

# Winshuttle Wrapper RFMs – WFM 20.2

## 1. /WINSHTLQ/BPAR\_C\_A\_VAT\_NUMBER

- ✓ **PURPOSE:** Check Additional Data of Business Partner
- ✓ **DESCRIPTION:** This WFM is a wrapper of 'BPAR\_C\_A\_VAT\_NUMBER' FM which is used to check the VAT number specific to the country entered.

### The function module interface

- ✓ **Import Parameters**
  1. **COUNTRY:** Enter country of Business partner
  2. **VAT\_NUMBER:** Enter VAT Registration number
- ✓ **Export Parameters**
  1. **MESSAGE:** Displays error message if any.
- ✓ **Exceptions**
  1. **ERROR:** This exception occurs in case of any error.
  2. **NO\_VAT\_NUMBER\_IN\_EG:** This exception occurs if the VAT number entered is invalid.
  3. **NOT\_AUTHORIZED\_MSG\_GETDETAIL:** This exception occurs if the user is unauthorized to run BAPI\_MESSAGE\_GETDETAIL FM.
  4. **NOT\_AUTH\_BPAR\_C\_A\_VAT\_NUMBER:** This exception occurs if the user is unauthorized to run /WINSHTLQ/BPAR\_C\_A\_VAT\_NUMBER FM.
  5. **NO\_RELEASE\_FOUND:** This exception occurs if no SAP BASIS release is found in the SAP system.

### Example

Enter Input as below:

Test for function group	/WINSHTLQ/FUGR_BPAR_VAT
Function module	/WINSHTLQ/BPAR_C_A_VAT_NUMBER
Uppercase/Lowercase	<input type="checkbox"/>
RFC target sys:	
Import parameters	Value
COUNTRY	DE
VAT_NUMBER	[123456789]

If the entered VAT number is invalid, the error message below appears.

```
Test for function group      /WINSHTLQ/FUGR_BPAR_VAT
Function module             /WINSHTLQ/BPAR_C_A_VAT_NUMBER
Uppercase/Lowercase        
Runtime:                    0 Microseconds
RFC target sys:            
```

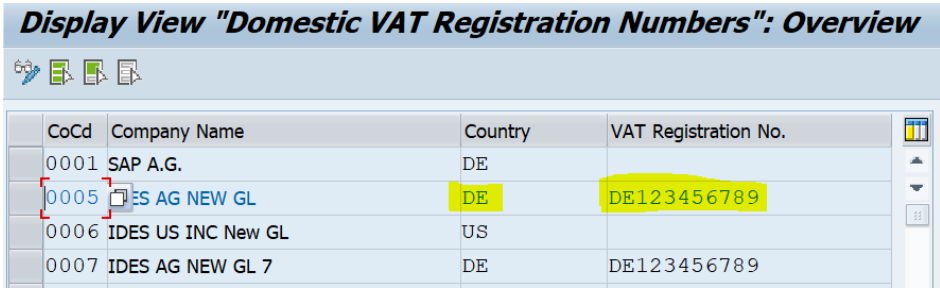
Import parameters	Value
COUNTRY	DE
VAT_NUMBER	123456789

Export parameters	Value
MESSAGE	ISO code 12 is not correct in the VAT registration number

However, when the correct VAT number is entered for the country as below, no error message is displayed.

**Display View "Domestic VAT Registration Numbers": Overview**



CoCd	Company Name	Country	VAT Registration No.
0001	SAP A.G.	DE	
0005	IDES AG NEW GL	DE	DE123456789
0006	IDES US INC New GL	US	
0007	IDES AG NEW GL 7	DE	DE123456789

## 2. /WINSHTLQ/BANK\_ACCOUNT\_CHECK

- ✓ **PURPOSE:** Check of Bank Account Number
- ✓ **DESCRIPTION:** This WFM is a wrapper of 'BANK\_ACCOUNT\_CHECK' FM which is used to check the entered Bank Account Number along with Bank Control Key, Bank country and Bank number.

### The function module interface

- ✓ **Import Parameters**
  1. **BANK\_ACCOUNT:** Enter Bank account number
  2. **BANK\_CONTROL\_KEY:** Enter Bank Control Key
  3. **BANK\_COUNTRY:** Enter Bank Country Key
  4. **BANK\_NUMBER:** Enter Bank Key
  
- ✓ **Export Parameters**
  1. **MESSAGE:** Displays error message if any.
  
- ✓ **Exceptions**
  1. **NOT\_VALID:** This exception occurs if entered Bank account number is invalid.
  2. **NOT\_AUTHORIZED\_MSG\_GETDETAIL:** This exception occurs if the user is unauthorized to run BAPI\_MESSAGE\_GETDETAIL FM.
  3. **NOT\_AUTH\_BANK\_ACCOUNT\_CHECK:** This exception occurs if the user is unauthorized to run /WINSHTLQ/BANK\_ACCOUNT\_CHECK FM.
  4. **NO\_RELEASE\_FOUND:** This exception occurs if no SAP BASIS release is found in the SAP system.

### Example

Enter Input as below:

Import parameters	Value
BANK_ACCOUNT	
BANK_CONTROL_KEY	
BANK_COUNTRY	DE
BANK_NUMBER	123456

If the entered input details are valid, no error message is displayed. However, if the entered Bank details are invalid, the error message is displayed.

Import parameters	Value
BANK_ACCOUNT BANK_CONTROL_KEY BANK_COUNTRY BANK_NUMBER	DE 123456

Export parameters	Value
MESSAGE	Enter a valid bank account number

### 3. /WINSHTLQ/BUSINESS\_NUMBER\_CHEK

- ✓ **PURPOSE:** Check of international location number (ILN)
- ✓ **DESCRIPTION:** This function module is a wrapper of 'BUSINESS\_NUMBER\_CHECK' and checks the international location number (ILN) for each country (controlled by the international license plate number).

The BUSINESS\_NUMBER\_CHECK function module checks the international location number (ILN) for each country (controlled by the international license plate number). The ILN consists of 3 parts: positions 1 to 7 (this part is described as the base number in the case of an ILN of type 2), positions 8 to 12, and the check digit.

If the check has negative results (the number is invalid), the NOT\_VALID exception is triggered. A corresponding message is issued at the same time. If the number is valid, nothing occurs.

For South Africa, the system carries out the check according to a unique algorithm. Checks do not take place for Belgium, Italy, Spain, or the Netherlands.

#### The function module interface

- ✓ **Import Parameters**
  1. **BUSINESS\_NUMBER:** International location number (part 1)
  2. **CHECK\_NUMBER:** Check digit for international location number
  3. **COUNTRY:** Country Key
  4. **SUPPLEMENT\_NUMBER:** International location number (Part 2)
- ✓ **Export Parameters**
  1. **MESSAGE:** Displays error message if any.
- ✓ **Exceptions**
  1. **NOT\_VALID:** This exception occurs if entered Business number is invalid.
  2. **NOT\_AUTHORIZED\_MSG\_GETDETAIL:** This exception occurs if the user is unauthorized to run BAPI\_MESSAGE\_GETDETAIL FM.
  3. **NOT\_AUTH\_BUIS\_NUMBER\_CHEK:** This exception occurs if the user is unauthorized to run /WINSHTLQ/BUSINESS\_NUMBER\_CHEK FM.
  4. **NO\_RELEASE\_FOUND:** This exception occurs if no SAP BASIS release is found in the SAP system.

## Example

Enter Input as below:

Import parameters	Value
BUSINESS_NUMBER	0001233
CHECK_NUMBER	0
COUNTRY	
SUPPLEMENT_NUMBER	00000

If the entered details are invalid, the error message below is displayed.

Import parameters	Value
BUSINESS_NUMBER	0001233
CHECK_NUMBER	0
COUNTRY	
SUPPLEMENT_NUMBER	00000

Export parameters	Value
MESSAGE	International location number 0001233 00000 0 is invalid

## 4. /WINSHTLQ/EU\_TAX\_NUMBER\_CHECK

- ✓ **PURPOSE:** Check of VAT registration number (EU tax number)
- ✓ **DESCRIPTION:** This function module is a wrapper of 'EU\_TAX\_NUMBER\_CHECK' and checks the VAT registration number (EU tax number) for each country.

The check rules are stored in table T005 in the field PRUIN. The length of the VAT number, which is important for this check, is stored in field UINLN.

If the check comes across an invalid format, an incorrect country key, or an incorrect check digit, the exception NOT\_VALID is triggered, and an error message is sent. If the check runs successfully, you will not receive a message.

### The function module interface

- ✓ **Import Parameters**
  1. **COUNTRY:** Country Key
  2. **EU\_TAX\_NUMBER:** VAT registration number (EU tax number)
- ✓ **Export Parameters**
  1. **MESSAGE:** Displays error message if any.
- ✓ **Exceptions**
  1. **NOT\_VALID:** This exception occurs if entered VAT number is invalid.
  2. **NOT\_AUTHORIZED\_MSG\_GETDETAIL:** This exception occurs if the user is unauthorized to run BAPI\_MESSAGE\_GETDETAIL FM.
  3. **NOT\_AUTHORIZED\_EU\_TAX\_NUM\_CHK:** This exception occurs if the user is unauthorized to run /WINSHTLQ/EU\_TAX\_NUMBER\_CHECK FM.
  4. **NO\_RELEASE\_FOUND:** This exception occurs if no SAP BASIS release is found in the SAP system.

## Example

Enter Input as below:

Import parameters	Value
COUNTRY EU_TAX_NUMBER	DE HY1212YYSVS

No such Tax number exists, so the error message is displayed.

Import parameters	Value
COUNTRY EU_TAX_NUMBER	DE HY1212YYSVS

Export parameters	Value
MESSAGE	ISO code HY is not correct in the VAT registration number



## 5. /WINSHTLQ/TAX\_NUMBER\_CHECK

- ✓ **PURPOSE:** Check Tax Numbers
- ✓ **DESCRIPTION:** This function module is a wrapper of 'TAX\_NUMBER\_CHECK' and checks the fields Tax Number 1 and Tax Number 2 using the rules defined in table T005.

