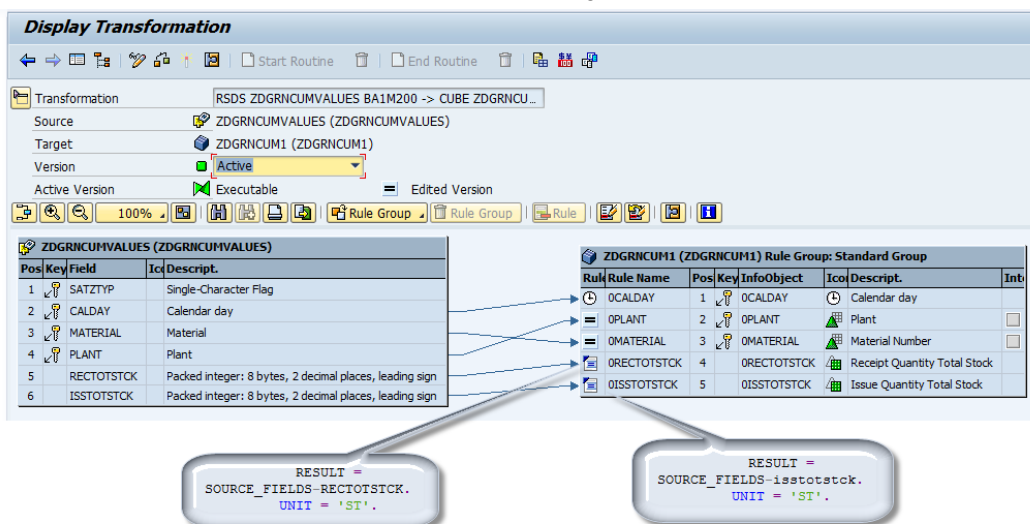


InfoCube	Techn. name / value	F...	O.	App...	Dat...	L	Key ...	C.	N.	A..	E...	Referenc...	Unit	Alias Name	C...	U.
InfoCube	ZDGRNCUM1															
Object Information																
Version	In Process															
Save	Saved															
Revised Version	Active Version															
Object Status	Active, executable															
Settings																
InfoCube type	Standard InfoCube															
Subtype	In-Memory InfoCube															
Dimensions																
Data Package	ZDGRNCUM1P															
Change run ID	0CHNGID					NUMC	14						0CHNGID			
Record type	0RECORDTP					NUMC	01						0RECORDTP			
Request ID	0REQUID					CHAR	30						0REQUID			
Time	ZDGRNCUM1T															
Calendar day	0CALDAY					DATS	08						0DATE			
Unit	ZDGRNCUM1U															
Base quantity unit	0BASE_UOM					UNIT	03						0UNIT			
Dimension 1	ZDGRNCUM11															
Plant	0PLANT					CHAR	04						0PLANT			
Material Number	0MATERIAL					CHAR	18						0MATERIAL			
Navigation Attributes																
Key Figures																
Quantity Total Stock	0TOTALSTCK					QUAN	09	Quant...			<input checked="" type="checkbox"/>	SUM	LAS	0BASE_UOM 0TOTALS...		
Receipt Quantity Total Stock	0RECTOTSTCK					QUAN	09	Quant...			<input checked="" type="checkbox"/>	SUM	SUM	0BASE_UOM 0RECTOT...		
Issue Quantity Total Stock	0ISSTOTSTCK					QUAN	09	Quant...			<input checked="" type="checkbox"/>	SUM	SUM	0BASE_UOM 0ISSTOTS...		

Picture 3: Data model of Inventory InfoCube

## 2.2 Transformation

To get the structure of the In Memory InfoCube populated we mapped the source fields to the InfoCube fields using the transformation seen in Picture 4.



Picture 4: Transformation

## 2.3 Source table and extractor

To enable extraction from our simplified source table, we created a generic extractor and set the flag for stock update in table according ROOSOURCE (see Picture 5).

**Data Browser: Table ROOSOURCE** 1 Hits

OLTPSOURCE	OBJVE...	TYPE	APPL...	BASOSOURCE	DEL...	STOCKUPD
YNCUMVALUES_EXTRACT	A	TRAN	DM			X

Picture 5: Stockupd = ,X' set in table ROOSOURCE

After replication to BW we get the following datasource (see Picture 6).



**Display DataSource YNCUMVALUES\_EXTRACT(BA1M200)**

DataSource: YNCUMVALUES\_EXTRACT Extract YNCUMVALUES

Source System: BA1M200 <-- B59M200

Version: Active Compare with...

Active Version: Executable Edited Version

General Info. Extraction Fields

**General Properties**

Short description: Extract YNCUMVALUES

Medium description: Extract table YNCUMVALUES

Long description: Extract table YNCUMVALUES

Application comp.: DM

Last changed by: HOFFMANNST Changed on: 12.03.2012 / 08:31:32

☐ DS for Data Synchronization ☐ Data Is Language Dependent

☐ PSA in CHAR Format ☐ Data Is Time Dependent

☒ Opening balance

Delivery of Duplicate Data Recs.: Undefined

**Content Properties**

Content Release Type: R/3 Content Version: 730

Picture 6: Datasource

## 2.4 Database structures and their initial views

Before we now start to populate the In Memory optimized InfoCube with data let us show you how the database structures look like after we have activated the InfoCube.

Within the package dimension (see Picture 7 ) we find as supposed the initial entry.

**Data Browser: Table /BIC/DZDGRNCUM1P 1 Hits**

DIM...	SID_0CHNG...	SID_0RECORDT...	SID_0REQU...
0	0	0	0

Picture 7: Package dimension (initial view)

As the InfoCube is partitioned, let us have a look to the partition specification by opening a HANA studio view (see Picture 8).

## First Guidance SAP BW 7.30 on HANA – Inventory InfoCubes

Table Name:	Schema:	Type:
/BIC/FZDGRNCUM1	SAPOH0	Column Store

Columns | Indexes | Further Properties | Runtime Information

Temporary table type: NONE

Partition Specification: ROUNDROBIN 1; RANGE KEY\_ZDGRNCUM1P 0,1,2,\*

Locations:

- hdbpoc00:30003
- hdbpoc00:30003
- hdbpoc00:30003
- hdbpoc00:30003

Picture 8: Partition Spec (HANA Studio)

As you can see by choosing Runtime Information, our InfoCube has 4 partitions.

Table Name:	Schema:
/BIC/FZDGRNCUM1	SAPOH0

Columns | Indexes | Further Properties | Runtime Information

**General**

Table Size: 230 KB      Size (Main): 22 KB  
Records: 0      Size (Delta): 207 KB

**Details for Table**

Parts | Columns

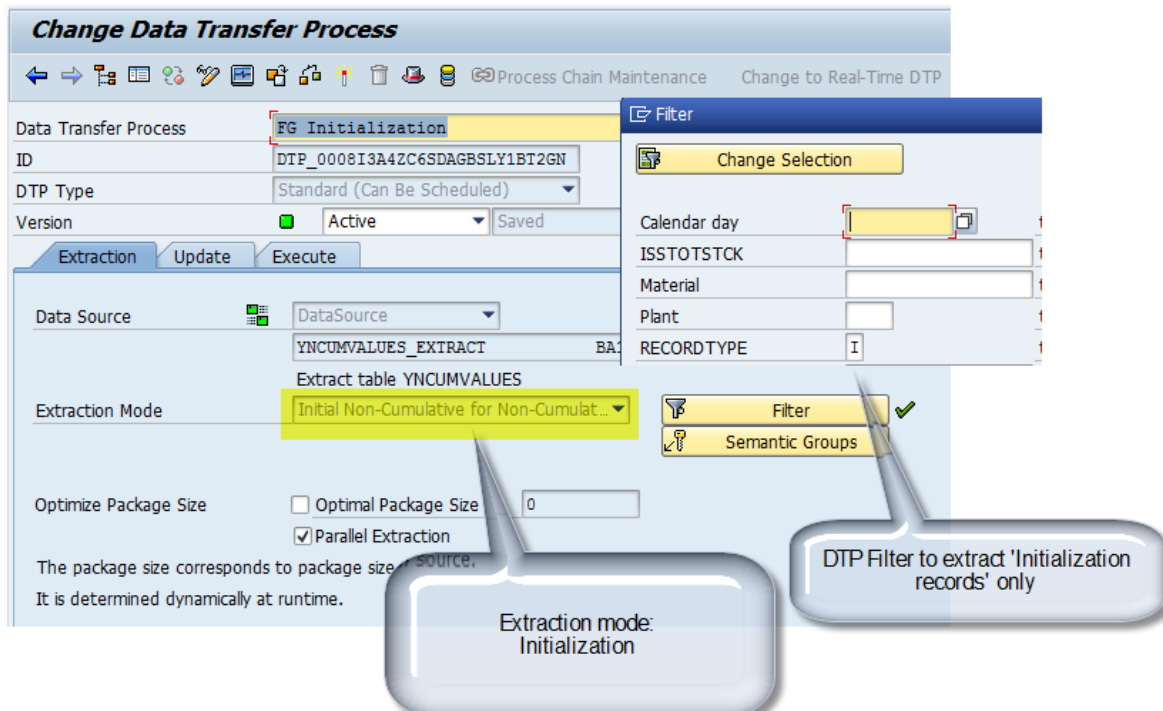
Part ID	Host	Port	Total Size [KB]	Main Size [KB]	Delta Size [KB]	Records
1	hdbpoc00	30003	58	6	52	0
2	hdbpoc00	30003	58	6	52	0
3	hdbpoc00	30003	58	6	52	0
4	hdbpoc00	30003	58	6	52	0

Picture 9: Runtime Information before data staging

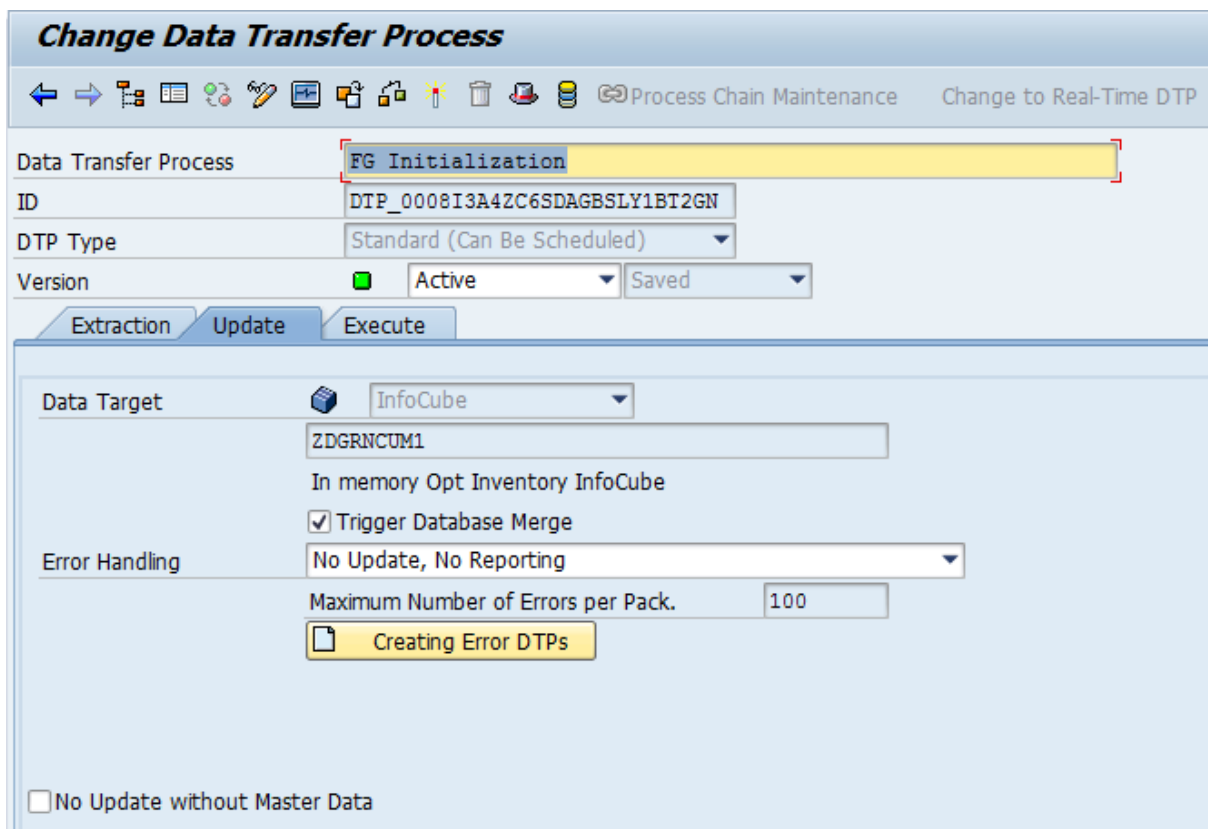
## 3. Data Staging

### 3.1 Initialization

To initialize the In Memory Optimized InfoCube, we use a DTP (see Picture 10 and Picture 11), which is using extraction mode Initialization. In our underlying source table we can filter which records are to be extracted just for the Initialization. That's why we also use a filter in our DTP. Be aware, this is just for simplification reasons (just for better explanation). In case you extract from ERP, please consider the relevant chapter 'Extraction from SAP ERP'.



Picture 10: DTP Initialization (Extraction Tab)



Picture 11: DTP Initialization (Update tab)

If we now schedule the DTP and look what happens behind the scenes we get the following pictures.

As you can see by having a look to the package dimension of our InfoCube (see Picture 12), by execution of the DTP we created an entry DIMID = '20', which correlates to field SID\_ORECORDTP = '1'. The comparison between Picture 12 and Picture 7 shows also that additional entries had been created (DIMID = '1', '2') in the package dimension.

**Data Browser: Table /BIC/DZDGRNCUM1P 4 Hits**

DIMI...	SID_OCHNGID	SID_ORECORDTP	SID_OREQUID
0	0	0	0
1	0	1	0
2	0	2	0
20	0	1	3.620.664

Picture 12: Package dimension after execution of Initialization DTP

You might ask, why that?

Well, that's a basic concept of the non-cumulative handling within SAP BW on HANA. The non-cumulative InfoCubes differentiate between Initialization records (stored with SID\_ORECORDTP = '1'), Historic Movements (stored with SID\_ORECORDTP = '2') and Deltas (stored with SID\_ORECORDTP = '0') by choosing the corresponding DIMIDs to those identifiers.

Let's have a look what happened on HDB level by choosing runtime information view of our fact table (see Picture 13).

Table Name: /BIC/FZDGRNCUM1 Schema: SAP0H0

Columns | Indexes | Further Properties | **Runtime Information**

General

Table Size: 234 KB Size (Main): 22 KB  
Records: 1 Size (Delta): 211 KB

#### Details for Table

Part ID	Host	Port	Total Size [KB]	Main Size [KB]	Delta Size [KB]	Records
1	hdbpoc00	30003	58	6	52	0
2	hdbpoc00	30003	58	6	52	0
3	hdbpoc00	30003	58	6	52	0
4	hdbpoc00	30003	61	6	56	1

Picture 13: Runtime Information after execution of Initialization DTP

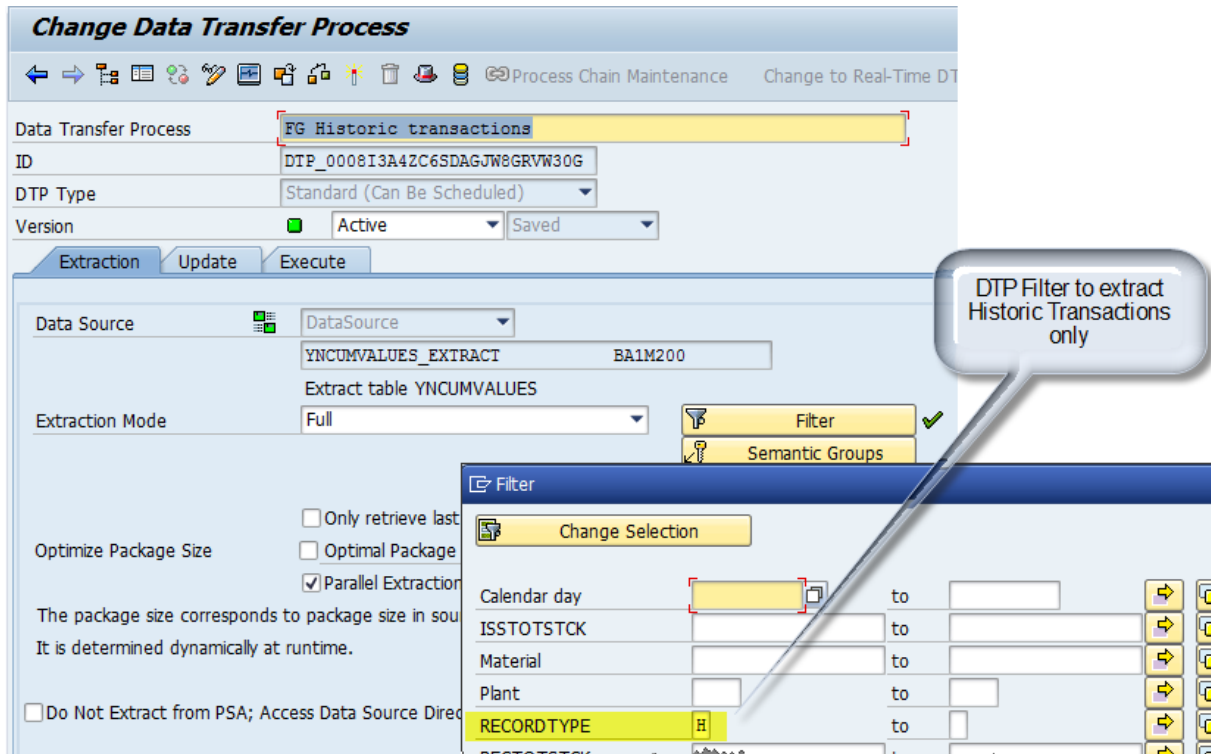
As you can see our initialization record of table YNCUMVALUES (Picture 2) was added to the '\*' Partition (Part ID 4).

