Technical Documentation - Stock Margin elimination (Model S)

- 1 Access Management
- 2 DataFlow
  - 2.1 Overview
  - 2.2 Technical Rules on Workbench
    - 2.2.1 Main steps:
    - 2.2.2
    - 2.2.3
    - 2.2.4 Particular shift of month definition:
  - 2.3 Intermediate Flag
  - 2.4 Reporting
- 3 Data Loading
  - 3.1 Info Providers and objects loaded
  - 3.2 Loading frequency
  - 3.3 Points to be careful during the loading:
  - 3.4 Average performance
- 4 Data Quality Control
- 5 Operational Documentation
  - 5.1 Procedures
  - 5.2 Scheduling
  - 5.3 Monitoring
  - 5.4 Error Handling
  - 5.5 Known Bugs
  - 5.6 Roadmap

Access Management

Roles & Access

List of application role + menu role and explanation if we have several applications role with specials rules.

Role Code	Role Description	Explanation
ZR_RCS_CA_M69	MS - Model S	Role for queries and workbook
ZBI_RCS_MODS_A01	MS - Model S - End User role	Role application
ZBI_MODS	MS - Model S	Authorization object

Authorization Objects

List of authorization objects mandatory for the application.

Authorization object	Explanation
C_COMPCDE__C_AUTHMA	ZR_*_CA_P00
CPFCTR1_2	ZR_*_CA_P05
C_COMPCDE__C_MNGAREA	ZR_*_CA_P08
C_COMPPRS	ZR_*_CA_P07

DataFlow

Overview

Material stock - ABMMIC01 (WP1_400) data flow - **Be carrefull ABCOMMIC01 has key figure in summation mode.**

Model S data flow

Technical Rules on Workbench

Main steps:

1. **Extract quantity total stock from ADSO ABMMIC01 in ADSO APCOMS01 with help of query BW_QRY_ABMMIC01_0001.**

   The principal thing to know here it's about the way to extract the data.

   The goal is to extract and stock in APCOMS01 "Quantity Total Stock" 0TOTALSTCK from ABMMIC01. Info object 0TOTALSTCK is not physically stored in invenotry DSO. The only way to extract the key figure is to use a query as datasource (BW_QRY_ABMMIC01_0001).

   In query BW_QRY_ABMMIC01_0001 we used a variable "V_CALMONTH11" based on class ZCL_BIU001_V_0CALMONTH_0031.

   The class determine the month to load from current date and more or less x month(s). To know the number of months to add or removre from current date, the class read master data global filter (C_GLBFILT) on stream "STOCK_LOG":

Stream	Rule	Counter	V	Change flag (I inserted / D deleted)	Global Filter Descri	Sign	Option	Low	Hight	Active
STOCK_LOG	MONTH_VAR	001	A		Month operator to have filter on BW_QRY_ABMMIC01_0001	I	EQ	-	1	Y

   Normally, by default the month to load in ADSO APCOMS01 is current month - 1. To load other month it's necessary to change fields "low" and / or "hight" in master data.
   It's in ADSO APCOMS01 where we define by default C_SOURCE = LOGISTIC. We will use this field in filters to extract data from caclulation views to separate LOGISTIC data and WORKING CAPITAL (equal TRANSIT) data.
2. **Use calculation views to consolidate the different data sources.**
   a. See details on calculation views here.
      To use calculation views to load ADSO APCOMS04 we have eleven infopackages:

MS: Model S from Calc View	APCOMS04
TRSF DTS_CV_FMCO_CO_MS_IM_FINAL_01 (WBD_HANAIV) -> APCOMS04	0QQE5G02GEN8F3KA5UAYJKHRXMW537AQ
DTS_CV_FMCO_CO_MS_IM_FINAL_01	DTS_CV_FMCO_CO_MS_IM_FINAL_01
IP: DTS_CV_FMCO_CO_MS_IM_FINAL - Full - Logistic - Manual	ZPAK_5K2KJZYT3Z2I1IKS0PF7BNOKQ
IP: DTS_CV_FMCO_CO_MS_IM_FINAL - Full - WC - Manual	ZPAK_5K2JNCQBZFG24G14XFUF51I5M
IP: DTS_CV_FMCO_CO_MS_IM_FINAL - Full -Logistic-Prev Month	ZPAK_5K0FTPY40VE83RCUVPCY1XRMY
IP: DTS_CV_FMCO_CO_MS_IM_FINAL - Full -WC-LastDayPrevMonth	ZPAK_5K2J77IDFNWTLI19ZWVF2VT4A
IP: DTS_CV_FMCO_CO_MS_IM_FINAL -Full-Log-Plant 0k-1k - PrevM	ZPAK_5KW20S9Y32Q36CKIDOHFD525M
IP: DTS_CV_FMCO_CO_MS_IM_FINAL -Full-Log-Plant >9k - PrevM	ZPAK_5KT6AN6M4JD52Y4W7LOZ6OL62
IP: DTS_CV_FMCO_CO_MS_IM_FINAL -Full-Log-Plant repai - PrevM	ZPAK_5L2XG8IPUGTLDLP5BQIEM2TZ3
IP: DTS_CV_FMCO_CO_MS_IM_FINAL -Full-Log-Plant1k->2k - PrevM	ZPAK_5KSZU83EW5XZQ2R5ABTJSS8Q2
IP: DTS_CV_FMCO_CO_MS_IM_FINAL -Full-Log-Plant2k->3k - PrevM	ZPAK_5KSF211OXSL2DU49BKMEJBGH6
IP: DTS_CV_FMCO_CO_MS_IM_FINAL -Full-Log-Plant3k->4k - PrevM	ZPAK_5KSP01LY8H6X4ZLQVSID7YJOQ
IP: DTS_CV_FMCO_CO_MS_IM_FINAL -Full-Log-Plant4k->5k - PrevM	ZPAK_5KSP03QRI380B908I55S0I6UI
IP: DTS_CV_FMCO_CO_MS_IM_FINAL -Full-Log-Plant5k->6k - PrevM	ZPAK_5KSZURAQAO7SAFDJVHO8XP162
IP: DTS_CV_FMCO_CO_MS_IM_FINAL -Full-Log-Plant6k->7k - PrevM	ZPAK_5KVR9BW6NTXC7RFYEXBLE4KKQ
IP: DTS_CV_FMCO_CO_MS_IM_FINAL -Full-Log-Plant7k->8k - PrevM	ZPAK_5KVHDTBCNNI1VGXUB7IRKUHBE
IP: DTS_CV_FMCO_CO_MS_IM_FINAL -Full-Log-Plant8k->9k - PrevM	ZPAK_5KSF29KY08PF2VQ7SZ81PI14A
Data Transfer Processes	APCOMS04

      - Ten infopackages to extract "logistic" data (filter on field C_SOURCE = LOGISTIC), each IP are in full mode for previous current month and splitted by plant range ([0-1000] - [1001-2000] - [2001-3000] etc...).
      - IP: DTS_CV_FMCO_CO_MS_IM_FINAL - Full -WC-LastDayPrevMonth filter on posting date to have by default the last day of previous month and filter on C_SOURCE to have only "WORKING CAPITAL" data (equal TRANSIT data, from DBFIGL13).

3. **Cross System Material definition**
   We can have some cases called "cross system" when for a batch his source system is different of the source system from his manufacturing plant.
   The goal is to determine the batch concerned and then for each material associated, find the associated material from the other system.
   a. **C_BATCH2 cross system**
      We determine if a batch is cross system or not.
      We compare manufacturing plant from c_batch2 with plant "of c_plant and if we found a different source system, the batch is considered like a source system and we fill object **C_FLGCROS** "Flag Cross System" with X.
      The definition of batch cross system is done during the daily delta loading for PF1 and WP1 system.
      There is also a transformation from c_batch2 to c_batch2 to reload all batch2 and define the cross system (using only for historical batch).

Batch number	C_BATCH2
Batch number (Attributes)	ATTRIBUTES C_BATCH2
IOBJ C_BATCH -> IOBJ C_BATCH2	075DR3ORBYZI0JJWNO0PDZ0MZRBJHFDM
ODSO DBMMIC07 -> IOBJ C_BATCH2	0RFYIPO2NQ5YR8NJZOIP7OE4MRYAOJYA
TRCS IFS_2LIS_03_BF -> IOBJ C_BATCH2	0RJLJ9GAAPIUB03I8BTLA1PBMF8G5YN2
TRSF: 2LIS_03_BF -> C_BATCH2 (Solvay)	08E8U6KSOJOCJY8NRPAD4FPSDXG18JIH
TRSF: C_BATCH2 -> C_BATCH2	0IR58D9AEQM63TRYEYRKXFCAAMFUVJ5B
TRSF: DPMMIC03 -> C_BATCH2 (Solvay)	0QWFFVNV07DWKQERGR03TEUNT8WYXT0J
TRSF: DTS_C_BATCH2 -> C_BATCH2 (Solvay)	0ED6F0S23YWBO38SLKMT2ZUIT6YZM5WT
Transformation: DTS_C_BATCH -> C_BATCH2	0DUTQWB6ROPERUF1ETS2A7P8SMCERZ3L