The following additional checks are made against the motor vehicle country key:

Country	Code	Tax Number 1	Tax Number 2
Argentina	RA	x	
Belgium	B	x	
Brazil	BRA	x	x
Denmark	DK		x
France	F	x	
Italy	I	x	x
Norway	N		x
Portugal	P	x	
Spain	E	x	
Sweden	S		x

### The function module interface

- ✓ **Import Parameters**
  1. **COUNTRY:** Country Key
  2. **NATURAL\_PERSON\_FLAG:** Natural Person
  3. **REGION:** Region
  4. **STKZU:** Country-Specific Check
  5. **TAX\_CODE\_1:** Tax Number 1
  6. **TAX\_CODE\_2:** Tax Number 2
  7. **TYPE\_OF\_TAX\_CODE\_1:** Tax Number Type
  8. **TAX\_CODE\_3:** Tax Number 3
  9. **TAX\_CODE\_4:** Tax Number 4
- ✓ **Export Parameters**
  1. **MESSAGE:** Displays error message if any.
- ✓ **Exceptions**
  1. **NOT\_VALID:** This exception occurs if entered Natural number is invalid.
  2. **NOT\_AUTHORIZED\_MSG\_GETDETAIL:** This exception occurs if the user is unauthorized to run BAPI\_MESSAGE\_GETDETAIL FM.
  3. **NOT\_AUTHORIZED\_TAX\_NUM\_CHK:** This exception occurs if the user is unauthorized to run /WINSHTLQ/TAX\_NUMBER\_CHECK FM.
  4. **NO\_RELEASE\_FOUND:** This exception occurs if no SAP BASIS release is found in the SAP system.
  5. **DIFFERENT\_FPRCD:** This exception occurs if Different Provincial Tax Codes are entered in STCD1 and STCD2 fields

## Example

Enter Input as below:

**Test Function Module: Initial Screen**

Debugging Test data directory

Test for function group /WINSHTLQ/FUGR\_ETNC  
Function module /WINSHTLQ/TAX\_NUMBER\_CHECK  
Uppercase/Lowercase

RFC target sys: \_\_\_\_\_

Import parameters	Value
COUNTRY	USA
NATURAL_PERSON_FLAG	X
REGION	IND
STKZU	
TAX_CODE_1	34
TAX_CODE_2	
TYPE_OF_TAX_CODE_1	
TAX_CODE_3	
TAX_CODE_4	

If the entered details are invalid, a message is displayed. If they are valid, you will not receive an error message.

**Test Function Module: Result Screen**

Test for function group /WINSHTLQ/FUGR\_ETNC  
Function module /WINSHTLQ/TAX\_NUMBER\_CHECK  
Uppercase/Lowercase

Runtime: 0 Microseconds

RFC target sys: \_\_\_\_\_

Import parameters	Value
COUNTRY	USA
NATURAL_PERSON_FLAG	X
REGION	IND
STKZU	
TAX_CODE_1	34
TAX_CODE_2	
TYPE_OF_TAX_CODE_1	
TAX_CODE_3	
TAX_CODE_4	

Export parameters	Value
MESSAGE	Country key USA is not defined

## 6. /WINSHTLQ/POSTAL\_CODE\_CHECK

- ✓ **PURPOSE:** Check of postal code and region
- ✓ **DESCRIPTION:** This function module is a wrapper of 'POSTAL\_CODE\_CHECK' and checks the entered postal code and region.

### The function module interface

- ✓ **Import Parameters**
  1. **COUNTRY:** Country Key
  2. **ONE\_TIME\_ACCOUNT:** Indicator for one-time transactions (one-time account)
  3. **POSTAL\_CODE:** Postal code
  4. **POSTAL\_CODE\_PO\_BOX:** P.O. Box postal code
  5. **PO\_BOX:** PO Box
  6. **REGION:** Region (federal state, state, province, county)
- ✓ **Export Parameters**
  1. **POSTAL\_CODE:** Postal code
  2. **EV\_POSTAL\_CODE:** Postal code
  3. **EV\_POSTAL\_CODE\_PO\_BOX:** P.O. Box Postal Code
  4. **MESSAGE:** Displays error message if any.
- ✓ **Exceptions**
  1. **NOT\_VALID:** This exception occurs if entered postal code and region is invalid.
  2. **NOT\_AUTHORIZED\_MSG\_GETDETAIL:** This exception occurs if the user is unauthorized to run BAPI\_MESSAGE\_GETDETAIL FM.
  3. **NOT\_AUTH\_POSTAL\_CODE\_CHECK:** This exception occurs if the user is unauthorized to run /WINSHTLQ/POSTAL\_CODE\_CHECK FM.
  4. **NO\_RELEASE\_FOUND:** This exception occurs if no SAP BASIS release is found in the SAP system.

### Example

Enter Input as below:

Import parameters	Value
COUNTRY	US
ONE_TIME_ACCOUNT	
POSTAL_CODE	80216
POSTAL_CODE_PO_BOX	-----
PO_BOX	1030
REGION	CO

If the entered details are valid, no error message will be displayed. Otherwise, an error message will be displayed.

Import parameters	Value
COUNTRY	US
ONE_TIME_ACCOUNT	
POSTAL_CODE	80216
POSTAL_CODE_PO_BOX	
PO_BOX	1030
REGION	CO

Export parameters	Value
POSTAL_CODE	
EV_POSTAL_CODE	
EV_POSTAL_CODE_PO_BOX	
MESSAGE	PostCd/PObox from position 1 to 5 can only contain numbers

## 7. /WINSHTLQ/RPC\_CALCULATE\_PRICES

- ✓ **PURPOSE:** WFM to Request Calculations and Adjust Requested Calculation
- ✓ **DESCRIPTION:** This BAPI (wrapper of 'BAPI\_RPC\_CALCULATE\_PRICES') or the CalculatePrices method in the RetailPricing business object, facilitates communication between the sales price calculation function of the SAP System and one or more external systems.

**You can use this BAPI to request and adjust calculations in external systems:**

- **Requesting a Pricing Item**

Every pricing item required must be requested precisely once using this BAPI. Requesting a pricing item causes the system to suggest all the relevant pricing data (such as markup, margins and sales prices, for example).

- **Adjusting a Requested Pricing Item**

If the pricing items requested do not meet your expectations - for example, if the sales price suggested by the system is either too high or too low, the pricing item can be adjusted using this BAPI.

Depending on whether a calculation is to be requested or whether a requested calculation is to be adjusted, the interface tables CALCITEMIN and CALCITEMINX can be filled in different ways. You will find further information on this in the documentation for the individual tables.

- **Results of the Calculation**

Each time the BAPI is called up, the result of the calculation is entered in the CALCITEMOUT export table. Any errors that may have occurred are displayed in the RETURN export table. You will find further information in the interface documentation for the relevant tables.

- **Customer-Specific Enhancements**

Further fields that are not contained in the CALCITEMIN, CALCITEMINX or CALCITEMOUT interface tables can be entered using the enhancement tables EXTENSIONIN and EXTENSIONOUT and the BAPI. You can find further information on this in the interface documentation for the individual tables.

## The function module interface

- ✓ **Import Parameters**

- **RPCCONTROLIN:** Control Sales Price Calculation for BAPI

- ✓ **Tables**

1. **CALCITEMIN:** BAPI Structure for Pricing Item (Import)
2. **CALCITEMINX:** Change BAPI Reference Structure for Structure BAPICALCITEMIN
3. **EXTENSIONIN:** Ref. structure for BAPI parameter ExtensionIn/ExtensionOut
4. **CALCITEMOUT:** BAPI Structure for Pricing Item (Export)
5. **EXTENSIONOUT:** Ref. structure for BAPI parameter ExtensionIn/ExtensionOut
6. **RETURN:** Return Parameter

- ✓ **Exceptions**

1. **NOT\_AUTHORIZED\_CALC\_PRICE:** This exception occurs if the user is unauthorized to run /WINSHTLQ/RPC\_CALCULATE\_PRICES FM.
2. **NOT\_AUTHORIZED\_BAPI\_CALC\_PRICE:** This exception occurs if the user is unauthorized to run BAPI\_RPC\_CALCULATE\_PRICES FM.
3. **NO\_RELEASE\_FOUND:** This exception occurs if no SAP BASIS release is found in the SAP system.

### Example

Enter Input as below:

- **Table: CALCITEMIN**

```
SALES_ORG = '0001'.  
DISTR_CHAN = '01'.  
MATERIAL = '100-100'.  
SP_VALID_FROM = '29.01.2018'.  
SP_VALID_TO = '29.01.2018'.  
SALES_CURRENCY = 'AUD'.  
FINAL_PRICE = '10'.  
SP_MARKED_FOR_ACTIVATION = 'X'.
```

- **Table: CALCITEMINX**

SALES\_ORG = '0001'.  
 DISTR\_CHAN = '01'.  
 MATERIAL = '100-100'.  
 FINAL\_PRICE = 'X'.  
 SP\_MARKED\_FOR\_ACTIVATION = 'X'.

**Output:**

**Return Table:**

1 Entry		
T	ID	NUM MESSAGE
		000 Pricing Document 0000001082 was created successfully

The created document can be seen in t-code VKP7.

**Display Pricing Document**

Purchasing Sales Log Change List Field Copy Price Field

Document Number 1082  
 Created by ECC6DEV  
 Created on 17.04.2018

Material WINSHUTTLE-01 test Scrn 1 / 1  
 EAN/UPC  
 Valid From 17.04.2018  
 Valid To 31.12.9999

Sta	Distr.	Uni	Vendor	DCrnc	Basic PP	Local	PP (nt/nt)	Mark-up(%)	Pl. MU (%)	Gross SP	Fin. pr.	Margin(gr)	Fin.pr.old	Mrg(g/c)
C	0001 01	PC	1000	MXN	0,00	EUR	0,00	999999,999	0,000	0,00	10,00	0,000	10,00	0,000

## 8. /WINSHTLQ/TRN\_TAX\_TXJCD\_TABLE

- ✓ **PURPOSE:** WFM for Tax Jurisdiction Code Determination
- ✓ **DESCRIPTION:** This function is intended for all entities (including most R/3 master records such as plant, customer, vendor, cost center, asset, etc.) that use the Central Address Management (CAM) to store their addresses.

### This function has two features:

- If a jurisdiction code is not specified, this function determines a unique jurisdiction code from the given address: Country, state, Zip code, and possibly city and/or county.
- If a jurisdiction code is specified, the function checks that this jurisdiction code is valid. To check if a jurisdiction code is valid, the country tax procedure needs to be determined to establish whether an external tax system is being used. If an external system is in use, the specified jurisdiction code will be checked against the specified address (state, zip code, for example). Alternatively, the specified jurisdiction code will be checked against table TTXJ and the corresponding jurisdiction code text will also be returned.

**Tax Jurisdiction Code:** Location identification for tax purposes. The combination of country, state, county, city, and zip code produces a unique jurisdiction code. The jurisdiction code is used by most US tax packages to simplify the US tax calculation process in which the place of origin and destination are of major significance. For this reason, a jurisdiction code usually belongs to an entity.