## 3.2 Historical Transactions

To transfer historical movement records to the In Memory Optimized InfoCube, we use a second DTP (see Picture 14). As you can see in Picture 14 we filter again only those records which are historical records (see Picture 2). By setting the flag for Historical Transactions in the DTP (see Picture 15), we ensure that SAP BW understands that the records we transfer are only Historical

## First Guidance SAP BW 7.30 on HANA – Inventory InfoCubes

Transactions. That's the major difference to the handling in SAP BW releases on traditional databases, as you there have to set flags whether you want to update a marker or not.



**Change Data Transfer Process**

Data Transfer Process: **FG Historic transactions**

ID: **DTP\_0008I3A4ZC6SDAGJW8GRVW30G**

DTP Type: **Standard (Can Be Scheduled)**

Version: **Active** | **Saved**

**Extraction** | Update | Execute

Data Source: **DataSource**  
**YNCUMVALUES\_EXTRACT** | **BA1M200**  
 Extract table **YNCUMVALUES**

Extraction Mode: **Full** | Filter | Semantic Groups

Optimize Package Size:  
☐ Only retrieve last  
☐ Optimal Package  
☒ Parallel Extraction

The package size corresponds to package size in source system.  
 It is determined dynamically at runtime.

☐ Do Not Extract from PSA; Access Data Source Directly

**Filter**

Change Selection

Field	From	To	Operator
Calendar day			
ISSTOTSTCK			
Material			
Plant			
<b>RECORDTYPE</b>			

DTP Filter to extract Historic Transactions only

Picture 14: DTP Historical Transactions (Extraction Tab)

**Change Data Transfer Process**

Process Chain Maintenance    Change to Real-Time

Data Transfer Process: **FG Historic transactions**

ID: DTP\_0008I3A4ZC6SDAGJW8GRVW30G

DTP Type: Standard (Can Be Scheduled)

Version: ☒ Active    Saved

Extraction    Update    Execute

Data Target: InfoCube  
ZDGRNCUM1  
In memory Opt Inventory InfoCube

☒ Trigger Database Merge

Error Handling: No Update, No Reporting

Maximum Number of Errors per Pack: 100

☐ Creating Error DTPs

☐ No Update without Master Data

☒ Historical Transactions

Extraction mode: Historical Transactions

Picture 15: DTP Historical Transactions (Update tab)

Again, when executing the DTP we transfer the 2 historical transactions from the source table to the InfoCube (see Error! Reference source not found.)

As you can see by having a look to the package dimension of our InfoCube (see Picture 16), by execution of the DTP we created an entry DIMID = '21', which correlates to field SID\_ORECORDTP = '2'.

**Data Browser: Table /BIC/DZDGRNCUM1P**    5 Hits

DIM...	SID_OCHNGID	SID_ORECORDTP	SID_OREQUID
0	0	0	0
1	0	1	0
2	0	2	0
20	0	1	3.620.664
21	0	2	3.620.665

Picture 16: Package dimension after execution of Historical Transactions DTP

Let's have a look what happened on HDB level by choosing runtime information view of our fact table (see Picture 17).

Table Name:

/BIC/FZDGRNCUM1

Schema:

SAPH0

Columns

Indexes

Further Properties

Runtime Information

General

Table Size: 234 KB

Size (Main): 22 KB

Records: 3

Size (Delta): 211 KB

Details for Table

Parts

Columns

Part ID	Host	Port	Total Size [KB]	Main Size [KB]	Delta Size [KB]	Records
1	hdbpoc00	30003	58	6	52	0
2	hdbpoc00	30003	58	6	52	0
3	hdbpoc00	30003	58	6	52	0
4	hdbpoc00	30003	62	6	56	3

Picture 17: Runtime information after execution of Historical Transactions DTP

As you can see our historical records of table YNCUMVALUES (Picture 2) had been added again to the '\*' Partition (Part ID 4).

### 3.3 Delta transactions

Again, we have created a DTP which can be used to extract delta records (see Picture 18) which extracts only those values we have marked as delta records.

**Display Data Transfer Process**

Process Chain Maintenance    Change to Real-Time DTP

Data Transfer Process: FG Delta transactions

ID: DTP\_0008I3A4ZC6SDAGJXPCJBALHE

DTP Type: Standard (Can Be Scheduled)

Version: Active    Saved

Extraction    Update    Execute

Data Source: DataSource

YNOCUMVALUES\_EXTRACT    BA1M200

Extract table YNCUMVALUES

Extraction Mode: Full

Filter    Semantic Groups

Optimize Package Size

☐ Only retrieve last request

☐ Optimal Package Size

☒ Parallel Extraction

The package size corresponds to package size in source.  
It is determined dynamically at runtime.

☐ Do Not Extract from PSA; Access Data Source Directly (for S...

**Filter**

Change Selection

Calendar day    to

ISSTOTSTCK    0,00    to

Material    to

Plant    to

**RECORDTYPE**    D    to

RECTOTSTCK    0,00    to

Extract only those records which are marked as delta records

Picture 18: DTP Delta transactions (Extraction tab)

When executing the DTP we transfer the 2 delta transactions from the source table to the InfoCube. As you can see by having a look to the package dimension of our InfoCube (see Picture 19), by execution of the DTP we created an entry DIMID = '22', which correlates to field SID\_ORECORDTP = '0'.

**Data Browser: Table /BIC/DZDGRNCUM1P**    6 Hits

DIM...	SID_OCHNGID	SID_ORECORDTP	SID_OREQUID
0	0	0	0
1	0	1	0
2	0	2	0
20	0	1	3.620.664
21	0	2	3.620.665
22	0	0	3.620.666

Picture 19: Package dimension after execution of Delta DTP

Let's have a look what happened on HDB level by choosing runtime information view of our fact table (see Picture 20).



Table Name:

/BIC/FZDGRNCUM1

Schema:

SAPH0

Columns

Indexes

Further Properties

Runtime Information

General

Table Size: 236 KB

Size (Main): 22 KB

Records: 5

Size (Delta): 213 KB

Details for Table

Parts

Columns


Part ID	Host	Port	Total Size [KB]	Main Size [KB]	Delta Size [KB]	Records
1	hdbpoc00	30003	58	6	52	0
2	hdbpoc00	30003	58	6	52	0
3	hdbpoc00	30003	58	6	52	0
4	hdbpoc00	30003	64	6	58	5

Picture 20: Runtime information after execution of Delta DTP

As you can see our delta records of table YNCUMVALUES (Picture 2) had been added again to the '\*' Partition (Part ID 4).

## 4. Contents of the fact table

Well, after now transferring all our data records from the source through usage of the various DTPs, how does now our fact table look like?

Data Browser: Table /BIC/FZDGRNCUM1							5 Hits
							
KEY_ZDGRNCUM1P	SID_0CALDAY	SID_0BASE_UOM	SID_0PLANT	SID_0MATERIAL	RECTOTSTCK	ISSTOTSTCK	
20	99.991.231	1.000.000.267	23.029	10.222.270	150,000	0,000	
21	20.120.101	1.000.000.267	23.029	10.222.270	100,000	0,000	
21	20.120.301	1.000.000.267	23.029	10.222.270	50,000	0,000	
22	20.120.501	1.000.000.267	23.029	10.222.270	75,000	0,000	
22	20.120.601	1.000.000.267	23.029	10.222.270	25,000	0,000	

Picture 21: Contents of the fact table (Fact table view) prior to compression

## 5. Differences between HANA based version and non-Hana based version of Inventory data staging

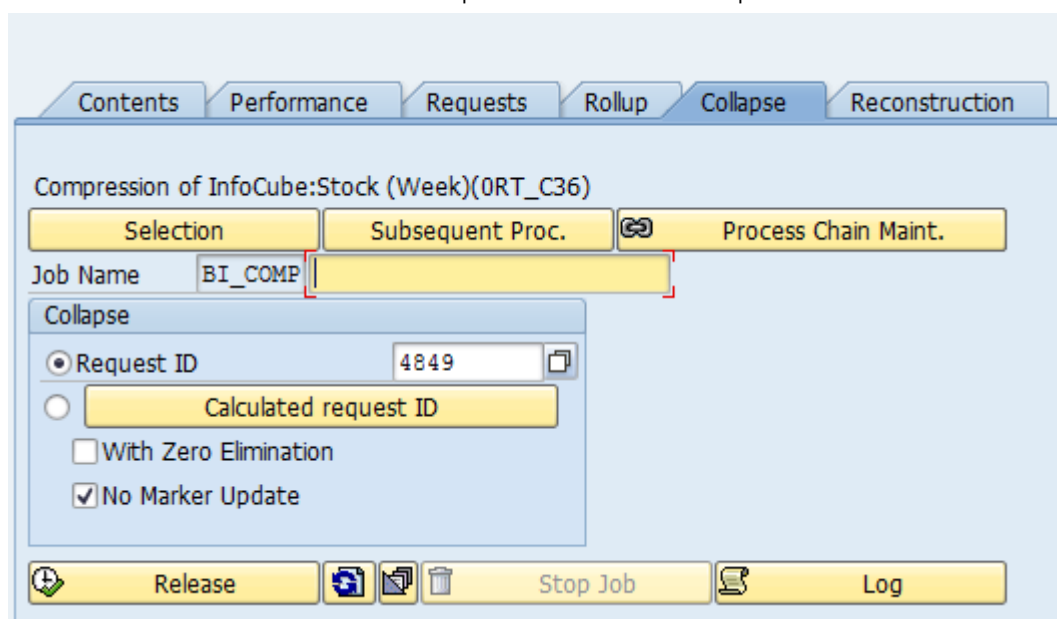
### 5.1 Logic of 'No marker update' pushed down

Instead of setting the flag for Marker update during the compression of an InfoCube you now have to decide within the DTP whether the records are treated to be historical transactions or deltas. That means the logic of the marker update had been pushed down to the data staging DTPs.

The storage of the records has been changed.

Historical transactions are stored in an own partition on database level which can be identified by SID\_ORECORDTP = 2. In the non-HANA based handling all historical transactions had been assigned to SID\_ORECORDTP = 0. These historical transactions had to be compressed with the flag set for 'No marker update' (see Picture 22).

Compression of the InfoCube does not change the value of the Initialization any longer. The concept of the previous known reference point in infinity changed. Therefore the keyfigures of the Initialization request remain unchanged during compression. This was completely different before, as all the delta movements led to an update of the reference point.



Picture 22: Flag for No Marker Update in non HANA based BW versions

## 5.2 No 'E fact table' any longer

In contrast to the non-HANA based Inventory InfoCube, the HANA based Inventory InfoCube has only one F fact table. The fact table itself is partitioned having in total 4 partitions. The purpose of the single partitions is summarized in Table 1: Partition purpose.

Partition	Purpose
1	Compressed deltas
2	Initialization records
3	Historic transactions
4	Uncompressed requests

Table 1: Partition purpose

For details see Picture 41 and Picture 42.

### Recommendation

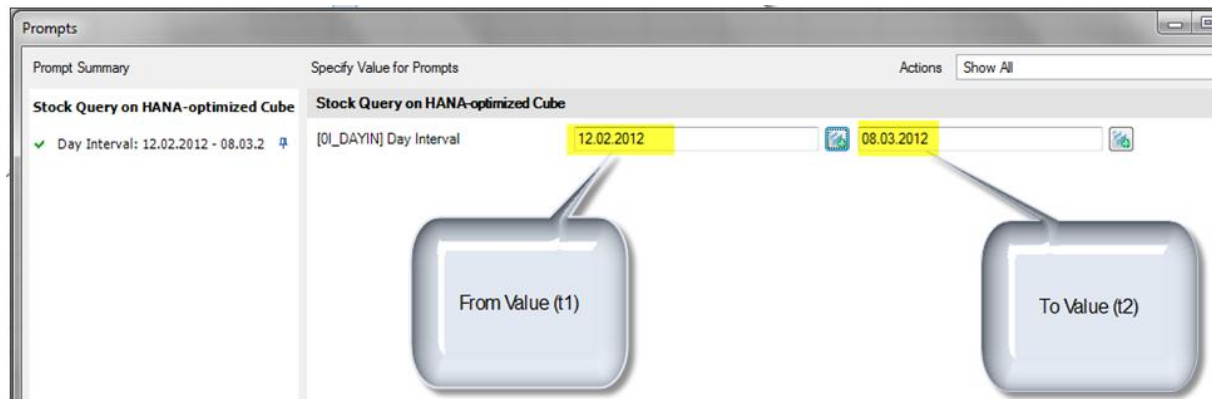
Collapse the Inventory InfoCube regularly to populate the relevant partitions.

## 6. Query

To make explanation of the query behavior of a query containing non-cumulative keyfigures easier, we decided to use a simple query. As you can see in Picture 25 the query has the option to select a range of days and displays the Product, Store and Day in the rows. As keyfigure to be displayed we have selected Total Stock Quantity.

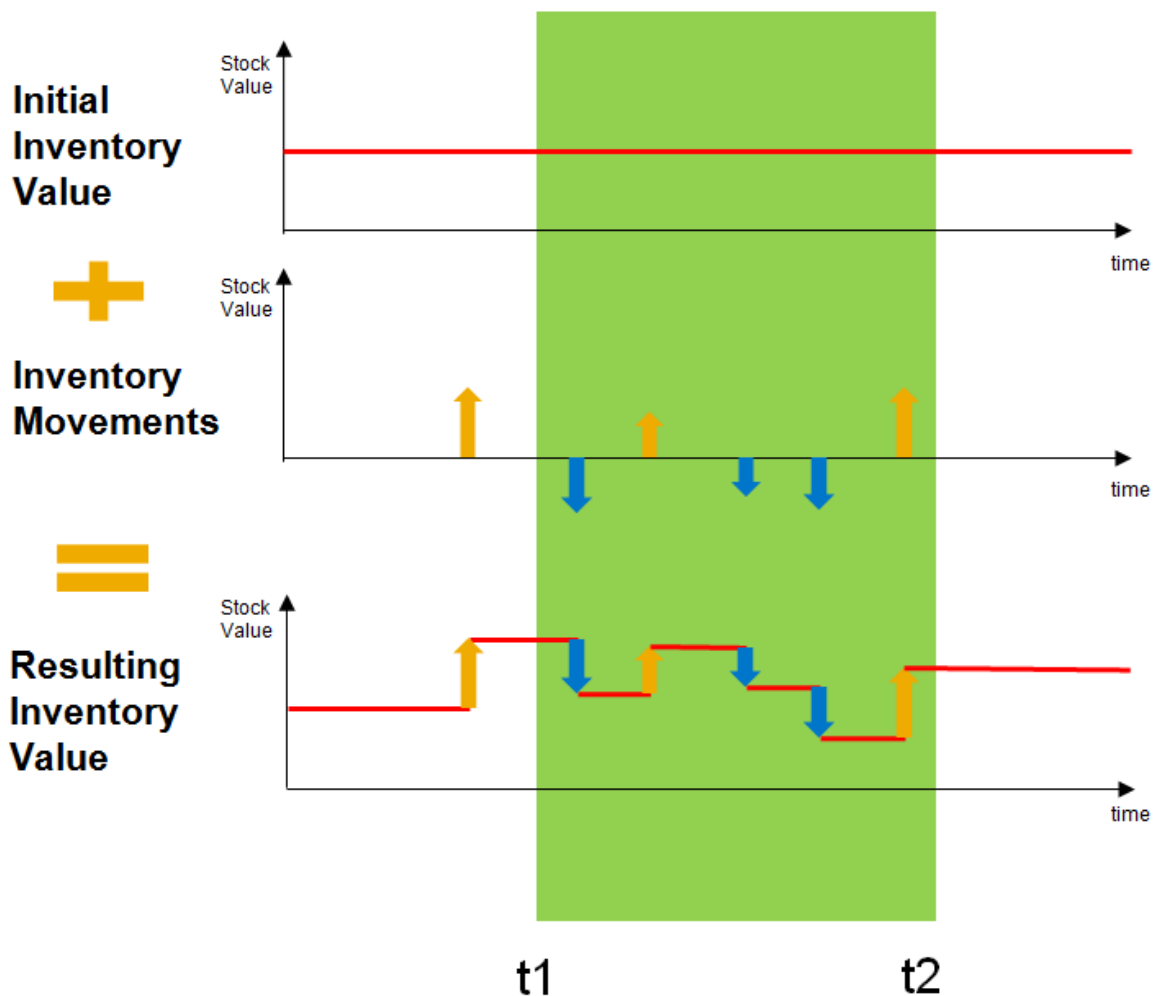
Technically the system always calculates the stock by running 2 queries.

