

BW Quantity Conversion

- [About refresh on WBP](#)
- [Conversion Unit C_MATNR2 - UOMCMAT2 - Specific rules](#)
- [SKU vs RU \(VKG\)](#)
- [Conversion Unit C_MATNR2 - UOMCMAT2 - Enhancement from Global Filter list](#)
- [Function Module Z_MD_CONVERT_MATERIAL_UNIT3](#)
 - [Function Module Logic](#)

About refresh on WBP

About refresh of unit please check the page below :

[Unit of measure / Quantity unit / UOM -> RSIMPCUST](#)

Conversion Unit C_MATNR2 - UOMCMAT2 - Specific rules

The recursive transformation **TSRF : UOMCMAT2 -> UOMCMAT2 (recursive)** add conversion factors to missing units : KG / LB / TO / VKG / VTN

Program Steps

1. Generate a matrix of unit per LogSys/Material/BuoM

	KG Unit	KG Nom	KG Dén	TO Unit	TO Nom	TO Dén	1KG Unit	1KG Nom	1KG Dén	...
WP1/Material 1/KG	KG	1.000	1.000	TO	1.000.000	1.000				...
WP1/Material 2/KG	KG	1.000	1.000				1KG	1.000	277	...

2. Add KG unit
 - a. Read TO unit if available
 - b. Read LB unit if available
3. Add TO unit
 - a. Read KG unit if available
 - b. Read LB unit if available
4. Add LB unit
 - a. Read KG unit if available
 - b. Read TO unit if available
5. Add VKG unit
 - a. Read by priority VTN > 1KG > 1TN > KGW > TOP > KRE > KG > TO if available
6. Add VTN unit
 - a. Read by priority VKG > 1KG > 1TN > TOP > KGW > KRE > KG > TO if available



Units and Conversion exit

A Unit has only one internal code and is corresponding to one or more external code (country specific display and other reasons ...)

For example :

- KGW (Internal) is corresponding to KAI / KMA / KGR ... (External)
- TOP (Internal) is corresponding to AIT / ... (External)

Only the Internal code need to be check during the process.

Example of conversion bloc.

Legend:

- **GREEN** : Current unit to add, don't change in a global IF.
- **RED** : Unit to check if available, change at each IF/ELSEIF

```

**** Add VTN
IF <FS_MAT_UNT>-unit VTN IS INITIAL.
  I_SOURCE_PACKAGE-UNIT = 'VTN'. Initialize the unit to add
  I_SOURCE_PACKAGE-record = lv_tabix + 1. Change the record number
  IF <FS_MAT_UNT>-unit VKG IS NOT INITIAL. Check if the corresponding Unit is filled in the matrix
  I_SOURCE_PACKAGE-uomz1d = <FS_MAT_UNT>-uomz1d VKG * 1000. Nominator (* factor if necessary)
  I_SOURCE_PACKAGE-uomn1d = <FS_MAT_UNT>-uomn1d VKG. Denominator (* factor if necessary)
  APPEND I_SOURCE_PACKAGE TO SOURCE_PACKAGE. Append in Result Package
  <FS_MAT_UNT>-UNIT VTN = I_SOURCE_PACKAGE-UNIT. Update the matrix
  <FS_MAT_UNT>-uomz1d VTN = I_SOURCE_PACKAGE-uomz1d. Update the matrix
  <FS_MAT_UNT>-uomn1d VTN = I_SOURCE_PACKAGE-uomn1d. Update the matrix
ELSE/ENDIF. Next unit in the priority list or end of process

```

SKU vs RU (VKG)

SKU = Stock-keeping unit, a unique identifier for each distinct product and service that can be purchased / sales in business. It bases on material master

RU = Reporting unit, it is a conversion unit called VKG, which used for displaying as of product 100% and reporting in order to align unit of measure to calculate sub total

For example, material 35077 (H2O2 AG-BATH-350)

At material master (MM03), it shows KG is Base Unit of Measure, which will be SKU

In order to converse to RU (VKG), go to 'Basic Data Text' on 'Basic data 1' tab

Go to 'Units of measure' tab, at row AUn = VKG, it shows calculation between VKG and base unit of measure in this case is KG

It means 1000 KG (SKU) of H2O2 AG-BATH-350 (35077) equals to 350 VKG (RU), which contain 350 KG of H2O2 and mix with something else like water 750 KG for example.