### The function module interface

#### ✓ Import Parameters

- **IM\_COUNTRY:** Country Key
- **IM\_REGION:** Region (State, Province, County)
- **IM\_ZIPCODE:** City postal code
- **IM\_CITY:** City
- **IM\_COUNTY:** District

#### ✓ Tables

1. **T\_JURTAB:** US tax interface: Communication structure for TXJCD determined.

#### ✓ Exceptions

1. **NOT\_AUTHORIZED\_TRN\_TAX\_TXJCD:** This exception occurs if the user is unauthorized to run /WINSHTLQ/TRN\_TAX\_TXJCD\_TABLE FM.
2. **NO\_RELEASE\_FOUND:** This exception occurs if no SAP BASIS release is found in the SAP system.

### Example

Enter Input as below:

Import parameters	Value
IM_COUNTRY	US
IM_REGION	PA
IM_ZIPCODE	19115
IM_CITY	PHILADELPHIA
IM_COUNTY	

Tables	Value
T_JURTAB	0 Entries

**Output:**

Import parameters	Value
IM_COUNTRY	US
IM_REGION	PA
IM_ZIPCODE	19115
IM_CITY	PHILADELPHIA
IM_COUNTY	

Tables	Value
T_JURTAB	0 Entries
Result:	79 Entries

79 Entries										
COU	STA	COUNTY	CITY	ZIPCODE	TX	TX	TX	TX	TXJCD	0
					00	00	00	00	100000000	
					00	00	00	00	101110000	
					00	00	00	00	101110001	
					00	00	00	00	AK0000000	
					00	00	00	00	AL0000000	
					00	00	00	00	AR0000000	
					00	00	00	00	AZ0000000	
					00	00	00	00	CA0000000	
					00	00	00	00	CA0011000	
					00	00	00	00	CA0012000	
					00	00	00	00	CA0013000	
					00	00	00	00	CA0014000	
					00	00	00	00	CA0015000	
					00	00	00	00	CA0016000	
					00	00	00	00	CA0017000	
					00	00	00	00	CO0000000	
					00	00	00	00	CT0000000	
					00	00	00	00	DC0000000	
					00	00	00	00	DE0000000	
					00	00	00	00	FL0000000	
					00	00	00	00	FL0011000	
					00	00	00	00	FL0012000	
					00	00	00	00	FL0013000	



## 9. /WINSHTLQ/SD\_CUST\_MAINTAIN\_ALL

- ✓ **PURPOSE:** Maintain Customer Master data
- ✓ **DESCRIPTION:** This FM is a wrapper of 'SD\_CUSTOMER\_MAINTAIN\_ALL' and is used to maintain General, Sales and Company code data for Customers.

### The function module interface

#### ✓ Import Parameters

- **I\_KNA1:** Maintain General data in Customer Master
- **I\_KNB1:** Maintain Customer Master (Company Code) data
- **I\_KNVV:** Maintain Customer Master Sales Data
- **I\_BAPIADDR1:** BAPI Reference Structure for Addresses (Org./Company)
- **I\_BAPIADDR2:** BAPI reference structure for addresses (nat. persons)
- **I\_MAINTAIN\_ADDRESS\_BY\_KNA1:** This flag should be 'X' if address is maintained in I\_KNA1 table
- **I\_KNB1\_REFERENCE**
- **I\_FORCE\_EXTERNAL\_NUMBER\_RANGE:**
- **I\_NO\_BANK\_MASTER\_UPDATE**
- **I\_CUSTOMER\_IS\_CONSUMER**
- **I\_RAISE\_NO\_BTE**
- **PI\_POSTFLAG**
- **PI\_CAM\_CHANGED**
- **PI\_ADD\_ON\_DATA:** Add-On Data Changed in Customer Master
- **I\_FROM\_CUSTOMERMMASTER**

#### ✓ Tables

1. **T\_JURTAB:** US tax interface: Communication structure for TXJCD determination.

#### ✓ Exceptions:

1. **NOT\_AUTHORIZED\_TRN\_TAX\_TXJCD:** This exception occurs if the user is unauthorized to run /WINSHTLQ/TRN\_TAX\_TXJCD\_TABLE FM.
2. **NO\_RELEASE\_FOUND:** This exception occurs if no SAP BASIS release is found in the SAP system.

### Example

**Case 1: To Maintain General data in Customer Master, Enter input as below**

**Exporting Parameters:**

I_MAINTAIN_ADDRESS_BY_KNA1	X
PI_POSTFLAG	X

Structure: I_KNA1	Values
I_KNA1-KUNNR	A1237
I_KNA1-MANDT	800
I_KNA1-LAND1	DE

I_KNA1-NAME1	TEST 1
I_KNA1-ORT01	SPRING
I_KNA1-PSTLZ	80558
I_KNA1-REGIO	11
I_KNA1-SORTL	US CLIENT
I_KNA1-STRAS	1 MAIN STREET
I_KNA1-MCOD3	GRAZ
I_KNA1-KTOKD	0160
I_KNA1-SPRAS	EN
I_KNA1-LZONE	0000000001
I_KNA1-NAME2	Kumar
I_KNA1-NAME3	Verma
I_KNA1-NAME4	Sharma
I_KNA1-MCOD1	GRAZ
I_KNA1-MCOD2	GRAZ
I_KNA1-PFACH	12
I_KNA1-PSTL2	SW1A 2SW
I_KNA1-BEGRU	0004
I_KNA1-STCD1	3456-56-789
I_KNA1-STCEG	DE-12345679

<b>Structure: I_BAPIADDR1</b>	<b>Values</b>
I_BAPIADDR1 -POSTL_COD1	80558
I_BAPIADDR1- STR_SUPPL1	STREET 1
I_BAPIADDR1- STR_SUPPL2	STREET 2
I_BAPIADDR1- STR_SUPPL3	STREET 3
I_BAPIADDR1-LOCATION	STREET 4
I_BAPIADDR1-TEL1_NUMBR	(215) 665 45
I_BAPIADDR1- FAX_NUMBER	2012020202
I_BAPIADDR1- EMAIL	Ramkumar4221@gmail.com
I_BAPIADDR1- COMM_TYPE	RML
I_BAPIADDR1- NAME	RUMI
I_BAPIADDR1- NAME_2	
I_BAPIADDR1- NAME_3	
I_BAPIADDR1- NAME_4	
I_BAPIADDR1- SORT1	Search term 1
I_BAPIADDR1- SORT2	Search term 2
I_BAPIADDR1- CITY	SPRING
I_BAPIADDR1- COUNTRY	DE
I_BAPIADDR1- LANGU	EN

Data saved successfully.

**Output:**

**Change Customer: General Data**

Other Customer Additional Data, Empties Additional Data, DSD

Customer A1237 TEST 1 SPRING

Address Control Data Payment Transactions Marketing Unloading Points Export

Preview Internat. versions

**Name**

Title [Dropdown menu highlighted in yellow]

Name TEST 1

**Search Terms**

Search term 1/2 US CLIENT

**Street Address**

Street/House number 1 MAIN STREET

Postal Code/City 80558 SPRING

Country DE Germany Region 11

Time zone CET

**Case 2: To Maintain data of Customer Company Code, Enter input as below**

**Input values as below:**

**Exporting Parameters:**

I_MAINTAIN_ADDRESS_BY_KNA1	X
PI_POSTFLAG	X

Structure: I_KNB1	Values
I_KNB1-BUKRS	1000
I_KNB1-AKONT	125000
I_KNB1-ZTERM	0001
I_KNB1-TOGRU	1000
I_KNB1-XZVER	X
I_KNB1-LOCKB	LCKBX
I_KNB1-BUSAB	D1
I_KNB1-KUNNR	A1237

Data saved successfully.

**Output:**

**Change Customer: Company Code Data**

Other Customer    General Data    Additional Data, Empties    Additional Data, DSD

Customer: A1237    TEST 1    SPRING

Company Code: 1000    IDES AG

Account Management    Payment Transactions    Correspondence    Insurance    Withholding Tax

**Accounting information**

Recon. account	125000	Sort key	
Head office		Preference ind.	
Authorization		Cash mgmt group	
Release group		Value adjustment	

**Case 3: To Maintain data in Customer Sales Data, Enter input as below**

**Input values as below:**

**Exporting Parameters:**

I_MAINTAIN_ADDRESS_BY_KNA1	X
PI_POSTFLAG	X

Structure: I_KNVV	Values
I_KNVV-VKORG	1000
I_KNVV-VTWEG	10
I_KNVV-SPART	00
I_KNVV-VKBUR	1000
I_KNVV-KLABC	
I_KNVV-PLTYP	
I_KNVV-VERSG	
I_KNVV-LPRIO	01
I_KNVV-VSBED	01
I_KNVV-KZTLF	
I_KNVV-ANTLF	9
I_KNVV-BOKRE	
I_KNVV-PERFK	
I_KNVV-PERRL	
I_KNVV-INCO1	CFR
I_KNVV-INCO2	HAMBURG
I_KNVV-ZTERM	ZB01
I_KNVV-KTGRD	
I_KNVV-ERNAM	ECC6DEV
I_KNVV-ERDAT	10.10.2018

I_KNWW-KUNNR	A1237
--------------	-------

Table: T_XKNVI	Values
T_XKNVI-KUNNR	A1237
T_XKNVI-TAXKD	1
T_XKNVI-ALAND	DE
T_XKNVI-TATYP	MWST

Data saved successfully.

**Output:**

**Change Customer: Sales Area Data**

Other Customer    General Data    Company Code Data    Additional Data, Empties    Additional Data, DSD

Customer: A1237    TEST 1    SPRING

Sales Org.: 1000    Germany Frankfurt

Distr. Channel: 10    Final customer sales

Division: 00    Cross-division

Sales    Shipping    **Billing Documents**    Partner Functions

**Billing document**

Subs. invoice processing     Rebate     Price determin.

Invoicing dates: [ ]

InvoicingListDates: [ ]

**Delivery and payment terms**

Incoterms: CFR    HAMBURG

Terms of payment: ZB01    14 Days 3%, 30/2%, 45 net    Paym.guar.proc. [ ]

Credit ctrl area: [ ]

## 10. /WINSHTLQ/TRN\_CREATE\_TEXT

- ✓ **PURPOSE:** To maintain SO10 Long text
- ✓ **DESCRIPTION:** This function is used to maintain SO10 long texts.