Let us introduce a few terms so we can explain the handling of a non-cumulative query better.



Picture 23: Selection range t1, t2

Within the query below we have the possibility to select a range of days. The upper limit of the selection should be t2 (in the case above equal to 08.03.2012), the lower limit t1 (in our example 12.02.2012) (see Picture 23).



Picture 24: Non-cumulative handling in SAP BW on HANA

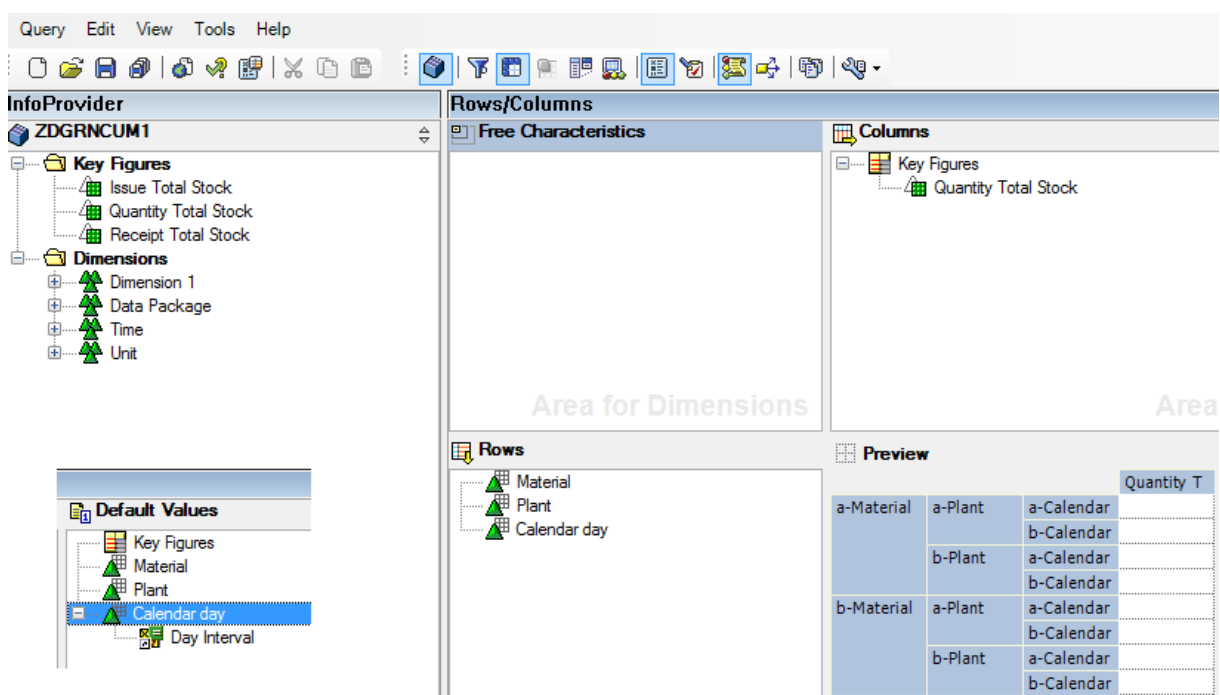
Non-cumulative handling (simplified expressed):

## First Guidance SAP BW 7.30 on HANA – Inventory InfoCubes

In a first query the system has to determine the offset (Initial Inventory Value) for the whole calculation. This offset is always the stock value of the upper time limit of the query time selection (here t2).

This offset is determined by selecting three different terms. Term A is the initialization record (Orecordtp = '1'). The query also has to select all historic transactions larger than t2 (Orecordtp = '2'), as these have influence on the height of the offset. Let's call this selection B (=all historic transactions larger than t2). Finally the system also needs to know all deltas smaller or equal than t2 (Orecordtp = '0'). This should be our selection C. By now aggregating this information and returning the values of A, B and C to the OLAP engine, the OLAP engine calculates the offset as  $A - B + C$ .

In a second query the system then collects all deltas between t2 and t1. By considering whether the delta was an in- or outflow the OLAP can now calculate any stock value in time between t1 and t2 (see Appendix Picture 70).



Picture 25: Query definition

To understand the output of our query, let us allow a small repetition what was stored in our source table.

Data Browser: Table YNCUMVALUES 5 Hits						
RECORDTYPE	CALDAY	MATERIAL	PLANT	RECTOTSTCK	ISSTOTSTCK	
D	01.05.2012	A001	S001	75,00	0,00	
D	01.06.2012	A001	S001	25,00	0,00	
H	01.01.2012	A001	S001	100,00	0,00	
H	01.03.2012	A001	S001	50,00	0,00	
I	01.04.2012	A001	S001	150,00	0,00	

Picture 26: Table contents of source table

## First Guidance SAP BW 7.30 on HANA – Inventory InfoCubes

If we now run our query on the Inventory InfoCube and select Day equal to the 01.04.2012 we get the following output (t2 = 01.04.2012, t1 = 01.04.2012), which is exactly our Initialization record. In the following screen shots you see the output of Advanced Analysis Office. To make the display easier, we used the selection of specific members for OCALDAY in the filter of the Advanced Analysis output (especially in Picture 30).

	A	B	C	D
1				Quantity Total Stock
2	<b>Material</b>	<b>Plant</b>	<b>Calendar day</b>	PCE
3	A001	S001	01.04.2012	150

Picture 27: Query t2=01.04.2012

In a second query we filter now day = 29.02.2012. As there was a inflow on March 1<sup>st</sup> we now get Total Stock Qty = Init – Inflow of March 1<sup>st</sup>.

	A	B	C	D
1				Quantity Total Stock
2	<b>Material</b>	<b>Plant</b>	<b>Calendar day</b>	PCE
3	A001	S001	29.02.2012	100

Picture 28: Query t2=29.02.2012

In the third query we filter day to 02.05.2012. Remember, there was a inflow on Mai 1<sup>st</sup>. Therefore Total Stock Qty is equal to Init + Inflow of May 1<sup>st</sup>.

	A	B	C	D
1				Quantity Total Stock
2	<b>Material</b>	<b>Plant</b>	<b>Calendar day</b>	PCE
3	A001	S001	02.05.2012	225

Picture 29: Query t2=02.05.2012

In the last query we now filter day between 31.12.2011 and 01.06.2012. Remember, there had been inflows on Jan 1<sup>st</sup>, March 1<sup>st</sup>, Mai 1<sup>st</sup> and June 1<sup>st</sup>.

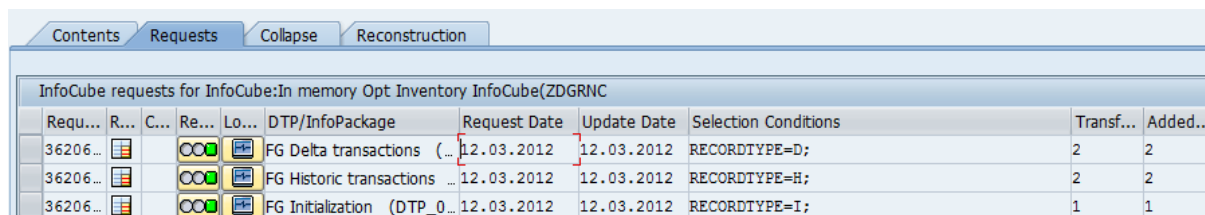
	A	B	C	D
1				Quantity Total Stock
2	<b>Material</b>	<b>Plant</b>	<b>Calendar day</b>	PCE
3	A001	S001	31.12.2011	0
4			01.01.2012	100
5			02.01.2012	100
6			29.02.2012	100
7			01.03.2012	150
8			02.03.2012	150
9			31.03.2012	150
10			01.04.2012	150
11			02.04.2012	150
12			30.04.2012	150
13			01.05.2012	225
14			02.05.2012	225
15			01.06.2012	250

Picture 30: Query t2=01.06.2012

## 7. Influence of InfoCube Compression

In the last section we now want to demonstrate what happens if you compress (collapse) the InfoCube (changes highlighted)

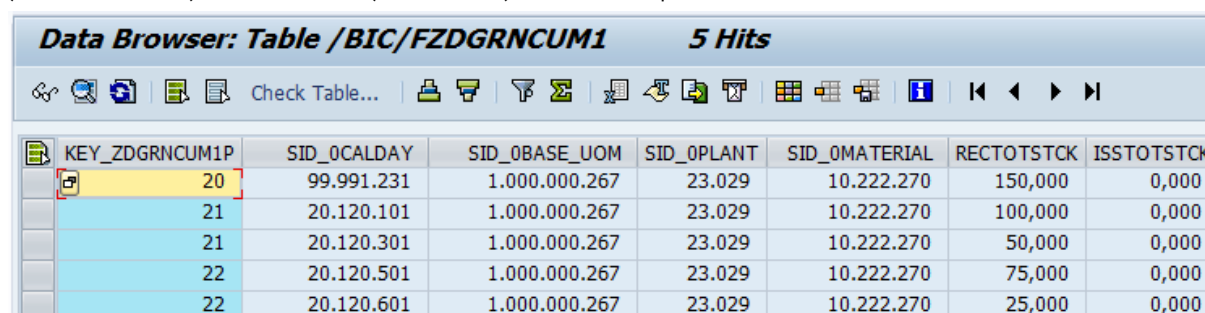
Status before compression:



Requ...	R...	C...	Re...	Lo...	DTP/InfoPackage	Request Date	Update Date	Selection Conditions	Transf...	Added...
36206...					FG Delta transactions (...)	12.03.2012	12.03.2012	RECORDTYPE=D;	2	2
36206...					FG Historic transactions ...	12.03.2012	12.03.2012	RECORDTYPE=H;	2	2
36206...					FG Initialization (DTP_0...	12.03.2012	12.03.2012	RECORDTYPE=I;	1	1

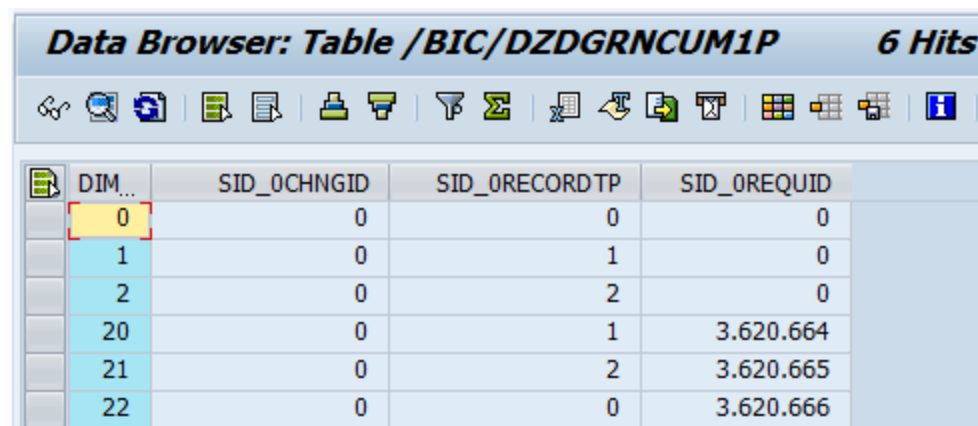
Picture 31: Monitor Status before compression

Again, let us have a look at the contents of the fact table (see Picture 32), package dimension table (see Picture 33) on HDB level (Picture 34) before compression.



KEY_ZDGRNCUM1P	SID_0CALDAY	SID_0BASE_UOM	SID_0PLANT	SID_0MATERIAL	RECTOTSTCK	ISSTOTSTCK
20	99.991.231	1.000.000.267	23.029	10.222.270	150,000	0,000
21	20.120.101	1.000.000.267	23.029	10.222.270	100,000	0,000
21	20.120.301	1.000.000.267	23.029	10.222.270	50,000	0,000
22	20.120.501	1.000.000.267	23.029	10.222.270	75,000	0,000
22	20.120.601	1.000.000.267	23.029	10.222.270	25,000	0,000

Picture 32: Fact table view before compression



DIM...	SID_0CHNGID	SID_0RECORDTP	SID_0REQUID
0	0	0	0
1	0	1	0
2	0	2	0
20	0	1	3.620.664
21	0	2	3.620.665
22	0	0	3.620.666

Picture 33: Package dimension view before compression

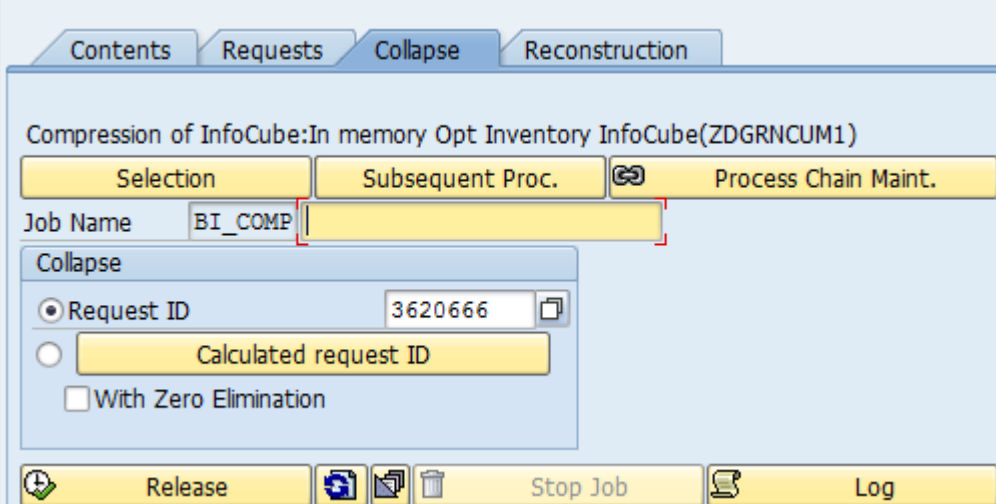
Table Name:	Schema:
/BIC/FZDGRNCUM1	SAPOH0
<a href="#">Columns</a>   <a href="#">Indexes</a>   <a href="#">Further Properties</a>   <a href="#">Runtime Information</a>	
<b>General</b> Table Size: 236 KB Records: 5 Size (Main): 22 KB Size (Delta): 213 KB	

## Details for Table

Parts		Columns					
Part ID	Host	Port	Total Size [KB]	Main Size [KB]	Delta Size [KB]	Records	
1	hdbpoc00	30003	58	6	52	0	
2	hdbpoc00	30003	58	6	52	0	
3	hdbpoc00	30003	58	6	52	0	
4	hdbpoc00	30003	64	6	58	5	

Picture 34: HDB View before compression

We now start the InfoCube compression by collapsing the InfoCube (see Picture 35).