Display Material 35077 (Z700-Chemicals (trade))

Main Data

Descriptions Units of measure Additional EANs Document data Basic data t...

Material 35077 1202 AG-BATH-350

Units of measure/EANs/dimensions

X	AUn	Measu...	<... Y	BUn	Measu...	EAN/UPC	Ct	Au	A	Length
1	KG	kg	<... 1	KG	kg					0,000
350	1KG	kg 100%	<... 1.000	KG	kg					0,000
350	VKG	kg % sale	<... 1.000	KG	kg					0,000
350	KNS	KGH2O2	<... 1.000	KG	kg					0,000
1	TO	t	<... 1.000	KG	kg					0,000
10	1TN	t 100%	<... 28.571	KG	kg					0,000

See below a technical documentation about the routines that load data in UOMCMAT2, including a recursive transformation used to load special units like VKG and VTN.

See below example about the end routine of recursive transformation to load special unit like VKG and VTN:

Conversion Unit C_MATNR2 - UOMCMAT2 - Enhancement from Global Filter list

Dynamic conversion in queries (using Conversion Types) is only working if source and target units are in the same Dimension.

Otherwise a flow of conversions is done with Base UoM as pivot unit, by using factor from T006 table.

When all units of this flow are not provided by SAP Material alternative units, the missing unit must be added to the UOMCMAT2 ADSO.

(in KG = in M and M KG)

Transformation **TRSF: IFS_MD_MATNR2_UOMCMAT2_01 -> UOMCMAT2** is used to add specific Unit conversions for a list of Materials.

The Infosource IFS_MATNR2__UOMCMAT2_0001 allows to create another flow from Master Data C_MATNR2, with a dedicated logic in Transformation.

TRSF: IFS_MD_MATNR2_UOMCMAT2_01 -> UOMCMAT2	0478WJLT5B8AL0TQ3Q5B32291RT8YZOP	Change	
Infosource C_MATNR2 -> UOMCMAT2	IFS_MD_MATNR2_UOMCMAT2_01	Change	InfoSources
TRSF: C_MATNR2 -> IFS_MD_MATNR2_UOMCMAT2_01	00MV0SZJYCQMYZ0T8AG4ZXJMG0TSDIY	Change	
Material (Attributes)	ATTRIBUTES C_MATNR2	Manage	InfoSources

The Materials/units to convert must be populated into the Master Data Global Filter C_GLBFLT, in each BW system to process.

Table: /BIC/PC_GLBFLT

BIC/C_STREAM	BIC/C_RULE	BIC/C_GLBFLT	OBJVERS	CHANGED	BIC/C_DESC	BIC/C_SIGN	BIC/C_OPTION	BIC/C_LOW	BIC/C_HIGH	BIC/C_ACTIVE
C_MATNR2	155895	000	A		Material Alternative Units conversion in Base UoM (UOMCMAT2)	I	EQ	WD1_110	IN	Y
C_MATNR2	155895	001	A		Material Alternative Units conversion in Base UoM (UOMCMAT2)	I	EQ	WD1_110	LB	Y
C_MATNR2	155895	002	A		Material Alternative Units conversion in Base UoM (UOMCMAT2)	I	EQ	WD1_110	M	Y
C_MATNR2	156001	000	A		Material Alternative Units conversion in Base UoM (UOMCMAT2)	I	EQ	WD1_110	IN	Y

C_STREAM : Always value 'C_MATNR2' to select all records during Transformation Start Routine

C_RULE : Material value

C_LOW : Logical system compounding Material

C_SIGN : Value 'I' to input record in UOMCMAT2, value 'E' to delete a record in UOMCMAT2

C_OPTION : Always 'EQ'

C_HIGH : Unit to add for conversion in Base UoM in UOMCMAT2 => Unit value in INTERNAL format (PC => ST, GAL => GLL, "2 => IN2, in => IN....).

C_ACTIVE : If not equal 'Y', the record is not selected during loading

Transformation logic :

Start Routine

Select all records with C_STREAM = 'C_MATNR2' from Master Data Global Filter C_GLBFLT.