### The function module interface

#### ✓ Import Parameters

1. **FID:** TextID
2. **FLANGUAGE:** Language Key
3. **FNAME:** Name
4. **FOBJECT:** Texts: Application Object
5. **SAVE\_DIRECT:** Default value is 'X'. This is set to 'X' for saving the long text.
6. **TEXT:** Long text
7. **FFORMAT:** Default value is '\*'. The tag column contains format keys which define the output formatting of the text or initiate control commands.

- A format key defines the beginning of a new paragraph and its formatting. All text lines directly after it which have blanks in this field belong to this paragraph. These lines are handled as continuous text lines and are always formatted by the SAP script editor so that as many words as possible fit into an editor line. Blanks between words are considered. The editor formatting is not the same as the output formatting.
- The output formatting of a text is not visible until the text is output to a printer or displayed on the screen. Here, the format definitions of the paragraph format are evaluated.
  - The format keys possible and their respective meanings are defined in styles or forms. If a style or form is assigned to a text module you can use the paragraph formats defined there to format your text. Format keys which can be defined by the user can consist of one or two characters. Only the letters A to Z and the numbers 0 to 9 are allowed. The paragraph format must always begin with a letter. If a format key is not contained in the assigned style or form, the default paragraph defined there is used instead.

Some format keys are defined by SAP script. They have a predefined meaning and can be used in all texts:

#### \* Default paragraph

The format definitions which are specified for the paragraph defined in the assigned style or form as the default paragraph are used for the output formatting of the paragraph involved.

#### / New line

The subsequent text is written to a new line during output formatting. The formatting attributes of the last paragraph format are retained.

### **/: SAP script control command**

The characters contained in the actual text line are not output as text but are regarded as a control command. They are not interpreted or executed until output formatting of the text. Control commands must always fit into a line fully. Subsequent lines are not allowed.

### **/\* Comment line**

This line is not considered during output formatting of the text.

### **= Extended line**

This line is not subject to the line formatting in the SAP script editor. The text contained in this line is appended directly to the character of the preceding text line which was output last. If this is not required, there must be at least one blank at the beginning of the extended line.

### **/= Like extended line**

But the subsequent text is placed in a new line during output formatting.

### **( Raw line**

The subsequent editor line is not interpreted by the SAP script composer during output formatting. This means that character formats, symbols, tab characters, masking characters, or hypertext links which may be contained in this line are not evaluated and are therefore passed unchanged to the output device. The text contained in this line is also appended directly to the character of the preceding text line which was output last. If this is not required, there must be at least one blank at the beginning of the extended line.

### **/( Like raw line**

But the subsequent text is placed in a new line during output formatting.

### **>x Fixed line**

The line is not ready for input in the SAP script editor. It can also not be deleted or separated. You can only create fixed lines with a program. You can therefore give a text a fixed structure, for example, which cannot be changed by the user. You can use any number or letter for the 'x'. You can therefore separate different sub-headings, for example. If several fixed lines occur consecutively with the same indicator, they are regarded as a unit by the SAP script editor. It is not possible to insert anything between these lines in the editor.

In the case of fixed lines, SAP script print formatting interprets the first two characters of the line as a paragraph format for formatting. You therefore need to enter the required paragraph format or blank here.

#### **✓ Tables**

1. **FLINES:** Text lines which needs to be updated in Long text.
2. **MESSAGES:** Get return messages

3. **WINOPTIONS:** RFC Table Read: Select Options / WHERE Clause (not in use as of March 2021).

✓ **Exceptions:**

1. **NOT\_AUTHORIZED\_RFC\_CREATE\_TEXT:** This exception occurs if the user is unauthorized to run /WINSHTLQ/TRN\_CREATE\_TEXT FM.
2. **DATA\_IS\_IN\_FLINES\_N\_TEXT\_BOTH:** Either 'TEXT' or 'FLINES' can have input once, not both. This exception occurs if Long text is entered on parameter 'TEXT' or in Table 'FLINES'.

## Examples

Enter Input as below:

**Test Function Module: Initial Screen**

Debugging Test data directory

Test for function group /WINSHTLQ/TRN\_FUGR CRT\_TXT  
Function module /WINSHTLQ/TRN\_CREATE\_TEXT  
Uppercase/Lowercase

RFC target sys: \_\_\_\_\_

Import parameters	Value
FID	ST
FLANGUAGE	EN
FNAME	TEXT
FOBJECT	TEXT
SAVE_DIRECT	X
FFORMAT	*
TEXT	

Long Text Table:

**Structure Editor: Change FLINES from Entry**

Column Entry New Line Double L



4 Entries

TD	TDLINE
[*]	THIS IS LONG TEXT LINE 1
*	THIS IS LONG TEXT LINE 2
*	THIS IS LONG TEXT LINE 3
*	THIS IS LONG TEXT LINE 4







This Long text will be saved in SO10 Text.

**Standard Text: Request**




 

Standard Text

Text Name	TEXT	
Text ID	ST	Standard text
Language	EN	

 Display  Change  Create

**Display Standard text: TEXT Language EN**

	.....1.....2.....3.....4
*	THIS IS LONG TEXT LINE 1
*	THIS IS LONG TEXT LINE 2
*	THIS IS LONG TEXT LINE 3
*	THIS IS LONG TEXT LINE 4

## 11. /WINSHTLQ/MATERIAL\_GET\_NEXT

- ✓ **PURPOSE:** To get next Material number available for the entered Material Type.
- ✓ **DESCRIPTION:** This function is used to get next Material number available for the entered Material Type.

### The function module interface

- ✓ **Import Parameters**
  - **MATERIALART:** Material Type
- ✓ **Export Parameters**
  1. **MATERIALNR:** Get Material Number
  2. **RETURNCODE:** Get Error messages
  3. **E\_RETURN:** Get Success message in this field if the FM has run successfully.
- ✓ **Exceptions:**
  1. **NO\_AUTHORIZED\_TO\_CREATE:** This exception occurs if the user is unauthorized to generate next Material Number.
  2. **NO\_RELEASE\_FOUND:** This exception occurs if no SAP BASIS release is found in the SAP system.
  3. **NO\_INTERNAL\_INTERVAL:** This exception occurs if no Internal interval is defined in the SAP for the entered Material Type.
  4. **TYPE\_NOT\_FOUND:** This exception occurs if the entered Material type is not found in the SAP.

### Example

Enter Input as below:

**Test Function Module: Initial Screen**

Debugging Test data directory

Test for function group /WINSHTLQ/FUGR\_MATNR\_NEXT  
Function module /WINSHTLQ/MATERIAL\_GET\_NEXT  
Uppercase/Lowercase

RFC target sys:

Import parameters	Value
MATERIALART	HAWA

You will get next material number available for entered Material Type.

**Test Function Module: Result Screen**

Test for function group /WINSHTLQ/FUGR\_MATNR\_NEXT  
 Function module /WINSHTLQ/MATERIAL\_GET\_NEXT  
 Uppercase/Lowercase

Runtime: 0 Microseconds

RFC target sys:

Import parameters	Value
MATERIALART	HAWA

Export parameters	Value
MATERIALNR	18103
RETURNCODE	
E_RETURN	Success

If the entered Material Type is invalid, it will display message as below:

**Test Function Module: Result Screen**

Test for function group /WINSHTLQ/FUGR\_MATNR\_NEXT  
 Function module /WINSHTLQ/MATERIAL\_GET\_NEXT  
 Uppercase/Lowercase

Runtime: 0 Microseconds

Exception TYPE\_NOT\_FOUND  
 Message ID: M3 Message number:  
 Message:  
 The material type HAW does not exist

RFC target sys:

Import parameters	Value
MATERIALART	HAW

Export parameters	Value
MATERIALNR	
RETURNCODE	
E_RETURN	

## 12. /WINSHTLQ/ACC\_DOC\_COPA\_POST

- ✓ **PURPOSE:** Accounting: Posting
- ✓ **DESCRIPTION:** This function is used for accounting posting.

### The function module interface

- ✓ **Import Parameters**
  - **DOCUMENTHEADER:** Enter Header details like company code, posting dates, Fiscal year etc.
  - **CUSTOMERCPD:** Enter One-time Customer details like Name, Address, Bank account number etc.
  - **CONTRACTHEADER:** Enter Additional Contract Accounts Receivable and Payable Header Line details like Doc no., Doc. Type, Reconciliation Key for General Ledger Accounting.
- ✓ **Export Parameters**
  1. **OBJ\_TYPE:** This parameter will return the Reference Transaction number
  2. **OBJ\_KEY:** This parameter will return the Reference Key
  3. **OBJ\_SYS:** This parameter will return the Logical system of source document
- ✓ **Tables**
  1. **ACCOUNTGL:** This Table is used for transferring G/L account line items

### The following fields must be filled out:

1. **GL\_ACCOUNT:** The currency fields are not contained in this parameter. They must be transferred separately in the parameter CurrencyAmount for the various currency types and currencies. The system creates a link to the relevant line item via the line-item number ITEMNO\_ACC, which for this reason must be unique for each line item.

**DEFAULT:** If this method can be used to post G/L account line items across several company codes, the following fields are also contained in the G/L account line items:

- **COMP\_CODE** Company code
- **PSTNG\_DATE** Posting date
- **DOC\_TYPE** Document type
- **AC\_DOC\_NO** Document number
- **FISC\_YEAR** Fiscal year
- **FIS\_PERIOD** Fiscal period

If the fields are blank, the information is copied from the document header (parameter **Document header**).

2. **ACCOUNTRECEIVABLE:** This table is used for transferring customer line items.

If the CUSTOMER number is not given, only the general ledger is updated. In this case, an entry must be made in the field GL\_ACCOUNT (G/L account for general ledger accounting).

The currency fields are not contained in this parameter. They must be transferred separately in the parameter CurrencyAmount for the various currency types and currencies. The system creates a link to the relevant line item via the line-item number ITEMNO\_ACC, which for this reason must be unique for each line item.

3. **ACCOUNTPAYABLE:** This table is used for transferring vendor line items.

If the vendor account number VENDOR\_NO is not given, only the general ledger is updated. In this case, an entry must be made in the field GL\_ACCOUNT (G/L account for general ledger accounting).

The currency fields are not contained in this parameter. They must be transferred separately in the parameter CurrencyAmount for the various currency types and currencies. The system creates a link to the relevant line item via the line-item number ITEMNO\_ACC, which for this reason must be unique for each line item.

4. **ACCOUNTTAX:** This table is used for transferring tax lines.

The following fields must be filled:

- **ITEMNO\_ACC** Item number
- **TAX\_CODE** Tax code
- **COND\_KEY** Condition type

If the Jurisdiction Code is active, these fields must also be filled.