Picture 35: Collapsing InfoCube

The success of the compression can also be seen by looking at the monitor (see Picture 36)

InfoCube requests for InfoCube:ZDGRNCUM1(ZDGRNCUM1)											
Request ID	Req...	Com...	Re...	Lo...	DTP/I...	Request Date	Update Date	Selection Conditions	Transf...	Added...	Type of Data Update
3620661					Stock D...	06.03.2012	06.03.2012	SATZTYP=D;	2	2	Full update
3620660					Historica...	06.03.2012	06.03.2012	SATZTYP=H;	2	2	Full update
3620659					Initializat...	06.03.2012	06.03.2012	SATZTYP=I;	1	1	Generate Initial Status

Picture 36: Monitor after compression

By comparison between Picture 37 and Picture 38 you can see, that the records of the initialization request got moved into the dimension relevant for Initialization records (Orecordtp = '1'). The values of the keyfigures remain the same, so these are not updated. This is also completely different to the behavior of BW releases on traditional database technologies. Also the historic transactions are still visible (DIMID = '2'), that's also different to the concept in a non HDB version.

**Data Browser: Table /BIC/FZDGRNCUM1 5 Hits**

KEY_ZDGRNCUM1P	SID_0CALDAY	SID_0BASE_UOM	SID_0PLANT	SID_0MATERIAL	RECTOTSTCK	ISSTOTSTCK
20	99.991.231	1.000.000.267	23.029	10.222.270	150,000	0,000
21	20.120.101	1.000.000.267	23.029	10.222.270	100,000	0,000
21	20.120.301	1.000.000.267	23.029	10.222.270	50,000	0,000
22	20.120.501	1.000.000.267	23.029	10.222.270	75,000	0,000
22	20.120.601	1.000.000.267	23.029	10.222.270	25,000	0,000

Picture 37: Fact table view before compression

**Data Browser: Table /BIC/FZDGRNCUM1 5 Hits**

KEY_ZDGRNCUM...	SID_0CALDAY	SID_0BASE_UOM	SID_0PLA...	SID_0MATERIAL	RECTOTSTCK	ISSTOTSTCK
0	20.120.501	1.000.000.267	23.029	10.222.270	75,000	0,000
0	20.120.601	1.000.000.267	23.029	10.222.270	25,000	0,000
1	99.991.231	1.000.000.267	23.029	10.222.270	150,000	0,000
2	20.120.101	1.000.000.267	23.029	10.222.270	100,000	0,000
2	20.120.301	1.000.000.267	23.029	10.222.270	50,000	0,000

Picture 38: Fact table view after compression

**Data Browser: Table /BIC/DZDGRNCUM1P 6 Hits**

DIM...	SID_0CHNGID	SID_0RECORDTP	SID_0REQUID
0	0	0	0
1	0	1	0
2	0	2	0
20	0	1	3.620.664
21	0	2	3.620.665
22	0	0	3.620.666

Picture 39: Package dimension before compression

**Data Browser: Table /BIC/DZDGRNCUM1P 3 Hits**

DIM...	SID_0CHNG...	SID_0RECORDT...	SID_0REQU...
0	0	0	0
1	0	1	0
2	0	2	0

Picture 40: Package dimension after compression

As you can see, the DIMIDs 20, 21 and 22 got removed from the package dimension table.



Table Name:		Schema:
/BIC/FZDGRNCUM1		SAPOH0
Columns	Indexes	Further Properties
Runtime Information		
General		
Table Size: 236 KB		Size (Main): 22 KB
Records: 5		Size (Delta): 213 KB

**Details for Table**

Parts	Columns					
Part ID	Host	Port	Total Size [KB]	Main Size [KB]	Delta Size [KB]	Records
1	hdbpoc00	30003	58	6	52	0
2	hdbpoc00	30003	58	6	52	0
3	hdbpoc00	30003	58	6	52	0
4	hdbpoc00	30003	64	6	58	5

Picture 41: HDB View before compression

Table Name:		Schema:
/BIC/FZDGRNCUM1		SAPOH0
Columns	Indexes	Further Properties
Runtime Information		
General		
Table Size: 246 KB		Size (Main): 24 KB
Records: 5		Size (Delta): 221 KB

**Details for Table**

Parts	Columns					
Part ID	Host	Port	Total Size [KB]	Main Size [KB]	Delta Size [KB]	Records
1	hdbpoc00	30003	63	6	57	2
2	hdbpoc00	30003	63	6	56	1
3	hdbpoc00	30003	63	6	57	2
4	hdbpoc00	30003	58	6	52	0

Picture 42: HDB view after compression

On HDB level you can see that the records got distributed to the various partitions. That's the main reason why you should compress your InfoCube regularly.

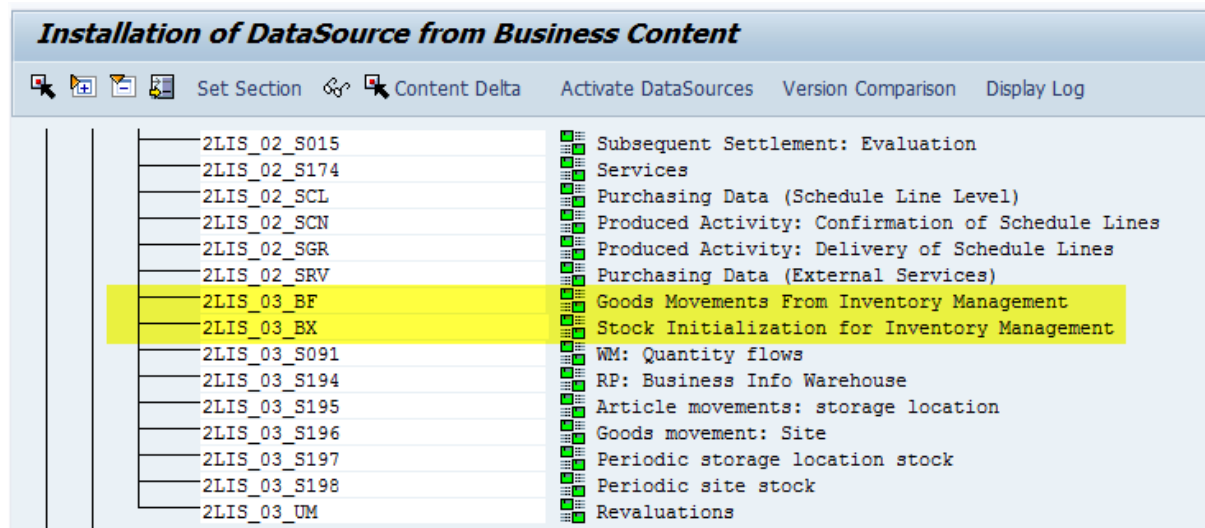
## 8. Extraction from SAP ERP

The following datasources are to be used in the extraction of an inventory scenario based on SAP ERP (see Picture 43: SAP ERP based inventory sources):

2LIS\_03\_BX Material stock

2LIS\_03\_BF Material movements

2LIS\_03\_UM Revaluations



Picture 43: SAP ERP based inventory sources

The first datasource (2LIS\_03\_BX) is used to extract an opening stock balance on a detailed level (material, plant, storage location and so on). The opening stock is the situation of the operative stock in the SAP ERP system to the time you scheduled the statistical setup ran for the datasource 2LIS\_03\_BX. (This is because no documents are to be posted during this run and so the stock does not change during this run). It is not possible to choose a key date freely.


The second datasource (2LIS\_03\_BF) is used to extract the material movements into the SAP BW system. This datasource is used to extract historic as well as delta transactions (see Picture 45).

The third of the above datasources (2LIS\_03\_UM) contains data from valuated revaluations in Financial Accounting (document BSEG). This data is required to update valuated stock changes for the calculated stock balance in the BW. This information is not required in many situations as it is often only the quantities that are of importance. Therefore the handling of that DataSource is not treated in this document. This datasource only describes financial accounting processes, not logistical ones. In other words, only the stock value is changed here, no changes are made to the stock quantities. Everything that is subsequently mentioned here about the upload sequence and compression regarding datasource 2LIS\_03\_BF also applies to this datasource.













The complete picture for a SAP Retail based extractions looks like in Picture 44.

Inventory Management	ORT_STOCK
Stock (Month)	ORT_C37
Stock (Week)	ORT_C36
TRCS 2LIS_03_BF_TR -> CUBE ORT_C36	03R8D1YZ0LV2M23...
TRCS 2LIS_03_BX_TR -> CUBE ORT_C36	0Q884VBV0GGWE...
Data Transfer Processes	ORT_C36
Delta movements 2LIS_03_BF to Inventory InfoCube	DTP_DI6X2E8ARYJ...
Historic Movements 2LIS_03_BF to Inventory InfoCube	DTP_DI6X2E8ARYJ...
Initialization 2LIS_03_BX to Inventory Cube	DTP_DI6X2E8ARYJ...

Picture 44: Inventory management for SAP Retail based InfoCube ORT\_C36

Selectable Data Targets for Administration									
Name	D...	Technical Name	Table Type						
Stock (Week)		ORT_C36	InfoCube						

InfoCube requests for InfoCube:Stock (Week)(ORT_C36)									
Request ID	R...	C...	D...	Re...	Lo...	DTP/InfoPackage	Request D...	Update Date	
59						Delta movements 2LIS_03_BF to Inventory InfoCube (DT...	15.03.2012	15.03.2012	
58						Historic Movements 2LIS_03_BF to Inventory InfoCube (...)	15.03.2012	15.03.2012	
50						Initialization 2LIS_03_BX to Inventory Cube (DTP_DI6X2E...	15.03.2012	15.03.2012	

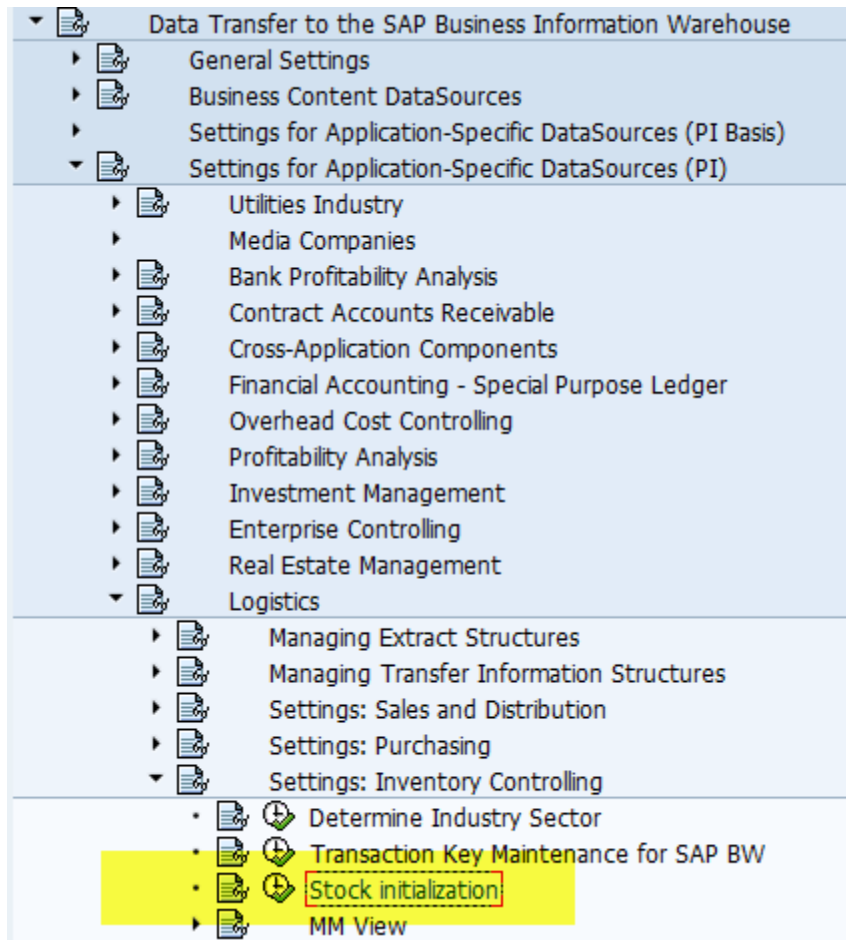
Picture 45: Inventory InfoCube fed from SAP sources 2LIS\_03\_BX and 2LIS\_03\_BF

## 8.1 Extracting inventory data (standard procedure)

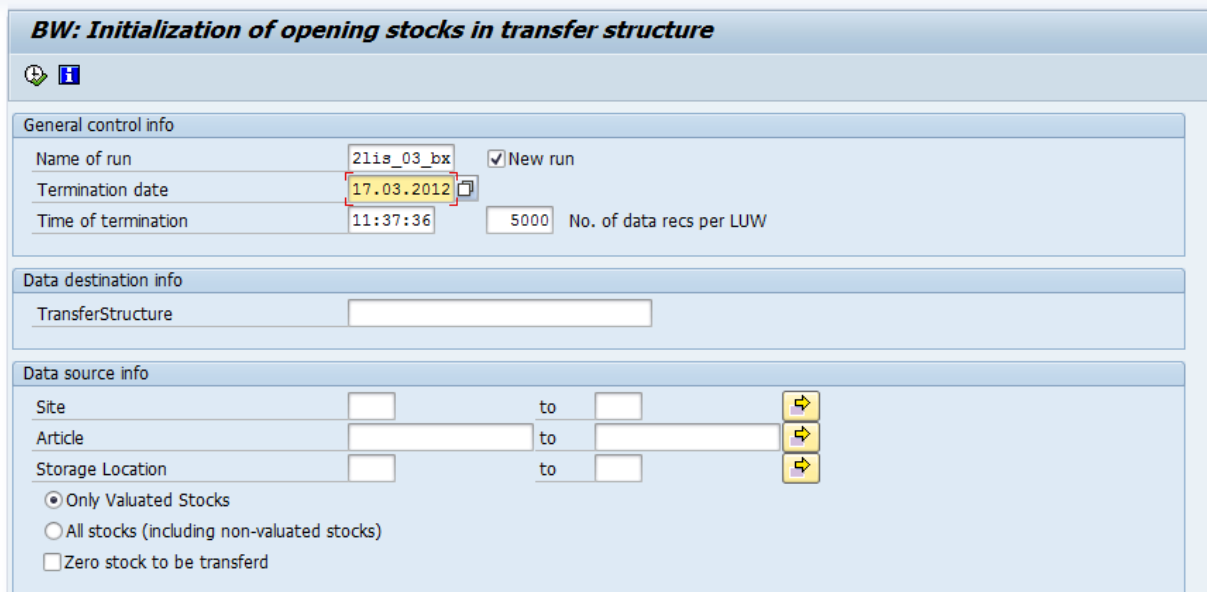
For previous releases of BW this is already very well documented in <http://scn.sap.com/docs/DOC-35750>

A data flow could look like this:

- Lock users in ECC  
While you initialize the setup tables for inventory transaction changes to the documents that are written to the setup table are not allowed. Therefore you have to lock the application in ERP system against changes to material documents.
- Fill setup table for 2LIS\_03\_BX.  
This can be done with transaction MCNB. In case you want to speed up the whole process you can schedule several variants of the report with distinct selections.  
Before you schedule the initialization please ensure that no data from a previous initialization exists in the setup table. Therefore please delete the content of the table first.



Picture 46: Transaction SBIW: Stock Initialization



The screenshot shows the SAP transaction SBIW: Initialization of opening stocks in transfer structure. The form is divided into three main sections: General control info, Data destination info, and Data source info.

**General control info**

Name of run	211s_03_bx	<input checked="" type="checkbox"/> New run
Termination date	17.03.2012	<input type="checkbox"/>
Time of termination	11:37:36	5000 No. of data recs per LUW

**Data destination info**

TransferStructure	
-------------------	--

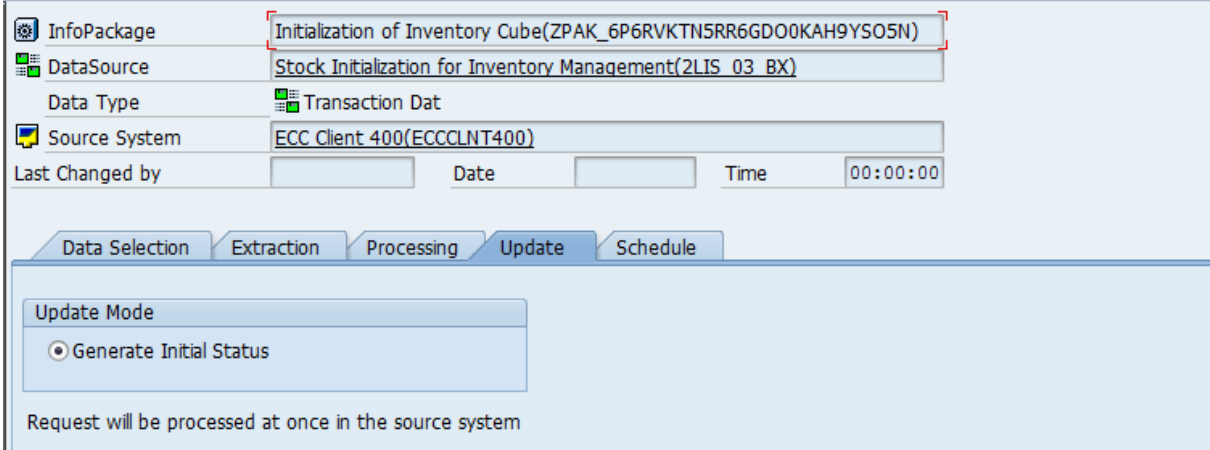
**Data source info**

Site		to		<input type="button" value="→"/>
Article		to		<input type="button" value="→"/>
Storage Location		to		<input type="button" value="→"/>

☒ Only Valuated Stocks  
☐ All stocks (including non-valuated stocks)  
☐ Zero stock to be transferd

Picture 47: Fill setup table for 2LIS\_03\_BX

- c) Extract Initialization records with DataSource 2LIS\_03\_BX to PSA and start subsequent DTP for Initialization.