Select all records from table /BI0/SUNIT (Unit SID) to populate in End Routine

End Routine

For each Logsys/Material record from source C_MATNR2 (use DTP filter to limit Materials to extract), a loop is done to process each Unit from Global Filter for this Logsys/Material.

The conversion added into UOMCMAT2 is from the Unit selected in C_GLBFLT, into the Base UoM of the Material from C_MATNR2. If the Base UoM is empty (should occur in Dev only !!), the target Unit is forced to 'KG' for conversion.

The conversion is done through the Function Module Z_MD_CONVERT_MATERIAL_UNIT3 which is the one used in BW data flows to convert units properly.

Conversion is working fine in this FM even if Units are in different dimensions.

The result of the conversion becomes the Numerator to add in UOMCMAT2 for the Unit converted.

The Input quantity to convert (1000) becomes the Denominator.

No matter the value to convert, the Factor result Numerator/Denominator will always be the same (at least a value of 1000 for accuracy purposes).

If C_SIGN = 'E' in Master Data Global Filter C_GLBFLT, the RECORDMODE is set to 'D' to delete the relevant record in UOMCMAT2.

Function Module Z_MD_CONVERT_MATERIAL_UNIT3

Material unit conversion in ABAP code can be handled by two Function Modules.

Z_MD_CONVERT_MATERIAL_UNIT retrieving alternative units **without** source system (0LOGSYS), based on ODS Object **ODS_MARM**.

Z_MD_CONVERT_MATERIAL_UNIT3 retrieving alternative units **with** source system (0LOGSYS), based on ODS Object **UOMCMAT2**.

Do not use Z_MD_CONVERT_MATERIAL_UNIT2 anymore in new ABAP code, since it is a previous uncomplete version of the Z_MD_CONVERT_MATERIAL_UNIT3.

This previous version can still be found anyway in a lot of ABAP code, as long as the Materials are not dealing with different alternative units in several source systems.

Function Module Logic

Same logic is used in both Function Modules.

Source unit is first converted into Material Base Unit of Measure, and then result is converted into target unit if different from Base Unit.

Statics variables are used to keep Material already processed across several calls of FM, preventing new DB access to get conversion factors.

=> This is where inconsistency is noticed in Z_MD_CONVERT_MATERIAL_UNIT2 with System not taken into account when checking Material already processed.

(Check reference NGC20220308 for updated code in Z_MD_CONVERT_MATERIAL_UNIT3)

Chronology of process :

1. Convert the input Unit in Basic UoM according to logic below

1. Conversion factors have already been determined for Material in last call
2. Else search for Material conversion factors from buffer table
3. Else determine data from the DB

1. Search first with the last Material read (small buffer)

2. Else try it in all Material processed (large buffer)

3. Else read valid Mat Units from **UOMCMAT2** for the Material/System (Z_MD_CONVERT_MATERIAL_UNIT3), or read valid Mat Units from **ODS_MARM** for the Material (Z_MD_CONVERT_MATERIAL_UNIT).

4. Else read all units from T006 table to get conversion factor from the Material Base UOM into the System International Unit of the same Dimension.

2. Convert the Basic UoM into target unit (if different from Base UoM) according to logic below.

1. Conversion factors have already been determined for Material in last call

2. Else search for Material conversion factors from buffer table

3. Else determine data from the DB

1. Search first with the last Material read (small buffer)

2. Else try it in all Material processed (large buffer)

3. Else read valid Mat Units from **UOMCMAT2** for the Material/System (Z_MD_CONVERT_MATERIAL_UNIT3), or read valid Mat Units from **ODS_MARM** for the Material (Z_MD_CONVERT_MATERIAL_UNIT).

4. Else read all units from T006 table to get conversion factor from the Material Base UOM into the System International Unit of the same Dimension.

Rounded Factors :

When the target unit is not the Basic UoM, and the conversion is done through the Basic UoM, it leads sometimes to a rounded factor used to convert from source unit to target unit.

This can raise differences between conversion done with the Function Module Z_MD_CONVERT_MATERIAL_UNIT3, and conversion done directly in a query with conversion types (RSUOM) where factor is not rounded.