The currency fields are not contained in this parameter. They must be transferred in the parameter Currency Amount for the different currency types or currencies. A connection is created to the corresponding document line item via the item number ITEMNO\_ACC which must be unique for all line items.

5. **CURRENCYAMOUNT:** This parameter describes the currency fields for a particular line item. The parameter is a required field for the method POST. If the parameter is not maintained for the method CHECK, only a general account assignment check is carried out. Individual line items or several documents can be checked at once in this way.

### The following fields must be filled out:

- **CURRENCY** Currency key
- **AMT\_DOCCUR** Currency amount

### Value range

The field CURR\_TYPE (currency type) describes the role played by a currency in the SAP System.

The currency type can take on the following values, for Example

- 00 Transaction currency
- 10 Company code currency
- 20 Controlling area currency
- 30 Group currency
- 40 Hard currency
- 50 Index-based currency
- 60 Global company currency

If an entry is not made in the field CURR\_TYPE, the default value "00" is retained.

If only one data record is transferred in transaction currency for each line item, the field CURR\_TYPE does not need to be filled out. If several data records are transferred for each line item, the currency type must be entered for each record.

### 6. **CRITERIA: CO-PA Account** Assignment Characteristics

You can use this parameter to transfer the Profitability Analysis (CO-PA) characteristics to an account assignment. The system creates a link to the relevant line item via the line-itemnumber ITEMNO\_ACC, which for this reason must be unique for each line item.

**Note:** The contents of the parameter must be sorted before being transferred. If not, errors may occur when assigning error messages to parameter lines.

### 7. **VALUEFIELD:** CO-PA Account Assignment Value Fields

This parameter can be used to transfer the value fields for the profitability analysis (CO-PA) for an accounting assignment.

The system creates a link to the relevant line item via the line item number ITEMNO\_ACC, which therefore must be unique for each line item.

**Note:** The parameter contents must be sorted when transferred, otherwise errors may occur when assigning error messages to the parameter lines.

### 8. **EXTENSION1:** Container for 'Customer Exit' Parameter

This exit can be used to edit the accounting document before it is transferred to the accounting components for the update.

9. **RETURN:** This table contains the return messages from the method call.

**Value range:** The first message contains the general status of processing:

Type	ID	Number	Message
S	RW	605	Document posted successfully
S	RW	614	Document check without errors
A	RW	609	Error in document

The fields MESSAGE\_V1, MESSAGE\_V2 and MESSAGE\_V3 contain the following information:

- Reference transaction ([OBJ\\_TYPE](#))
- Reference key ([OBJ\\_KEY](#))
- Logical system of source document ([OBJ\\_SYS](#))

Additional messages are transferred with the following information:

- Parameter (PARAMETER)
- Row (ROW)
- Field (FIELD) in which the error occurred.

10. **PAYMENTCARD:** In this parameter data for credit card payment can be transferred.

#### **Link to debtor line:**

The item is also transferred to the debtor line (ACCOUNTRECEIVABLE) and must therefore contain the same item number (ITEMNO\_ACC).

11. **CONTRACTITEM:** Additional Contract Accounts Receivable and Payable Document Line Item.

In this parameter fields are transferred that are concerned with an accounting document not including the contract accounts receivable and payable (FI-CA) at the item level.

#### **Link to debtor line:**

The item is also transferred to the debtor line (ACCOUNTRECEIVABLE) and must therefore contain the same item number (ITEMNO\_ACC).

**12. EXTENSION2:** Reference Structure for BAPI Parameters  
EXTENSIONIN/EXTENSIONOUT

This enhancement structure can be used to transfer additional fields.

There are four fields available for each data record (VALUEPART1 - 4), which each contain 240 characters. A defined ABAP dictionary structure can be transferred in the field STRUCTURE, based on which the field contents are read.

The accounting document can be supplemented and changed in an implemented BADI ( ACC\_DOCUMENT) before the active accounting component is called.

**Default:**

The class CL\_EXM\_IM\_ACC\_DOCUMENT is available as example implementation. Here, for each data record of the parameter that is transferred the ABAP dictionary structure is checked and the content of the fields VALUEPART1 to -4 is set in the structure. If the structure that is transferred contains a field POSNR, the associated lines of the accounting document are updated using the contents of the enhancement.

**13. REALESTATE:** Real Estate Account Assignment Data

This parameter can be used to transfer account assignments for Real Estate Management. The current line represents an addition to the accounting document line items (ACCOUNTGGL, ACCOUNTRECEIVABLE, ACCOUNTPAYABLE). The line in parameter REALESTATE must be connected to the document line item (ITEMNO\_ACC) using the line number.

**14. ACCOUNTWT:** This table is used for Withholding Tax Information like ITEMNO\_ACC, Withholding Tax Base Amount (Local Currency), Withholding TaxBase Amount (Document Currency), **Withholding tax** code etc.



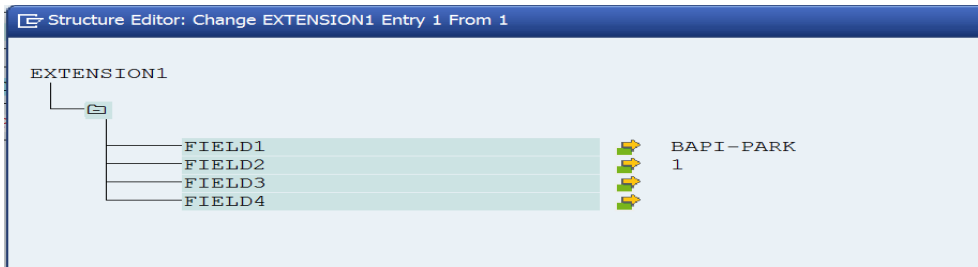
## Example

Enter Input as below:

DOCUMENTHEADER	
OBJ_TYPE	
OBJ_KEY	
OBJ_SYS	
BUS_ACT	
USERNAME	ECC6DEV
HEADER_TXT	HEADER
COMP_CODE	1000
DOC_DATE	03.03.2014
PSTNG_DATE	03.03.2014
TRANS_DATE	
FISC_YEAR	2014
FIS_PERIOD	00
DOC_TYPE	SA
REF_DOC_NO	
AC_DOC_NO	
OBJ KEY R	

- **AccountGl table:**  
ITEMNO\_ACC = '0000000001'.  
GL\_ACCOUNT = '473000'.  
ITEM\_TEXT = 'LINE 1'.  
COSTCENTER = '3010'.  
ITEMNO\_ACC = '0000000002'.  
GL\_ACCOUNT = '405200'.  
ITEM\_TEXT = 'LINE 2'.  
COSTCENTER = '3020'.
- **CURRENCYAMOUNT Table:**  
ITEMNO\_ACC = '0000000001'.  
CURR\_TYPE = '00'.  
CUURENCY = 'EUR'.  
AMT\_DOCCUR = '100'.  
  
ITEMNO\_ACC = '0000000002'.  
CURR\_TYPE = '00'.  
CUURENCY = 'EUR'.  
AMT\_DOCCUR = '-100'.

- **EXTENSION1 Structure:**



**Output:**

RFC target sys:

Import parameters	Value
DOCUMENTHEADER CUSTOMERCPD CONTRACTHEADER	
Export parameters	Value
OBJ_TYPE OBJ_KEY OBJ_SYS	BKPPF 010000005610002014 T90CLNT090

Document 0100000056 is parked and can be seen in t-code FBV0.

**Display Document: Data Entry View**

Display Currency    General Ledger View

Document Number	1000000056	Company Code	1000	Fiscal Year	2014
Document Date	03.03.2014	Posting Date	03.03.2014	Period	3
Reference		Cross-Comp.No.			
Currency	EUR	Texts exist	<input type="checkbox"/>	Ledger Group	

CoCd	Item	PK	S	Account	Description	Amount	Currency	Tx	Cost Center	Order	Profit Center	Segment	Billing Doc.	SLNo
1000	1	40		473000	Postage	10,00	EUR	V0	3010		1400	SERV		
	2	50		405200	Consumable Supplies	10,00	EUR	V0	3020		1400	SERV		

## 13. /WINSHTLQ/POST\_ACDOC\_MULT\_CURR

- ✓ **PURPOSE:** Accounting: Posting
- ✓ **DESCRIPTION:** This function is used for Posting Acc Doc in Multiple Currencies

### The function module interface

#### ✓ Import Parameters

1. **DOCUMENTHEADER:** Enter Header details like company code, posting dates, Fiscal year etc.
2. **CUSTOMERCPD:** Enter Partner billing doc (load receivable) details like Name, Address, Bank account number etc.
3. **CONTRACTHEADER:** Enter Additional Contract Accounts Receivable and Payable Header Line details like Doc no., Doc. Type, Reconciliation Key for General Ledger Accounting.
4. **CHECK\_POST:** For posting, this Flag should be set to 'P' and for checking data, it should be set to 'C'.

#### ✓ Export Parameters

1. **OBJ\_TYPE:** This parameter will return the Reference Transaction number
2. **OBJ\_KEY:** This parameter will return the Reference Key
3. **OBJ\_SYS:** This parameter will return the Logical system of source document

#### ✓ Tables

1. **ACCOUNTGL:** This Table is used for transferring G/L account line items

The following fields must be filled out:

- ✓ **GL\_ACCOUNT** G/L account: **Default:** If this method can be used to post G/L account line items across several company codes, the following fields are also contained in the G/L account line items.
- ✓ **COMP\_CODE** Company code
- ✓ **PSTNG\_DATE** Posting date
- ✓ **DOC\_TYPE** Document type
- ✓ **AC\_DOC\_NO** Document number
- ✓ **FISC\_YEAR** Fiscal year
- ✓ **FIS\_PERIOD** Fiscal period

If the fields are blank, the information is copied from the document header (parameter Document header).

2. **ACCOUNTRECEIVABLE:** This table is used for transferring customer line items.

If the CUSTOMER number is not given, only the general ledger is updated. In this case, an entry must be made in the field GL\_ACCOUNT (G/L account for general ledger accounting).

The currency fields are not contained in this parameter. They must be transferred separately in the parameter CurrencyAmount for the various currency types and currencies. The system creates a link to the relevant line item via the line item number ITEMNO\_ACC, which for this reason must be unique for each line item.

3. **ACCOUNTPAYABLE:** This table is used for transferring vendor line items.

If the vendor account number VENDOR\_NO is not given, only the general ledger is updated. In this case, an entry must be made in the field GL\_ACCOUNT (G/L account for general ledger accounting).