InfoPackage: Initialization of Inventory Cube(ZPAK\_6P6RVKTN5RR6GDO0KAH9YSO5N)

DataSource: Stock Initialization for Inventory Management(2LIS\_03\_BX)

Data Type: Transaction Dat

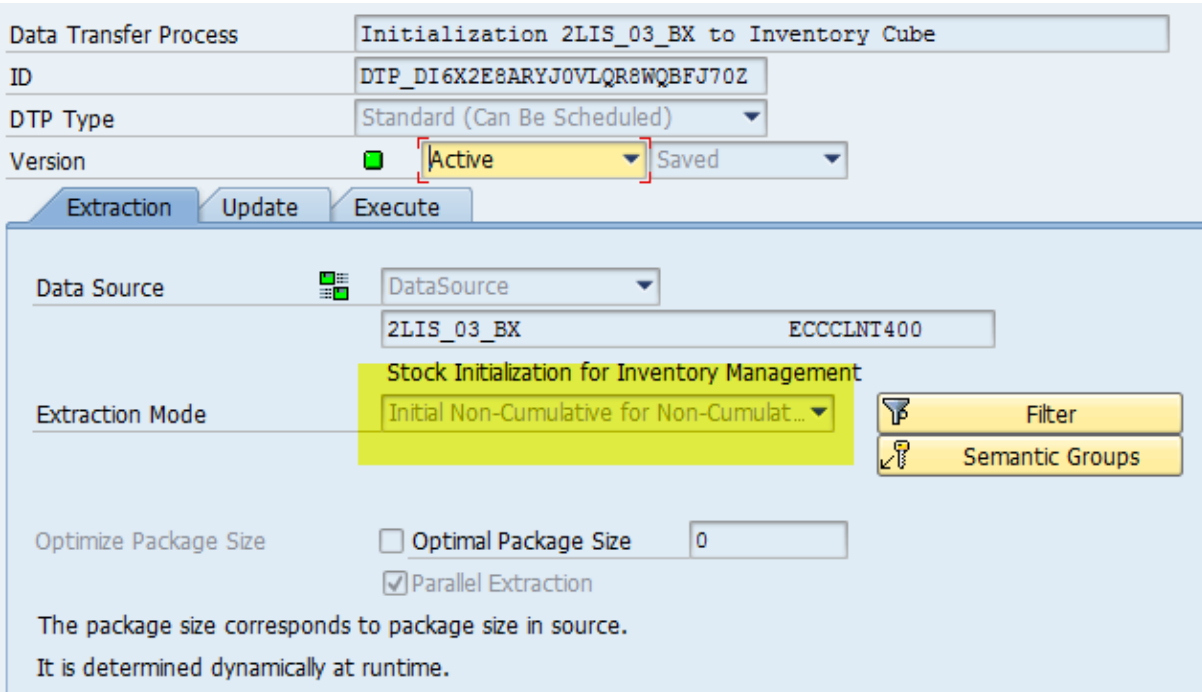
Source System: ECC Client 400(ECCCLNT400)

Last Changed by: Date: Time: 00:00:00

Update Mode: ☒ Generate Initial Status

Request will be processed at once in the source system

Picture 48: Initialization InfoPackage for 2LIS\_03\_BX



Data Transfer Process: Initialization 2LIS\_03\_BX to Inventory Cube

ID: DTP\_DI6X2E8ARYJ0VLQR8WQBFJ70Z

DTP Type: Standard (Can Be Scheduled)

Version: ☒ Active ☐ Saved

Extraction Mode: Initial Non-Cumulative for Non-Cumulat...

Data Source: 2LIS\_03\_BX ECCCLNT400

Optimize Package Size: ☐ Optimal Package Size 0 ☒ Parallel Extraction

The package size corresponds to package size in source.  
It is determined dynamically at runtime.

Picture 49: DTP for Initialization (2LIS\_03\_BX)

- d) Fill setup table for material movements in ERP.

Before starting the filling of the setup table please ensure that the setup table is empty.

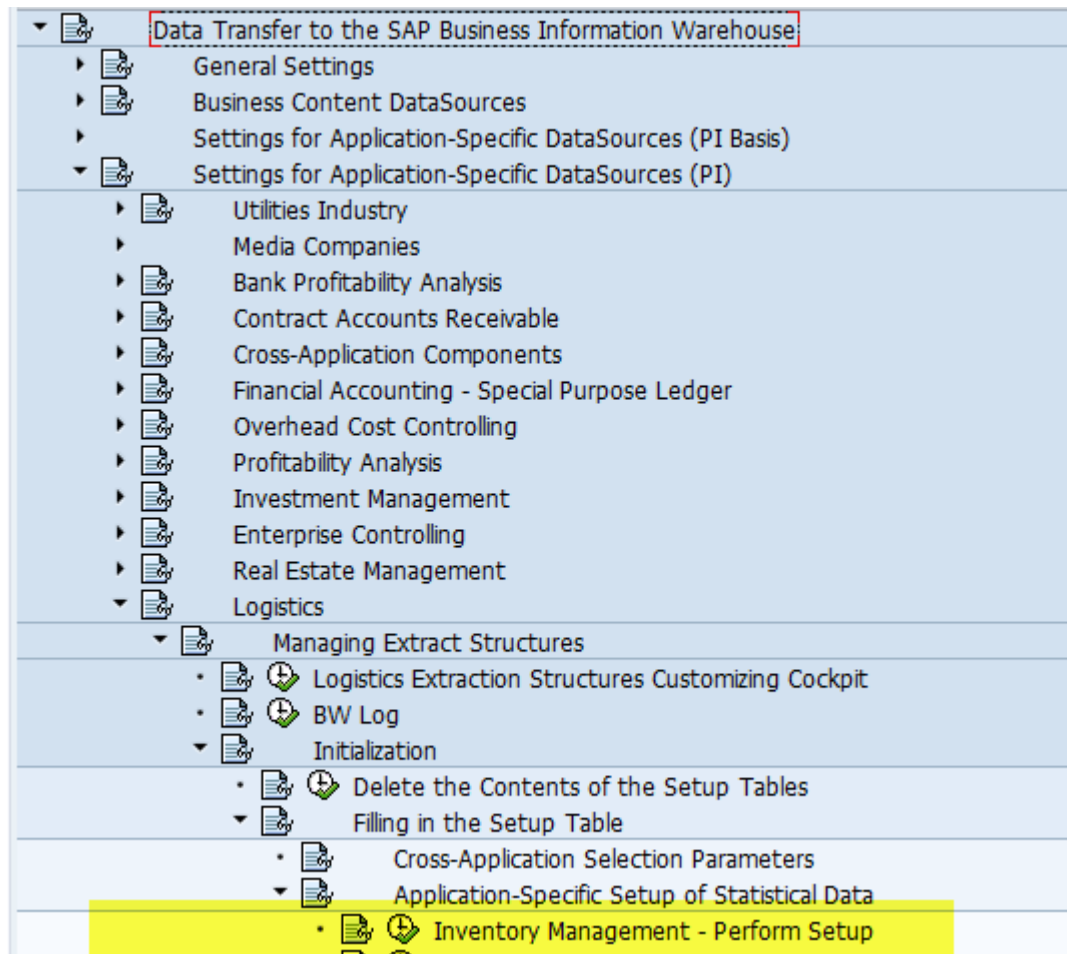
Therefore delete the content of the setup table first.

Fill the setup table by calling transaction OLI1BW. You can schedule various variants of the program with distinct selections to speed up the whole process of filling the setup table.

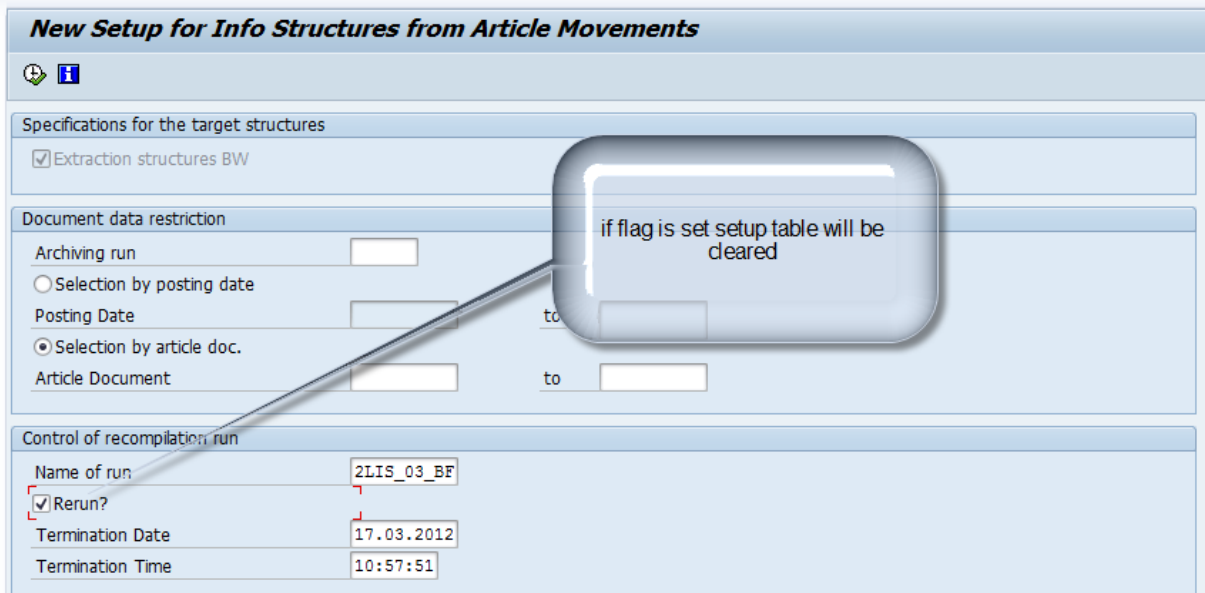


To minimize the downtime for SAP ERP you can split the history of material movement documents into those for which still changes apply and those for which no changes are allowed anymore (due to closed periods, etc).

I will describe that scenario in section 8.2 of this document. In case you have low volumes in ECC and the lock phase in ECC is not an issue, you can continue with the steps described in this section.



Picture 50: Customizing: Fill setup tables for historic inventory transactions



The screenshot shows the 'New Setup for Info Structures from Article Movements' dialog box. It contains the following sections:

- Specifications for the target structures:** ☒ Extraction structures BW
- Document data restriction:**
  - Archiving run:
  - Selection by posting date: ☐ Posting Date:  to
  - Selection by article doc.: ☒ Article Document:  to
- Control of recompilation run:**
  - Name of run: 2LIS\_03\_BF
  - ☒ Rerun?
  - Termination Date: 17.03.2012
  - Termination Time: 10:57:51

A callout box points to the 'Rerun?' checkbox with the text: 'if flag is set setup table will be cleared'.

Picture 51: Fill setup table for 2LIS\_03\_BF

- e) Delta initialization for material movements.  
Once the filling of the setup table is finished you can initialize the delta upload for DataSource 2LIS\_03\_BF by starting the InfoPackage in BW.

InfoPackage	Delta movements(ZPAK_CO7FXQZS5LMTR4SF5UXTRXPOR)		
DataSource	Goods Movements From Inventory Management(2LIS_03_BF)		
Data Type	Transaction Dat		
Source System	ECC Client 400(ECCCLNT400)		
Last Changed by	HOFFMANNST	Date	15.03.2012 Time 14:24:46

Data Selection

Extraction

Processing

Data Targets

Update

Schedule

Update Mode

☐ Full update  
☐ Delta Update  
☒ Initialize Delta Process  

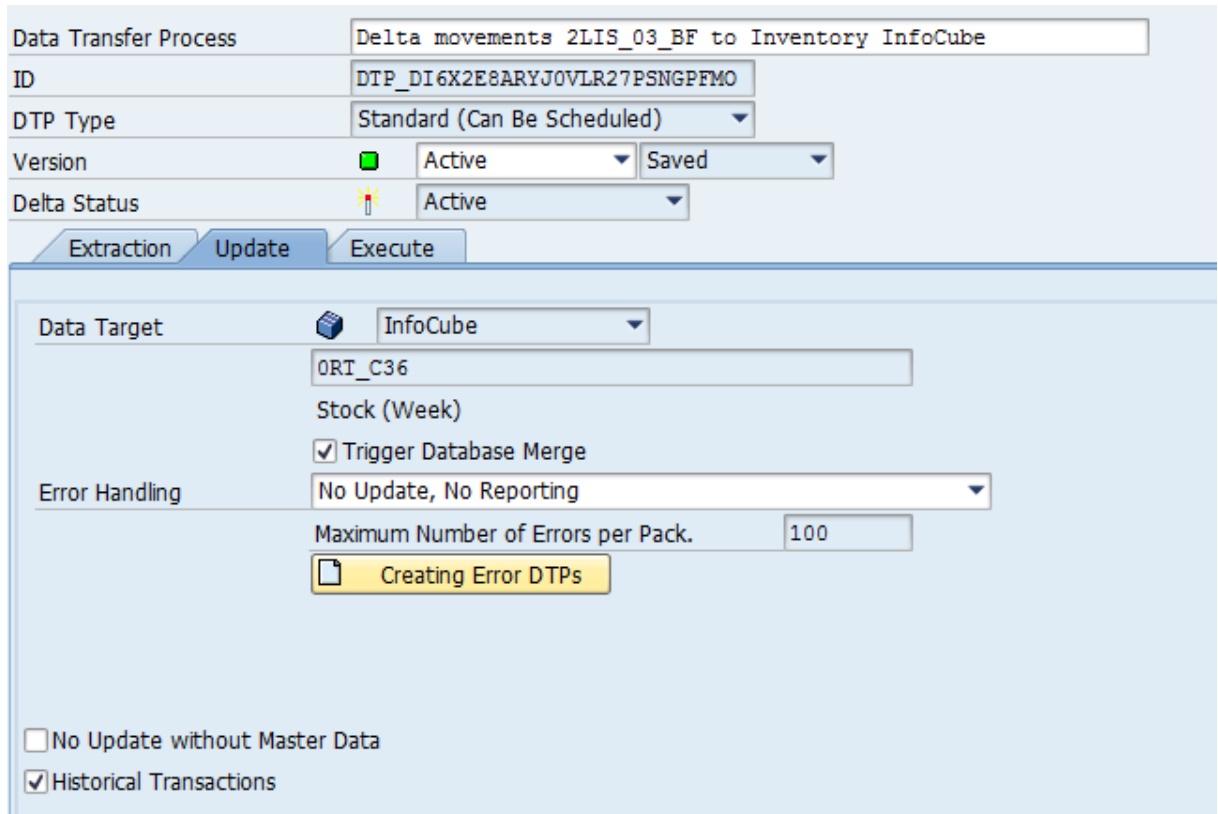
☒ Initialization with Data Transfer  
☐ Initialize Without Data Transfer  
☐ Early Delta Initialization  
☐ Switch InfoPack. in PC to Delta (F1)

Request will be processed at once in the source system

Picture 52: InfoPackage for delta documents

- f) Unlock users in ERP again  
After successful creation of the delta queue in ECC for 2LIS\_03\_BF (can be checked with transaction RSA7 in ECC), you can allow again changes to material movement documents in ECC.
- g) Start subsequent DTP for delta transactions (Delta DTP).



The screenshot shows the SAP Delta Transfer Process (DTP) configuration interface. The top section contains fields for 'Data Transfer Process' (Delta movements 2LIS\_03\_BF to Inventory InfoCube), 'ID' (DTP\_DI6X2E8ARYJ0VLR27PSNGPFMO), 'DTP Type' (Standard (Can Be Scheduled)), 'Version' (Active), and 'Delta Status' (Active). Below these are tabs for 'Extraction', 'Update', and 'Execute'. The 'Update' tab is selected, showing the 'Data Target' as 'InfoCube' with a cube icon. The 'Data Target' field is expanded, showing 'ORT\_C36' and 'Stock (Week)'. The 'Error Handling' section is also expanded, showing 'No Update, No Reporting' as the selected option, 'Maximum Number of Errors per Pack.' set to 100, and a button labeled 'Creating Error DTPs'. At the bottom, there are checkboxes for 'No Update without Master Data' (unchecked) and 'Historical Transactions' (checked).

Picture 53: Delta DTP for Delta transactions (material movement documents of 2LIS\_03\_BF)



We recommend putting a LSA architecture in place from which you could in case of reconstruction at least get the initialization and historic records again.

- h) Setup extracting regularly deltas from ECC followed by Delta DTPs



InfoPackage	Delta movements(ZPAK_CO7FXQZS5LMTR4SF5UXTRXPOR)		
DataSource	Goods Movements From Inventory Management(2LIS_03_BF)		
Data Type	Transaction Dat		
Source System	ECC Client 400(ECCCLNT400)		
Last Changed by	HOFFMANNST	Date	15.03.2012 Time 14:24:46



  

Data Selection
Extraction
Processing
Data Targets
Update
Schedule



  

☐ Start Data Load Immediately  
☒ Start Later in Background

Job Name Prefix/Suffix: BI\_BTCH
 

 Scheduling Options ✓  
 Subsequent Process.

☐ Request Batch Process Runs Until All Data Has Been Updated in BW

 Start
  Job(s)

Picture 54: InfoPackage to extract deltas from ECC


Data Transfer Process	Delta movements 2LIS_03_BF to Inventory InfoCube	
ID	DTP_DI6X2E8ARYJ0VLR27PSNGPFMO	
DTP Type	Standard (Can Be Scheduled)	
Version	Active	Saved
Delta Status	Active	

Extraction
Update
Execute

Data Target: InfoCube  
 ORT\_C36  
 Stock (Week)  
☒ Trigger Database Merge

Error Handling: No Update, No Reporting  
 Maximum Number of Errors per Pack: 100  
 Creating Error DTPs

☐ No Update without Master Data  
☐ Historical Transactions

Picture 55: Delta DTP for Delta transactions (material movement documents of 2LIS\_03\_BF)

## 8.2 Extracting inventory data (procedure to minimize locking phase in ECC)

In case you want to minimize the time you have to lock the users in ECC, you can also proceed as follows

A data flow could look like this:

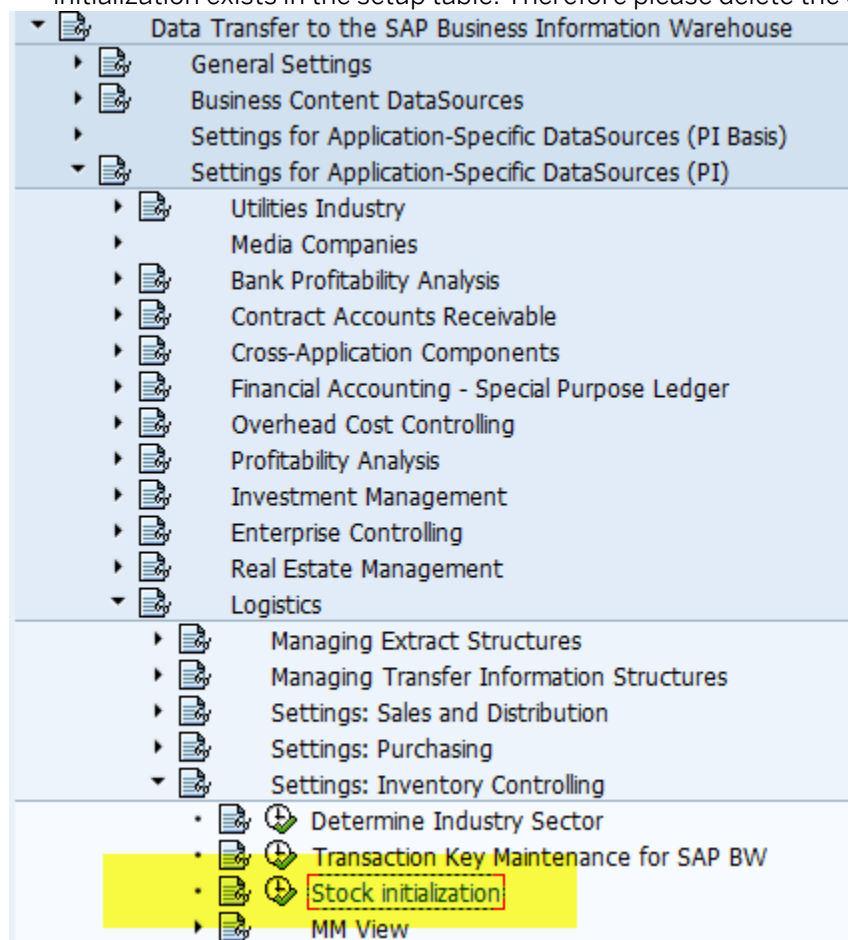
- a) Lock users in ECC

While you initialize the setup tables for inventory transaction changes to the documents that are written to the setup table are not allowed. Therefore you have to lock the application in ERP system against changes to material documents.

- b) Fill setup table for 2LIS\_03\_BX.

This can be done with transaction MCNB. In case you want to speed up the whole process you can schedule several variants of the report with distinct selections.

Before you schedule the initialization please ensure that no data from a previous initialization exists in the setup table. Therefore please delete the content of the table first.



Picture 56: Transaction SBIW: Stock Initialization

**BW: Initialization of opening stocks in transfer structure**

General control info

Name of run	21is_03_bx	<input checked="" type="checkbox"/> New run
Termination date	17.03.2012	
Time of termination	11:37:36	5000 No. of data recs per LUW

Data destination info

TransferStructure

Data source info

Site to

Article to

Storage Location to

☒ Only Valuated Stocks  
☐ All stocks (including non-valuated stocks)  
☐ Zero stock to be transferd

Picture 57: Fill setup table for 2LIS\_03\_BX

- c) Extract Initialization records with DataSource 2LIS\_03\_BX to PSA and start subsequent DTP for Initialization.

InfoPackage Initialization of Inventory Cube(ZPAK\_6P6RVKTN5RR6GDO0KAH9YSO5N)

DataSource Stock Initialization for Inventory Management(2LIS\_03\_BX)

Data Type Transaction Dat

Source System ECC Client 400(ECCCLNT400)

Last Changed by Date Time 00:00:00

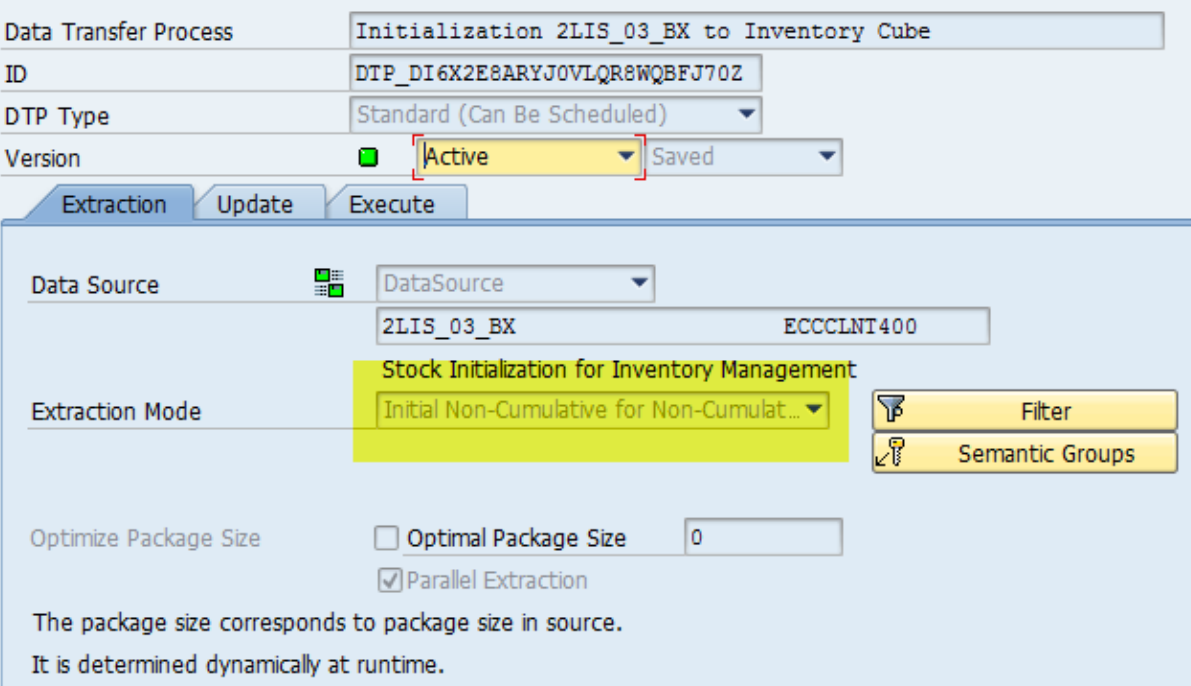
Data Selection Extraction Processing Update Schedule

Update Mode

☒ Generate Initial Status

Request will be processed at once in the source system

Picture 58: Initialization InfoPackage for 2LIS\_03\_BX



**Data Transfer Process** Initialization 2LIS\_03\_BX to Inventory Cube

**ID** DTP\_DI6X2E8ARYJ0VLQR8WQBFJ70Z

**DTP Type** Standard (Can Be Scheduled)

**Version** ■ Active Saved

**Extraction** **Update** **Execute**

**Data Source** DataSource  
2LIS\_03\_BX ECCCLNT400

**Extraction Mode** Stock Initialization for Inventory Management  
Initial Non-Cumulative for Non-Cumulat...

**Optimize Package Size** ☐ Optimal Package Size 0 ☒ Parallel Extraction

The package size corresponds to package size in source.  
It is determined dynamically at runtime.

**Filter**  
**Semantic Groups**

Picture 59: DTP for Initialization (2LIS\_03\_BX)

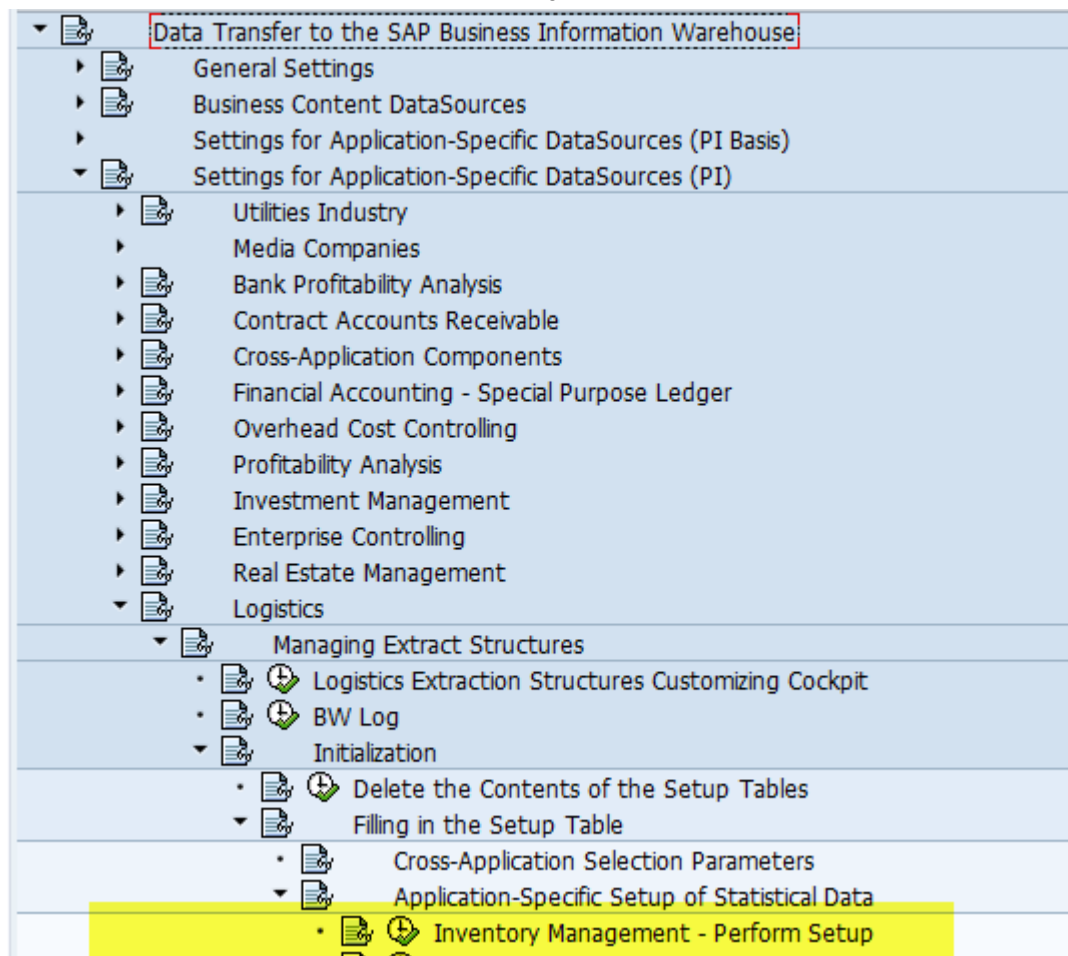
- d) Fill setup table for material movements transactions in ERP to which still changes can happen (Selection A).

This is performed by calling transaction OLI1BW. You can schedule various variants of the program with distinct selections to speed up the whole process of filling the setup table. Before filling the setup table please ensure that there is no data in from a previous run. Therefore delete the content of the table first.

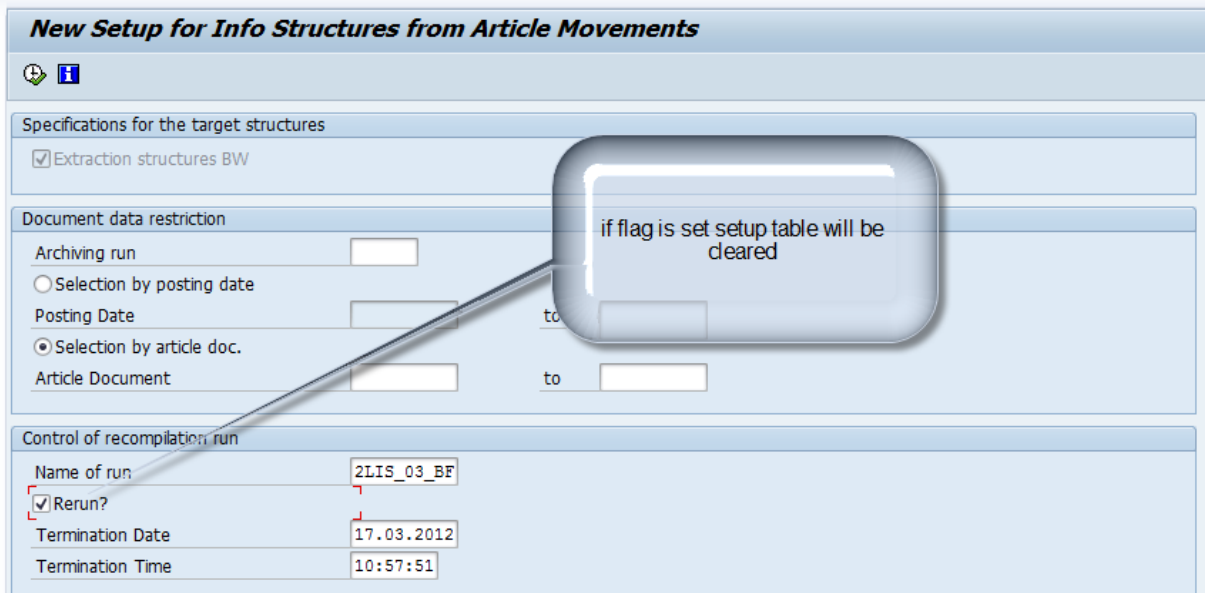


To minimize the downtime for SAP ERP you split the history of inventory documents into those for which still changes apply (Selection A) and those for which no changes are allowed anymore (due to closed periods, etc) (Selection B).

Fill the setup table by restricting the selection in OLI1BW to that period / documents where changes to material documents still are allowed to occur (Selection A).