The currency fields are not contained in this parameter. They must be transferred separately in the parameter CurrencyAmount for the various currency types and currencies. The system creates a link to the relevant line item via the line item number ITEMNO\_ACC, which for this reason must be unique for each line item.

2. **ACCOUNTTAX:** This table is used for transferring tax lines.

The following fields must be filled:

- **ITEMNO\_ACC** Item number
- **TAX\_CODE** Tax code
- **COND\_KEY** Condition type

If the Jurisdiction Code is active, these fields must also be filled.

The currency fields are not contained in this parameter. They must be transferred in the parameter Currency Amount for the different currency types or currencies. A connection is created to the corresponding document line item via the item number ITEMNO\_ACC which must be unique for all line items.

3. **CURRENCYAMOUNT:** This parameter describes the currency fields for a line item. The parameter is a required field for the method POST. If the parameter is not maintained for the method CHECK, only a general account assignment check is carried out. Individual line items or several documents can be checked at once in this way.

The following fields must be filled out:

- ✓ **CURRENCY** Currency key
- ✓ **AMT\_DOCCUR** Currency amount

## Value range

The field CURR\_TYPE (currency type) describes the role played by a currency in the SAP System. The currency type can take on the following values, for

Example

- ✓ 00 Transaction currency
- ✓ 10 Company code currency
- ✓ 20 Controlling area currency
- ✓ 30 Group currency
- ✓ 40 Hard currency
- ✓ 50 Index-based currency
- ✓ 60 Global company currency

If an entry is not made in the field CURR\_TYPE, the default value "00" is retained.

If only one data record is transferred in transaction currency for each line item, the field CURR\_TYPE does not need to be filled out. If several data records are transferred for each line item, the currency type must be entered for each record.

#### 4. **CRITERIA:** CO-PA Account Assignment Characteristics

You can use this parameter to transfer the Profitability Analysis (CO-PA) characteristics to an account assignment.

The system creates a link to the relevant line item via the line item number ITEMNO\_ACC, which for this reason must be unique for each line item.

**Note: The contents of the parameter must be sorted before being transferred. If not, errors may occur when assigning error messages to parameter lines.**

#### 5. **VALUE:** CO-PA Account Assignment Value Fields

This parameter can be used to transfer the value fields for the profitability analysis (CO-PA) for an accounting assignment.

The system creates a link to the relevant line item via the line item number ITEMNO\_ACC, which therefore must be unique for each line item.

**Note: The parameter contents must be sorted when transferred, otherwise errors may occur when assigning error messages to the parameter lines.**

6. **EXTENSION1:** Container for 'Customer Exit' Parameter

This exit can be used to edit the accounting document before it is transferred to the accounting components for the update.

7. **RETURN:** This table contains the return messages from the method call.

**Value range**

The first message contains the general status of processing:

Type	ID	Number	Message
S	RW	605	Document posted successfully
S	RW	614	Document check without errors
A	RW	609	Error in document

The fields MESSAGE\_V1, MESSAGE\_V2 and MESSAGE\_V3 contain the following information:

- ✓ Reference transaction ( OBJ\_TYPE)
- ✓ Reference key ( OBJ\_KEY)
- ✓ Logical system of source document ( OBJ\_SYS)

Additional messages are transferred with the following information:

- ✓ Parameter ( PARAMETER)
- ✓ Row ( ROW)
- ✓ Field (FIELD) in which the error occurred.

8. **PAYMENTCARD:** In this parameter data for credit card payment can be transferred.

**Link to debtor line:**

The item is also transferred to the debtor line (ACCOUNTRECEIVABLE) and must therefore contain the same item number (ITEMNO\_ACC).

9. **CONTRACTITEM:** Additional Contract Accounts Receivable and Payable Document Line Item

In this parameter fields are transferred that are concerned with an accounting document not including the contract accounts receivable and payable (FI-CA) at the item level.

### **Link to debtor line:**

The item is also transferred to the debtor line (ACCOUNTRECEIVABLE) and must therefore contain the same item number (ITEMNO\_ACC).

### **10. EXTENSION2:** Reference Structure for BAPI Parameters EXTENSIONIN/EXTENSIONOUT

This enhancement structure can be used to transfer additional fields.

There are four fields available for each data record (VALUEPART1 - 4), which each contain 240 characters. A defined ABAP dictionary structure can be transferred in the field STRUCTURE, based on which the field contents are read.

The accounting document can be supplemented and changed in an implemented BAdI (ACC\_DOCUMENT) before the active accounting component is called.

**Default:** The class CL\_EXM\_IM\_ACC\_DOCUMENT is available as example implementation. Here, for each data record of the parameter that is transferred the ABAP dictionary structure is checked and the content of the fields VALUEPART1 to -4 is set in the structure. If the structure that is transferred contains a field POSNR, the associated lines of the accounting document are updated using the contents of the enhancement.

### **11. REALESTATE:** Real Estate Account Assignment Data

This parameter can be used to transfer account assignments for Real Estate Management.

The current line represents an addition to the accounting document line items (ACCOUNTGL, ACCOUNTRECEIVABLE, ACCOUNTPAYABLE). The line in parameter REALESTATE must be connected to the document line item (ITEMNO\_ACC) using the line number.

**12. ACCOUNTWT:** This table is used for Withholding Tax Information like ITEMNO\_ACC, Withholding Tax Base Amount (Local Currency), Withholding Tax Base Amount (Document Currency), Withholding tax code etc.

## Example

Enter Input as below:



The screenshot shows the SAP Structure Editor for the DOCUMENTHEADER table. The table fields and their values are as follows:

Field Name	Value
OBJ_TYPE	
OBJ_KEY	
OBJ_SYS	
BUS_ACT	
USERNAME	ECC6DEV
HEADER_TXT	HEADER
COMP_CODE	1000
DOC_DATE	03.03.2014
PSTNG_DATE	03.03.2014
TRANS_DATE	
FISC_YEAR	2014
FIS_PERIOD	00
DOC_TYPE	SA
REF_DOC_NO	
AC_DOC_NO	
OBJ_KEY_R	
REASON_REV	
COMPO_ACC	
REF_DOC_NO_LONG	
ACC_PRINCIPLE	
NEG_POSTNG	
OBJ_KEY_INV	
BILL_CATEGORY	

- **AccountGl table:**

ITEMNO\_ACC = '0000000001'.  
GL\_ACCOUNT = '473000'.  
ITEM\_TEXT = 'LINE 1'.  
COSTCENTER = '3010'.  
TRANS\_CURR\_TYPE = '00'.  
TRANS\_CURRENCY = 'EUR'.  
TRANS\_AMT\_DOCCUR = '100'.

ITEMNO\_ACC = '0000000002'.  
GL\_ACCOUNT = '405200'.  
ITEM\_TEXT = 'LINE 2'.  
COSTCENTER = '3020'.  
TRANS\_CURR\_TYPE = '00'.  
TRANS\_CURRENCY = 'EUR'.  
TRANS\_AMT\_DOCCUR = '-100'.

- CHECK\_POST = 'P'. (If document needs to be posted, else 'C' to check the data before posting document)



**Output:** Return Table shows Document Number (Concatenated with Company code and fiscal year) posted.

**Structure Editor: Display RETURN from Entry 1**

Column Entry Metadata

1 Entry

T	ID	NUM	MESSAGE
S	RW	605	Document posted successfully: BKPFF 010000005210002014 T90CLNT090

Posted Document can be seen in t-code FB03.

**Display Document: Data Entry View**

Display Currency General Ledger View

Data Entry View

Document Number	100000052	Company Code	1000	Fiscal Year	2014
Document Date	03.03.2014	Posting Date	03.03.2014	Period	3
Reference		Cross-Comp.No.			
Currency	EUR	Texts exist	<input type="checkbox"/>	Ledger Group	

CoCd	Item	PK	S	Account	Description	Amount	Currency	Tx	Cost Center	Order	Profit Center	Segment	Billing Doc.	SLNo
1000	1	40		473000	Postage	100,00	EUR	V0	3010		1400	SERV		
	2	50		405200	Consumable Supplies	100,00-	EUR	V0	3020		1400	SERV		

## 14. /WINSHTLQ/WFM\_MAT\_EXTEND

- ✓ **PURPOSE:** To extend Material Number
- ✓ **DESCRIPTION:** This function module is used for extending Material Numbers. It is a wrapper of 'BAPI\_MATERIAL\_SAVEREPLICA' FM.

### The Function Module Interface

#### ✓ Import Parameters

**1. NOAPPLLOG:** This checkbox structure can be used to specify whether an application log is generated or not.

If the parameter is initial, the system generates an application log.

**2. NOCHANGEDOC:** This checkbox structure can be used to specify whether change documents should be created during processing of material data.

If the parameter is initial, the system generates change documents.

**3. TESTRUN:** This checkbox structure can be used to specify whether data is processed with or without a subsequent update.

If the parameter is initial, the system updates the data after processing.

**4. INPFLDCHECK:** You must set this flag if you want to receive a message if there are fields that are not ready for input.

**5. FLAG\_CAD\_CALL:** This flag must be set if the BAPI is called from a CAD system.

**6. NO\_ROLLBACK\_WORK:** This is an optional parameter and used to override Rollback Work.

**7. FLAG\_ONLINE:** This is an optional parameter and used for specifying No ALE Field Selection

#### ✓ Export Parameters

**1. RETURN:** This structure is used to return the return code to the calling program. For information on the meaning of the individual fields in the structure, see the corresponding data element documentation on the individual fields.

Since numerous information and error messages can occur during processing of the material, the messages are collected and stored in table RETURNMESSAGES.

The return parameter only contains the information as to whether the material could be successfully created/changed or not.

If the material could be created/changed without any major errors, the return parameter receives a message of type S (success message). Otherwise, it receives a message of type E (error message).

## ✓ Tables

1. **HEADDATA:** This structure is used for storing data required for maintaining a material. The fields match the fields displayed in the initial screen for online maintenance when a material is created for the first time.

When creating new material master data, only a single material can be created or changed in this structure.

For information on the meaning of the individual fields in the structure, see the corresponding data element documentation for the individual fields.

**Notes:** The following fields must contain data in the structure when a material is created for the first time:

### •MATERIAL

Only external number assignments are used for material number (MATERIAL). If an internal number is to be assigned, method BAPI\_MATERIAL\_GETINTNUMBER must be called. This method defines an internal number for the material that is to be created. The new internal number should be entered in field HEADDATA-MATERIAL.

### •IND\_SECTOR

### •MATL\_TYPE

When data is changed, the field MATERIAL in the structure must contain data.