Picture 60: Customizing: Fill setup tables for historic inventory transactions



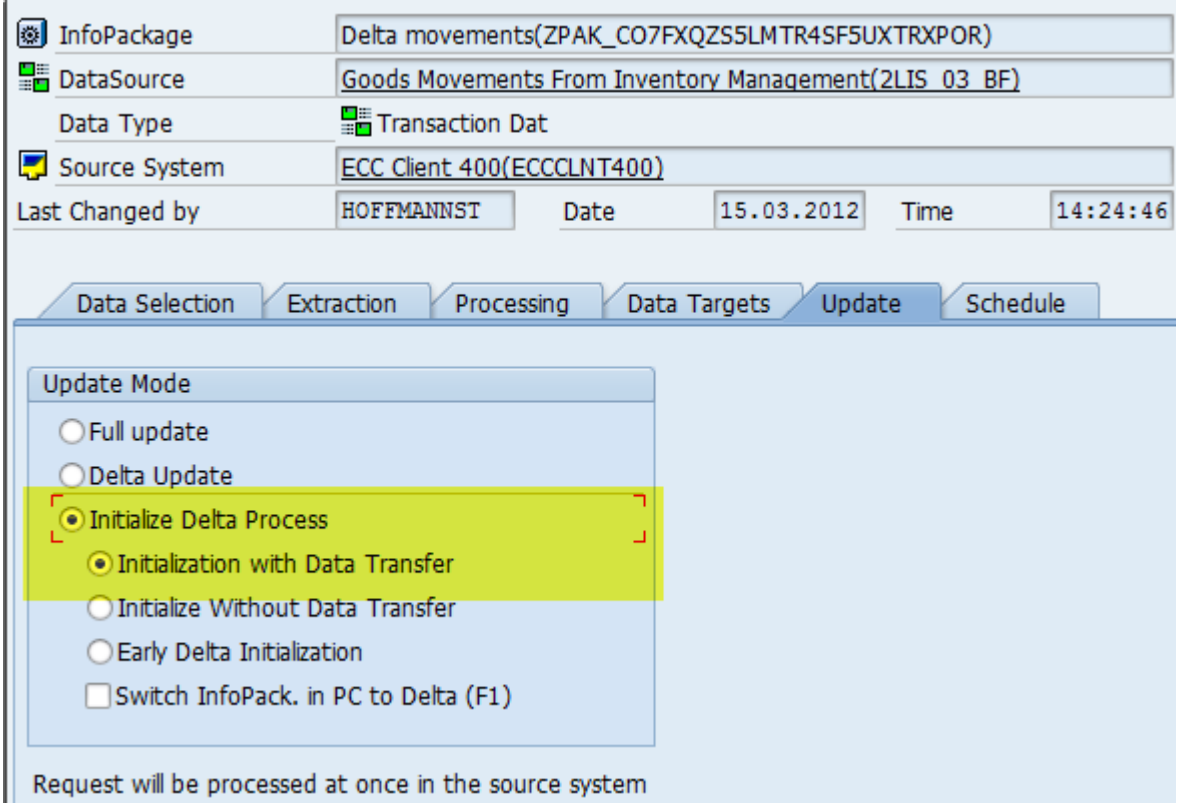
The dialog box is titled **New Setup for Info Structures from Article Movements**. It contains the following sections:

- Specifications for the target structures:** ☒ Extraction structures BW
- Document data restriction:**
  - Archiving run:
  - ☐ Selection by posting date
  - Posting Date:  to
  - ☒ Selection by article doc.
  - Article Document:  to
- Control of recompilation run:**
  - Name of run:
  - ☒ Rerun?
  - Termination Date:
  - Termination Time:

A callout box points to the **Rerun?** checkbox with the text: "if flag is set setup table will be cleared".

Picture 61: Fill setup table for 2LIS\_03\_BF

- e) Delta initialization for material movements.  
Once the filling of the setup table is finished you can initialize the delta upload for DataSource 2LIS\_03\_BF by starting the InfoPackage in BW.



The screenshot shows the SAP BW configuration interface for an InfoPackage. The 'InfoPackage' field is set to 'Delta movements(ZPAK\_CO7FXQZS5LMTR4SF5UXTRXPOR)'. The 'DataSource' is 'Goods Movements From Inventory Management(2LIS\_03\_BF)'. The 'Data Type' is 'Transaction Dat'. The 'Source System' is 'ECC Client 400(ECCCLNT400)'. The 'Last Changed by' is 'HOFFMANNST', the 'Date' is '15.03.2012', and the 'Time' is '14:24:46'. The 'Update' tab is selected, showing the 'Update Mode' section. The 'Initialize Delta Process' option is selected, with 'Initialization with Data Transfer' as a sub-option. Other options include 'Full update', 'Delta Update', 'Initialize Without Data Transfer', 'Early Delta Initialization', and 'Switch InfoPack. in PC to Delta (F1)'. A note at the bottom states 'Request will be processed at once in the source system'.

InfoPackage	Delta movements(ZPAK_CO7FXQZS5LMTR4SF5UXTRXPOR)		
DataSource	Goods Movements From Inventory Management(2LIS_03_BF)		
Data Type	Transaction Dat		
Source System	ECC Client 400(ECCCLNT400)		
Last Changed by	HOFFMANNST	Date	15.03.2012
		Time	14:24:46




Update Mode

- ☐ Full update
- ☐ Delta Update
- ☒ Initialize Delta Process
  - ☒ Initialization with Data Transfer
  - ☐ Initialize Without Data Transfer
- ☐ Early Delta Initialization
- ☐ Switch InfoPack. in PC to Delta (F1)

Request will be processed at once in the source system

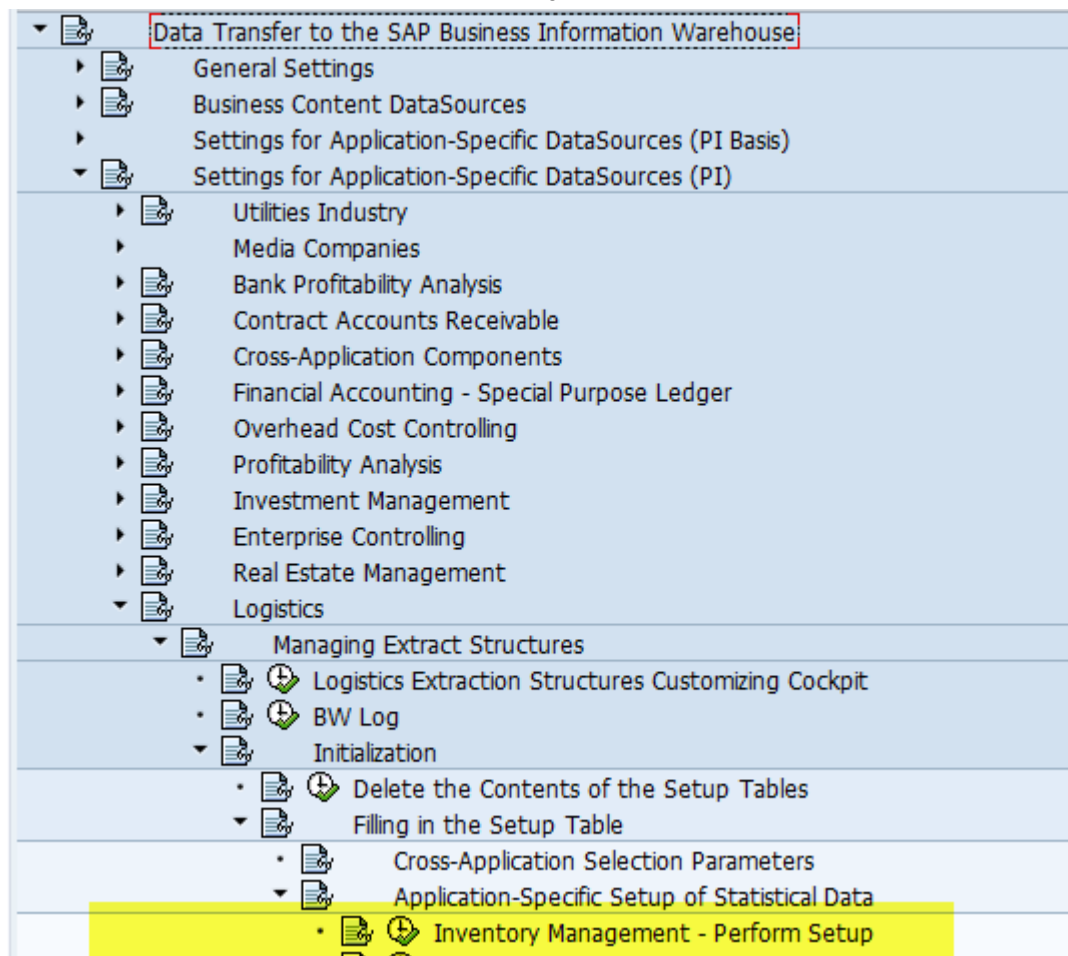
Picture 62: InfoPackage for delta documents

- f) Unlock users in ERP again (in case you locked them before)  
After successful creation of the delta queue in ECC for 2LIS\_03\_BF (can be checked with transaction RSA7 in ECC), you can allow again changes to material movement documents in ECC.
- g) Start subsequent DTP for delta transactions (Delta DTP).

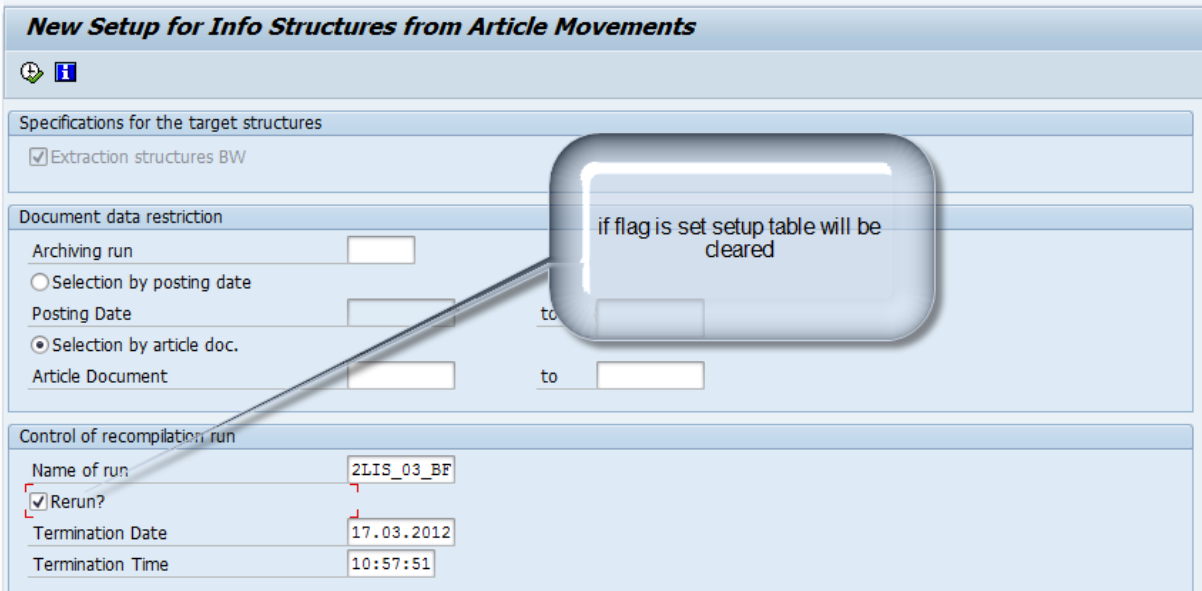
Data Transfer Process	Delta movements 2LIS_03_BF to Inventory InfoCube		
ID	DTP_DI6X2E8ARYJ0VLR27PSNGPFMO		
DTP Type	Standard (Can Be Scheduled)		
Version	<input checked="" type="checkbox"/> Active	<input type="checkbox"/> Saved	
Delta Status	 Active		
<div>Extraction   <b>Update</b>   Execute</div>			
Data Target	 InfoCube		
	0RT_C36		
	Stock (Week)		
	<input checked="" type="checkbox"/> Trigger Database Merge		
Error Handling	No Update, No Reporting		
	Maximum Number of Errors per Pack.	100	
	 Creating Error DTPs		
<input type="checkbox"/> No Update without Master Data <input checked="" type="checkbox"/> Historical Transactions			

Picture 63: Delta DTP for Historic Transactions

- h) Fill setup table for material movements transactions in ERP to which no changes can happen any longer (Selection B).  
 Before filling the setup table again delete the content of the previous run in the setup table first.



Picture 64: Customizing: Fill setup tables for historic inventory transactions



The screenshot shows the 'New Setup for Info Structures from Article Movements' dialog box. It contains the following sections:

- Specifications for the target structures:** ☒ Extraction structures BW
- Document data restriction:**
  - Archiving run:
  - ☐ Selection by posting date
  - Posting Date:  to
  - ☒ Selection by article doc.
  - Article Document:  to
- Control of recompilation run:**
  - Name of run:
  - ☒ Rerun?
  - Termination Date:
  - Termination Time:

A callout box points to the 'Rerun?' checkbox with the text: "if flag is set setup table will be cleared".

Picture 65: Fill setup table for 2LIS\_03\_BF

- i) Extract historic movements with DataSource 2LIS\_03\_BF to PSA and start subsequent DTP for historic transactions (Delta DTP).



InfoPackage	Historic transactions Inventory InfoCube(ZPAK_9CQKPPS6MJQB62MHZ65PT)		
DataSource	Goods Movements From Inventory Management(2LIS_03_BF)		
Data Type	Transaction Dat		
Source System	ECC Client 400(ECCCLNT400)		
Last Changed by	HOFFMANNST	Date	15.03.2012 Time 14:33:10

Data Selection

Extraction

Processing

Data Targets

Update

Schedule

Update Mode

☒ Full update
   
☐ Delta Update
   
☐ Initialize Delta Process
   
     ☒ Initialization with Data Transfer
   
     ☐ Initialize Without Data Transfer
   
     ☐ Early Delta Initialization
   
☐ Switch InfoPack. in PC to Delta (F1)

Request will be processed at once in the source system

Picture 66: InfoPackage for historic transactions

Data Transfer Process	Historic Movements 2LIS_03_BF to Inventory InfoCube		
ID	DTP_DI6X2E8ARYJ0VLR2422U5I53V		
DTP Type	Standard (Can Be Scheduled)		
Version	<input checked="" type="checkbox"/> Active	<input type="checkbox"/> Saved	

Extraction

Update

Execute

Data Target

InfoCube

ORT\_C36

Stock (Week)

☒ Trigger Database Merge

Error Handling

No Update, No Reporting

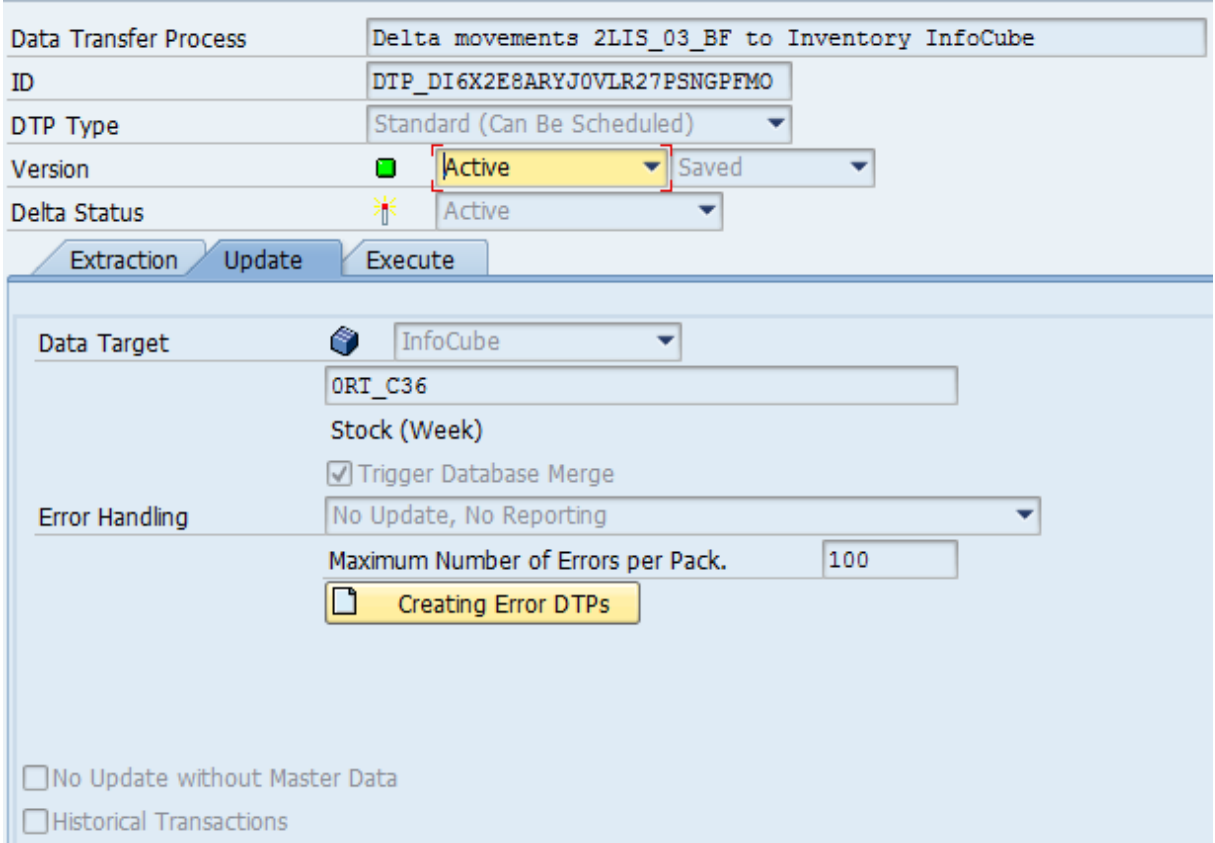
Maximum Number of Errors per Pack. 100

Creating Error DTPs

☐ No Update without Master Data
   
☒ Historical Transactions

Picture 67: Full DTP for historic transactions (2LIS\_03\_BF)

- j) Delete full request with historic transactions from PSA
- k) Start subsequent InfoPackage to get deltas and DTP for delta transactions (Delta DTP).