At least one of the following views must be selected:

- BASIC\_VIEW (required when a new material is created)
- SALES\_VIEW
- PURCHASE\_VIEW
- MRP\_VIEW
- FORECAST\_VIEW
- WORK\_SCHED\_VIEW
- PRT\_VIEW
- STORAGE\_VIEW
- WAREHOUSE\_VIEW
- QUALITY\_VIEW
- ACCOUNT\_VIEW
- COST\_VIEW

Update structures (for example, CLIENTDATAX, PLANTDATAX) exist for individual data structures (for example, CLIENTDATA, PLANTDATA). The update structures contain information about which fields in the assigned data structure should be changed.

2. **CLIENTDATA:** Use this structure to display the general basic data for the materials.

For information on the meaning of the individual fields in the structure, see the corresponding data element documentation for the individual fields.

#### **Notes**

If you have defined other fields as required entries in Customizing for the Material Master in Maintain Field Selection for Data Screens, you must also specify valid values for these fields.

The definition of further required fields can also affect the other interface parameters because certain fields there then also require entries.

Field PO\_UNIT (or PO\_UNIT\_ISO) should only contain a unit of measure if the order unit of measure does not match the base unit of measure (Feld BASE\_UOM or BASE\_UOM\_ISO).

If units of measure that do not yet exist in the system are transferred in the following fields, an entry in the structure UNITSOFMEASURE must also be transferred for these new units of measure:

- BASE\_UOM (or BASE\_UOM\_ISO)
- PO\_UNIT (or PO\_UNIT\_ISO)

3. **CLIENTDATA:**

Use this structure (checkbox) to control which fields in the assigned data structure (CLIENTDATA) contain change relevant data.

This makes it possible to supply valid values only to the relevant fields in the assigned user data structure. These fields are then selected in the checkbox structure, ensuring that all non-selected fields retain their current value in the database.

All fields in the structure use the data element BAPIUPDATE.

#### **Example**

Only the old material number is to be changed for the material with MAT01:

Structure CLIENTDATA contains the following data:

- OLD\_MAT\_NO MAT01\_OLD

The remaining fields in structure CLIENTDATA can contain any data (or can simply be left blank).

Structure CLIENTDATA must be configured as follows:

- OLD\_MAT\_NO X

All other fields in structure CLIENTDATA must be blank. The material number is defined in field HEADDATA-MATERIAL.

4. **PLANTDATA:**

This structure can be used to transfer plant-specific data for the material.

For information on the meaning of the individual fields in the structure, see the corresponding data element documentation for the individual fields.

5. **PLANTDATA\_X:**

This structure (checkbox structure) is used to control which fields in the assigned user data structure (PLANTDATA) contain change-relevant information.

This makes it possible to supply valid values only to the relevant fields in the assigned user data structure. These fields are then selected in the checkbox structure, ensuring that all non-selected fields retain their current value in the database.

All fields in the structure use the data element BAPIUPDATE.

6. **FORECASTPARAMETERS:**

This structure can be used to transfer the forecast parameters for the material.

For information on the meaning of the individual fields in the structure, see the corresponding data element documentation for the individual fields.

7. **FORECASTPARAMETERS\_X:**

This structure (checkbox structure) is used to control which fields in the assigned user data structure (FORECASTPARAMETERS) contain change-relevant information.

This makes it possible to supply valid values only to the relevant fields in the assigned user data structure. These fields are then selected in the checkbox structure, ensuring that all non-selected fields retain their current value in the database.

All fields in the structure use the data element BAPIUPDATE.

8. **PLANNINGDATA:**

This structure can be used to transfer the preplanning data for the material.

9. **PLANNINGDATA\_X:**

This structure (checkbox structure) is used to control which fields in the assigned user data structure (PLANNINGDATA) contain change-relevant information.

This makes it possible to supply valid values only to the relevant fields in the assigned user data structure. These fields are then selected in the checkbox structure, ensuring that all non-selected fields retain their current value in the database.

All fields in the structure use the data element BAPIUPDATE.

**10. STORAGELOCATIONDATA:**

This structure can be used to transfer the storage location-specific data for the material.

**11. STORAGELOCATIONDATA:**

This structure (checkbox structure) is used to control which fields in the assigned user data structure (STORAGELOCATIONDATA) contain change-relevant information.

This makes it possible to supply valid values only to the relevant fields in the assigned user data structure. These fields are then selected in the checkbox structure, ensuring that all non-selected fields retain their current value in the database.

All fields in the structure use the data element BAPIUPDATE.

**12. VALUATIONDATA:**

This structure can be used to transfer the valuation data for the material.

**13. VALUATIONDATA:**

This structure (checkbox) controls which fields contain change relevant information for the assigned data structure (VALUATIONDATA).

This makes it possible to supply valid values only to the relevant fields in the assigned user data structure. These fields are then selected in the checkbox structure, ensuring that all non-selected fields retain their current value in the database.

All fields in the structure use the data element BAPIUPDATE.

**14. WAREHOUSENUMBERDATA:**

This structure can be used to transfer the warehouse number-specific data for the material.

**15. WAREHOUSENUMBERDATA:**

Use this structure (checkbox list) to control which fields in the assigned data structure (WAREHOUSENUMBERDATA) contain change-relevant data.

This makes it possible to supply valid values only to the relevant fields in the assigned user data structure. These fields are then selected in the checkbox structure, ensuring that all non-selected fields retain their current value in the database.

All fields in the structure use the data element BAPIUPDATE.

**16. SALESDATA:**

This structure can be used to transfer the material's sales data.

#### 17. SALESDATAX:

This structure (checkbox structure) is used to control which fields in the assigned user data structure (SALESDATA) contain change-relevant information.

This makes it possible to supply valid values only to the relevant fields in the assigned user data structure. These fields are then selected in the checkbox structure, ensuring that all non-selected fields retain their current value in the database.

All fields in the structure use the data element BAPIUPDATE.

#### 18. STORAGETYPEDATA:

This structure can be used to transfer the storage type-specific data for the material.

#### 19. STORAGETYPEDATAX:

This structure (checkbox structure) is used to control which fields in the assigned user data structure (STORAGELOCATIONDATA) contain change-relevant information.

This makes it possible to supply valid values only to the relevant fields in the assigned user data structure. These fields are then selected in the checkbox structure, ensuring that all non-selected fields retain their current value in the database.

All fields in the structure use the data element BAPIUPDATE.

#### 20. MATERIALDESCRIPTION:

This structure can be used to transfer the material's short texts.

#### 21. UNITSOFTMEASURE:

Use this structure to adopt the units of measure for a material.

For information on the meaning of the individual fields in the structure, see the corresponding data element documentation for the individual fields.

This structure is used for adopting data for the base unit of measure for an article and for alternative units of measure. The base unit of measure is the unit of measure defined in structure CLIENTDATA in field BASE\_UOM (or BASE\_UOM\_ISO).

#### **Note:**

If field EAN\_UPC contains an EAN that does not exist in your system, an appropriate entry must be made for the new EAN in structure INTERNATIONALARTNOS.

## 22. **UNITSOFMASUREX:**

This structure (checkbox structure) is used to control which fields in the assigned user data structure (UNITSOFMASURE) contain change-relevant information.

This makes it possible to supply valid values only to the relevant fields in the assigned user data structure. These fields are then selected in the checkbox structure, ensuring that all non-selected fields retain their current value in the database.

All fields in the structure use the data element BAPIUPDATE.

## 23. **INTERNATIONALARTNOS:**

Use this structure to transfer the European Article Numbers (EAN) for a material.

For information on the meaning of the individual fields in the structure, see the corresponding data element documentation for the individual fields.

In this structure, data for the main EAN and alternative EANs is entered for each unit of measure. The main EAN is the EAN that has been entered in structure UNITSOFMASURE in field EAN\_UPC for the relevant unit of measure.

### **Notes:**

If EANs are also used for specific vendors (see structure VENDOREAN), they must also be transferred in structure INTERNATIONALARTNOS.

If data is transferred for a unit of measure (field UNIT or UNIT\_ISO) that does not yet exist in the system, an entry in the structure UNITSOFMASURE must also be transferred for this new unit of measure.

If a material EAN is to be changed (for example, the assignment to the main EAN is to be changed), all units of measure assigned in structure UNITSOFMASURE have to be transferred because this structure contains information identifying the main EAN for the appropriate unit of measure.



#### 24. MATERIALLONGTEXT:

Use this structure to display material long texts.

**Note:** Different long texts (for example, basic data texts, purchasing texts, internal notes, sales and distribution texts, material notes) for be created for a material. To do this, fields APPBJECT, TEXT\_NAME, TEXT\_ID and LANGU or LANGU\_ISO have to be filled in table parameter MATERIALLONGTEXT. Depending on the contents of field TEXT\_ID, field TEXT\_NAME must be structured as follows: the material number is displayed first. If the material number contains few than 18 characters, the remaining character spaces must be left blank. If the long text concerned is a material note (LTXT) the site key must also be included as of character 19. For a sales text (0001), the sales organization and the distribution channel must be included. The following table defines which combinations are permitted for the four different fields.

APPBJECT	TEXT_NAME	TEXT_ID	LANGU
MATERIAL	Number (18 characters) PO, green		Language
MDTXT	Number (18 characters) LTXT Site		Language
MVKE	Number (18 characters) 0001 Sales organization and distribution channel		Language

#### 25. TAXCLASSIFICATIONS:

This structure can be used to transfer the tax classification of the material.

#### 26. PRTDATA:

This structure can be used to transfer the material's production resources/tools data.

#### 27. PRTDATA:

This structure (checkbox structure) is used to control which fields in the assigned user data structure (PRTDATA) contain change-relevant information.

This makes it possible to supply valid values only to the relevant fields in the assigned user data structure. These fields are then selected in the checkbox structure, ensuring that all non-selected fields retain their current value in the database.

All fields in the structure use the data element BAPIUPDATE.

## 28. EXTENSION IN:

Use this structure to transfer customer-specific fields for materials.

Information for transferring these fields is detailed in the function module documentation.  
Value range

In addition to previously defined table fields, you can also provide customer-specific table fields with data. As these fields are created by the customer, they can only be recognized during runtime and must, therefore, be determined dynamically.