**Data Transfer Process**: Delta movements 2LIS\_03\_BF to Inventory InfoCube

**ID**: DTP\_DI6X2E8ARYJ0VLR27PSNGPFMO

**DTP Type**: Standard (Can Be Scheduled)

**Version**: Active Saved

**Delta Status**: ✶ Active

**Extraction** **Update** **Execute**

**Data Target**: InfoCube

**ORT\_C36**

**Stock (Week)**

☒ **Trigger Database Merge**

**Error Handling**: No Update, No Reporting

**Maximum Number of Errors per Pack.**: 100

Creating Error DTPs

☐ No Update without Master Data

☐ Historical Transactions

Picture 68: Delta DTP for Delta transactions (material movement documents of 2LIS\_03\_BF)



We recommend putting a LSA architecture in place from which you could in case of reconstruction at least get the initialization and historic records again.

## 9. Considerations regarding conversion of a non In Memory Optimized Inventory InfoCube into a In Memory optimized Inventory InfoCube

Before running the database migration you should check in the source BW (BW running on classic DB) whether all requests in your Inventory Cube which contain historical movements are compressed (shouldn't update the reference point). If there are such requests still uncompressed in the InfoCube (in the F fact table), please compress those (using no marker update option) and all requests with a REQUID smaller than the one carrying historic movements.

If you already migrated your database to HANA please proceed as follows to convert your standard non-cumulative InfoCube into an In Memory optimized InfoCube using transaction RSMIGRHANADB. As the recommendations depend on the installation type of your HANA database, we have to differentiate between a non-distributed HANA system (single node) and a distributed HANA system (multi-node).

### 9.1 Single Node Systems

1. Check whether there are uncompressed historic movements in the InfoCube (see note <http://service.sap.com/sap/support/notes/1766577>). If there are such requests:

- a. Compress **with marker update** all requests with a REQUID smaller than those with historic movements.
  - b. Compress using **no marker update** all requests with historic movements.
2. In general it is recommended to compress the standard non-cumulative InfoCube completely before running the conversion to an In Memory optimized InfoCube. Therefore, compress all other uncompressed requests **with marker update**.
3. In case the complete compression of the standard Inventory InfoCube is not possible / desired, please consider the instructions of note <http://service.sap.com/sap/support/notes/1780575>. This alternative approach is only possible to avoid compression of "regular" requests. **All requests with historic movements must be compressed in any case.**
4. Convert the InfoCube to an In Memory optimized InfoCube.

## 9.2 Multi Node System

1. Check whether there are uncompressed historic movements in the InfoCube (see note 1766577). If there are such requests, delete these requests from InfoCube and reload after InfoCube is converted.
2. Set RSADMIN parameter FORCE\_NCUM\_CUBE\_CONVERSION (see note <http://service.sap.com/sap/support/notes/1780575>).
3. Convert the InfoCube to an In Memory optimized InfoCube.
4. Reload the historic movements.

## 10. Conclusions / Recommendations

You should compress the In Memory Optimized InfoCube regularly, as this keeps the size of the '\*' partition small.

Use semantic partitioning feature to split the InfoCube into semantic partitions. This allows you to control the overall size of the single database partitions. Especially partition 0 and 1 should be of interest. As a rule of thumb you should split your InfoCube into semantic partitions as soon as you reach more than 500 million records per InfoCube.

Non-cumulative InfoCubes should not be partitioned by time.

## 11. Appendix

### 11.1 Query logic

#### Formula – non-HANA BW

$$value(t0) = \left( \sum_{rectp=1} \text{tuple} + \sum_{requid>0} \text{tuple} \right) - \sum_{t>t0} \text{tuple}$$

Ref-Point at infinity
Diff to Ref-Point

**Algorithmic processing to compute inventory values of interval [t1, t2]:**

- Query 1:  
Read the corresponding reference point (recordtp=1) and  
Read all deltas (recordtp=0) which are not yet compressed (requid>0)  
**(no time-cha in group-by)**
- Aggregate result to get inventory value at infinity
- Query 2:  
Read all deltas between t1 and oo  
**(with time-cha in group-by)**
- Loop backwards in time over deltas  
subtract values from corresponding ref-point;  
for each requested time return current value

Picture 69: non-HANA BW non-cumulative handling

## Formula – BW-HANA

$$value(t0) = \left( \sum_{rectp=1} tuple + \sum_{\substack{rectp=0 \\ t \leq t2}} tuple - \sum_{\substack{rectp=2 \\ t > t2}} tuple \right) - \sum_{t2 \geq t > t0} tuple$$

Ref-Point at t2
Diff to Ref-Point

### Algorithmic processing to compute inventory values of interval [t1, t2]:

- Query 1:
  - Read initialization records (rectp=1) and
  - Read all deltas (rectp=0) smaller than or equal to t2 and
  - Read all hist. movements (rectp=2) larger than t2
  - (no time-cha in group-by)**
- Aggregate result to get inventory value at t2
- Query 2:
  - Read all deltas between t1 and t2
  - (with time-cha in group-by)**
- Loop backwards in time over deltas
  - subtract values from corresponding ref-point;

Picture 70: HANA BW non-cumulative handling

## 11.2 Query examples

```

SELECT
  "F"."SID_OPLANT" AS "S___026"
  , "F"."SID_OMATERIAL" AS "S___237"
  , "F"."SID_OBASE_UOM" AS "S___420"
  , "DP"."SID_ORECORDTP" AS "S___020"
  , COUNT( * ) AS "Z___024"
  , SUM ( "F"."RECTOTSTCK" ) AS "Z___7828"
  , SUM ( "F"."ISSTOTSTCK" ) AS "Z___7827"
FROM
  "/BIC/FZDGRNCUM1" "F"
JOIN
  "/BIC/DZDGRNCUM1P" "DP"
ON
  "F" . "KEY_ZDGRNCUM1P"
= "DP" . "DIMID"
WHERE
  ( ( ( (
    "DP"."SID_OREQUID"
    <= 3620661
  ) ) ) ) AND ( ( ( (
    "DP"."SID_ORECORDTP"
    = 1
  ) ) ) ) OR ( ( ( (
    "DP"."SID_ORECORDTP"
    = 2
  AND
    "F"."SID_OCALDAY"
    > 20120530
  ) ) ) ) OR ( ( ( (
    "DP"."SID_ORECORDTP"
    = 0
  AND
    "F"."SID_OCALDAY"
    <= 20120530
  ) ) ) )
  ) ) ) )
GROUP BY
  "F"."SID_OPLANT"
  , "F"."SID_OMATERIAL"
  , "F"."SID_OBASE_UOM"
  , "DP"."SID_ORECORDTP"
WITH HINT( OLAP_PARALLEL_AGGREGATION )
/*$ dmuid=0008I3A4ZC6SDAGK0H2R78V2G */

```

Just get the Initialization  
records  
(sid\_orecordtp = 1)

History larger than t2  
(sid\_orecordtp = 2)

Delta records smaller or equal  
to t2 (sid\_recordtp = 0)

no group by time

Picture 71: 'Reference point' query

```

SELECT
  "F"."SID_OPLANT" AS "S___026"
, "F"."SID_OCALDAY" AS "S___035"
, "F"."SID_OMATERIAL" AS "S___237"
, "F"."SID_OBASE_UOM" AS "S___420"
, COUNT( * ) AS "Z___024"
, SUM ( "F"."RECTOTSTCK" ) AS "Z___7828"
, SUM ( "F"."ISSTOTSTCK" ) AS "Z___7827"
FROM
  "/BIC/FZDGRNCUM1" "F"
JOIN
  "/BIC/DZDGRNCUM1P" "DP"
ON
  "F" . "KEY_ZDGRNCUM1P"
= "DP" . "DIMID"
WHERE
  ( ( ( (
    "F"."SID_OCALDAY"
  BETWEEN 20120201
  AND 20120530
  ) ) AND ( (
    "DP"."SID_ORECORDTP"
  IN (
    0 , 2
  )
  ) ) AND ( (
    "DP"."SID_OREQUID"
  <= 3620661
  ) ) ) )
GROUP BY
  "F"."SID_OPLANT"
, "F"."SID_OCALDAY"
, "F"."SID_OMATERIAL"
, "F"."SID_OBASE_UOM"
ORDER BY
  "S___026"
, "S___237"
, "S___420"
WITH HINT( OLAP_PARALLEL_AGGREGATION )
/*$ dmuid=0008I3A4ZC6SDAGK0H2R78V2G */

```

CALDAY between  
t1=20120201  
and  
t2 = 20120530

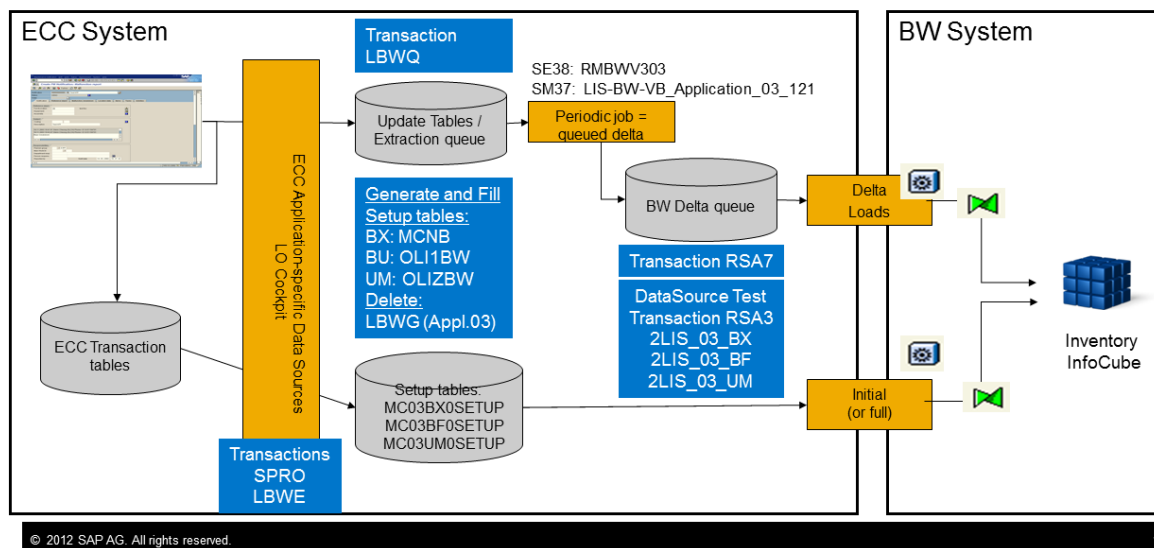
RECORDTP in (0, 2) means  
0: Deltas  
2: Historic Transactions

Group by  
OCALDAY

Picture 72: 'Movements'

## 11.3 Overview BW LO Cockpit MM Inventory Management

### SAP BW LO Cockpit Delta Extraction for MM Inventory Management



Picture 73: Overview BW LO Cockpit

## 11.4 Notes

<http://service.sap.com/sap/support/notes/1548125>

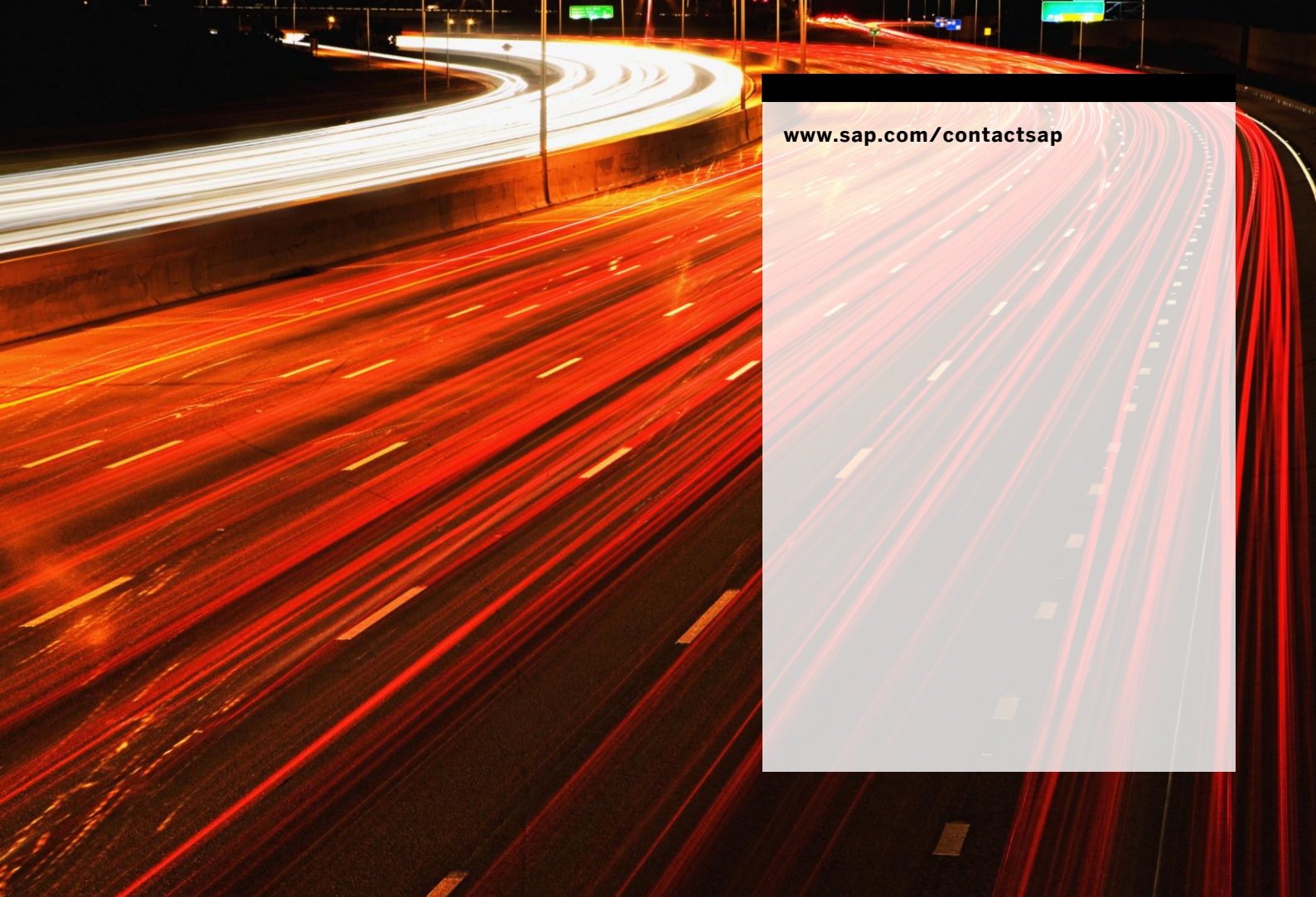
<http://service.sap.com/sap/support/notes/1679974>

<http://service.sap.com/sap/support/notes/1691932>

<http://service.sap.com/sap/support/notes/1683889>

<http://service.sap.com/sap/support/notes/1679099>





[www.sap.com/contactsap](http://www.sap.com/contactsap)



The Best-Run Businesses Run SAP™