First, structures BAPI\_TE\_<NAME> (<NAME> = MARA, MARC, MARD, MBEW, MLGN, MLGT, MVKE, MARM, MAKT) and the appropriate check structures BAPI\_TE\_<NAME> (<NAME> = MARAX, for example) must be extended by the customer to include the required fields. The structures that are delivered as standard only contain the relevant key fields. When new fields are added to this structure, you should ensure that a field has the same name as the field in the database table. Furthermore, the fields in the structures BAPI\_TE\_<NAME> can only be CHARACTER fields. Data element BAPIUPDATE is to be used for the fields in the check structure (exception to this: key fields).

Parameters EXTENSIONIN and EXTENSIONINX are used for transferring data to the method. Field STRUCTURE contains the name of the structure (for example, BAPI\_TE\_MARA or BAPI\_TE\_MARAX). This name is used to identify the work area (for example, WA\_BAPI\_TE\_MARA or WA\_BAPI\_TE\_MARAX) to which the data is transferred.

The remaining fields in parameters EXTENSIONIN or EXTENSIONINX contain data for the key fields (for example, the material number) and the data for customer-specific fields. The number of characters reserved in both parameters for the contents of the customer-specific fields must match the number of characters for the relevant field in the work area. If the number of required characters is less, the remaining characters are left blank. Only when all characters have been filled can the contact for an additional field be transferred. Bear in mind that only data for which an appropriate indicator has been set in the work area can be updated to the database.

## 29. EXTENSIONINX:

This structure (checkbox structure) is used to control which fields in the assigned user data structure (EXTENSIONIN) contain change-relevant information.

This makes it possible to supply valid values only to the relevant fields in the assigned user data structure. These fields are then selected in the checkbox structure, ensuring that all non-selected fields retain their current value in the database.

All fields in the structure use the data element BAPIUPDATE.

## 30. FORECASTVALUES:

This structure can be used to transfer the forecast values for the material.

**31. UNPLNDCONSUMPTION:**

Use this structure to display uncalculated consumption for a specific material.

**32. TOTALCONSUMPTION:**

This structure can be used to transfer the total consumption of the material.

**33. RETURNMESSAGES:**

This table is used for storing messages processed by this method.

Information about the importance of individual fields in the structure can be found in the appropriate data element documentation for the individual fields.

As numerous warning and error messages may occur when the material is being processed, any messages that are generated are stored in this table. The last message to be processed is sent to the return parameter (RETURN).

**34. WFM\_RET\_MESSAGES:**

Return Message structure for WFM MAT Extend.

**Examples:**

**Enter Input as below:**

Import parameters	Value
NOAPPLLOG	<input type="checkbox"/>
NOCHANGEDOC	<input type="checkbox"/>
TESTRUN	<input type="checkbox"/>
INPFLDCHECK	<input checked="" type="checkbox"/>
FLAG_CAD_CALL	<input type="checkbox"/>
NO_ROLLBACK_WORK	<input type="checkbox"/>
FLAG_ONLINE	<input type="checkbox"/>

Tables	Value
HEADDATA	<input checked="" type="checkbox"/> 1 Entry
CLIENTDATA	<input checked="" type="checkbox"/> 1 Entry
CLIENTDATA_X	<input checked="" type="checkbox"/> 1 Entry
PLANTDATA	<input checked="" type="checkbox"/> 1 Entry
PLANTDATA_X	<input checked="" type="checkbox"/> 1 Entry
FORECASTPARAMETERS	<input checked="" type="checkbox"/> 0 Entries
FORECASTPARAMETERS_X	<input checked="" type="checkbox"/> 0 Entries
PLANNINGDATA	<input checked="" type="checkbox"/> 0 Entries
PLANNINGDATA_X	<input checked="" type="checkbox"/> 0 Entries
STORAGELOCATIONDATA	<input checked="" type="checkbox"/> 0 Entries
STORAGELOCATIONDATA_X	<input checked="" type="checkbox"/> 0 Entries
VALUATIONDATA	<input checked="" type="checkbox"/> 1 Entry
VALUATIONDATA_X	<input checked="" type="checkbox"/> 1 Entry
WAREHOUSENUMBERDATA	<input checked="" type="checkbox"/> 0 Entries
WAREHOUSENUMBERDATA_X	<input checked="" type="checkbox"/> 0 Entries
SALESDATA	<input checked="" type="checkbox"/> 0 Entries
SALESDATA_X	<input checked="" type="checkbox"/> 0 Entries

STORAGETYPEDATA	0 Entries
STORAGETYPEDATA	0 Entries
MATERIALDESCRIPTION	1 Entry
UNITSOFMEASURE	0 Entries
UNITSOFMEASUREX	0 Entries
INTERNATIONALARTNOS	0 Entries
MATERIALLONGTEXT	0 Entries
TAXCLASSIFICATIONS	0 Entries
PRTDATA	0 Entries
PRTDATA	0 Entries
EXTENSIONIN	0 Entries
EXTENSIONINX	0 Entries
FORECASTVALUES	0 Entries
UNPLNDCONSUMPTION	0 Entries
TOTALCONSUMPTION	0 Entries
RETURNMESSAGES	0 Entries

HEADDATA:

HEADDATA	
FUNCTION	[INS]
MATERIAL	ADSNEW-010
IND_SECTOR	M
MATL_TYPE	FERT
BASIC_VIEW	X
SALES_VIEW	
PURCHASE_VIEW	
MRP_VIEW	
FORECAST_VIEW	
WORK_SCHED_VIEW	
PRT_VIEW	
STORAGE_VIEW	X
WAREHOUSE_VIEW	
QUALITY_VIEW	
ACCOUNT_VIEW	X
COST_VIEW	
MATERIAL_EXTERNAL	
MATERIAL_GUID	
MATERIAL_VERSION	

CLIENTDATA:

CLIENTDATA	
FUNCTION	[INS]
MATERIAL	ADSNEW-010
DEL_FLAG	
MATL_GROUP	001
OLD_MAT_NO	
BASE_UOM	
BASE_UOM_ISO	ST
PO_UNIT	
PO_UNIT_ISO	
DOCUMENT	
DOC_TYPE	
DOC_VERS	
DOC_FORMAT	
DOC_CHG_NO	
PAGE_NO	
NO_SHEETS	000
PROD_MEMO	
PAGEFORMAT	
SIZE_DIM	
BASIC_MATL	
STD_DESCR	
DSN_OFFICE	
PUR_VALKEY	
NET_WEIGHT	0.000

CLIENTDATA:

CLIENTDATA	
FUNCTION	[INS]
MATERIAL	ADSNEW-010
DEL_FLAG	
MATL_GROUP	X
OLD_MAT_NO	
BASE_UOM	
BASE_UOM_ISO	X
PO_UNIT	
PO_UNIT_ISO	
DOCUMENT	
DOC_TYPE	
DOC_VERS	
DOC_FORMAT	
DOC_CHG_NO	
PAGE_NO	
NO_SHEETS	
PROD_MEMO	
PAGEFORMAT	
SIZE_DIM	
BASIC_MATL	
STD_DESCR	
DSN_OFFICE	
PUR_VALKEY	
NET_WEIGHT	

PLANTDATA:

PLANTDATA	
FUNCTION	[INS]
MATERIAL	ADSNEW-010
PLANT	1000
DEL_FLAG	
ABC_ID	
CRIT_PART	
PUR_GROUP	
ISSUE_UNIT	
ISSUE_UNIT_ISO	
MRPPROFILE	
MRP_TYPE	
MRP_CTRLER	
PLND_DELR	0
GR_PR TIME	0
PERIOD_IND	
ASSY_SCRAP	0.00

PLANTDATA X:

PLANTDATA X	
FUNCTION	[INS]
MATERIAL	ADSNEW-010
PLANT	1000
DEL_FLAG	
ABC_ID	
CRIT_PART	
PUR_GROUP	
ISSUE_UNIT	
ISSUE_UNIT_ISO	
MRPPROFILE	
MRP_TYPE	

VALUATIONDATA:

VALUATIONDATA	
FUNCTION	[INS]
MATERIAL	ADSNEW-010
VAL_AREA	1000
VAL_TYPE	
DEL_FLAG	
PRICE_CTRL	S
MOVING_PR	0.0000
STD_PRICE	123.4500
PRICE_UNIT	1
VAL_CLASS	7920
PR_CTRL_PP	
MOV_PR_PP	0.0000
STD_PR_PP	0.0000
PR_UNIT_PP	0
VCLASS_PP	
PR_CTRL_PY	
MOV_PR_PY	0.0000
STD_PR_PY	0.0000
VCLASS_PY	
PR_UNIT_PY	0
VAL_CAT	
FUTURE_PR	0.0000
VALID_FROM	
TAXPRICE_1	0.0000

VALUATIONDATA X:

VALUATIONDATA X	
FUNCTION	[INS]
MATERIAL	ADSNEW-010
VAL_AREA	1000
VAL_TYPE	
DEL_FLAG	
PRICE_CTRL	X
MOVING_PR	
STD_PRICE	X
PRICE_UNIT	X
VAL_CLASS	X
PR_CTRL_PP	
MOV_PR_PP	
STD_PR_PP	
PR_UNIT_PP	
VCLASS_PP	
PR_CTRL_PY	
MOV_PR_PY	
STD_PR_PY	
VCLASS_PY	

**MATERIALDESCRIPTION:**

MATERIALDESCRIPTION

FUNCTION	INS
MATERIAL	ADSNEW-010
LANGU	
LANGU_ISO	EN
MATL_DESC	ADS NEW 010 BAPI SAVEREPLICA
DEL_FLAG	
MATERIAL_EXTERNAL	
MATERIAL_GUID	
MATERIAL_VERSION	

**OUTPUT**

Structure Editor: Display RETURNMESSAGES from Entry 1

5 Entries

T	ID	NUM	MESSAGE
S	MK	102	Trying to create: ADSNEW-010 1000 1000
S	MG	160	The material cannot be maintained since no maintainable data transferred
S	MK	103	Trying to change: ADSNEW-010 1000 1000
A	MG	140	The field MBEW-STPRS/BAPI_MBEW-STD_PRICE is not ready for input and was therefore not transferred
S	M3	801	Material ADSNEW-010 changed

**MM03:**

**Display Material ADSNEW-010 (Finished product)**

Additional data Organizational levels

Basic data 1 Basic data 2 Plant data / stor. 1 Plant data / stor. 2 Accounting 1 Accounting 2

Material ADSNEW-010 ADS NEW 010 BAPI SAVEREPLICA TEST

General data

Base Unit of Measure	SHT	Sheet	Material Group	001
Old material number			Ext. Matl Group	
Division			Lab/Office	
Product allocation			Prod.hierarchy	
X-plant matl status			Valid from	
<input type="checkbox"/> Assign effect. vals			GenItemCatGroup	