Manufacturing plant were we force the flag cross system for the associated batch:
In the transformations to fill the cross system in the master data C_BATCH2, there is a rule to force for specifcs manufacturing plant the cross system associated (defined in global filter master data):

Data Browser: Table /BIC/PC_GLBFILT Select Entries 6

/BIC/C_STRE	/BIC/C_RULE	/BIC/C_GLBFILT	OBJVE	CHANGED	/BIC/C_DESC	/BIC/C_SIGN	/BIC/C_OPTIO	/BIC/C_LOW	/BIC/C_HIG
MODEL S	CROSSSYS_P	001	A		Exception PLANT cross system FOR C_BATCH2	I	EQ	7897	WP1_400
	CROSSSYS_P	002	A		Exception PLANT cross system FOR C_BATCH2	I	EQ	8575	WP1_400
	CROSSSYS_P	003	A		Exception PLANT cross system FOR C_BATCH2	I	EQ	8585	WP1_400

When we have defined which batch are cross system, we can use the following rules to find the material cross system associated.

b. aDSO cross system

After to have identified the C_BATCH2 with a cross system (C_FLGCROS = X), we can load the material associated in a aDSO ABMD001 to have the correspondence / conversion of materials between the systems.
In aDSO - "Material Cross System" there is the list of all materials with a cross system.
To find the conversion for a material we have two solutions, using a table from MPG system and use the old material attribute from master data c_matnr2.
The result found with the MPG table has priority over the result founded with old material number. It's why during the loadings **it's important to load (and activate) before the data from master data c_matnr2 and then the data from MPG system**, like that the MPG cross system will be the last "vision" of cross system.

i. *Rule with old material (from master data c_matnr2).*
The idea here is to find all material with an old material number who starts with "PF1" or "WP1".

Data Browser: Table /BIC/PC_MATNR2

LOGSYS	/BIC/C_MATN...	OBJVE...	/BIC/C_BISMT
WP1_400	139669	A	PF1 167181

If we found a result and if the source system from material is different from the start of old material number, it's a cross system, so the material cross system for material WP1_400/139669 is PF1_020/167181.
And it means also the cross system for material PF1_020/167181 it's WP1_400/139669 (if the reverse (old material number not filled for example) is not directly present in master data c_matnr2, we need to calculate it during the transformation between c_matnr2 and ABMD001).
Result in ABMD001:

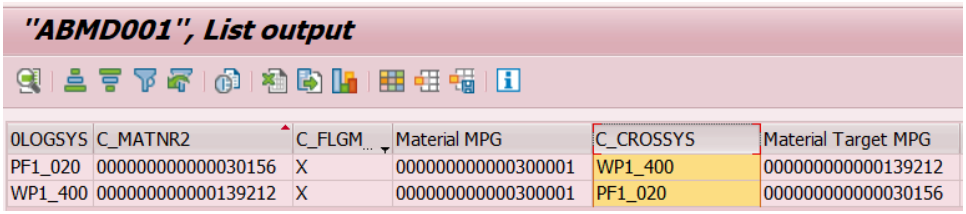
"ABMD001", List output

0LOGSYS	C_MATNR2	C_CROSSYS	C_FLGBISM	Old Material Target
WP1_400	000000000000139669	PF1_020	X	000000000000167181
PF1_020	000000000000167181	WP1_400	X	000000000000139669

ii. *Rule with MPG data.*
We extract table Z4R_MDG_T_TARSYS from MPG system in propagation aDSO APMD001 In this table there is the conversion of the material by system.
blocked URL

In This exemple PF1(ERPSOLV) 000000000000030156 is related to WP1 (ERPRCS) 000000000000139212 material code because both are link to the same MPG material code 000000000000300001
For MPG material code it's a perfect example, it means the material 30156 from PF1 has as cross system material 139212 from WP1 and vice versa, 139212 from WP1 has as cross system metrial 20156 from PF1.

In business aDSO ABMD001 we will have this result:



"ABMD001", List output

0LOGSYS	C_MATNR2	C_FLGM...	Material MPG	C_CROSSYS	Material Target MPG
PF1_020	000000000000030156	X	000000000000300001	WP1_400	000000000000139212
WP1_400	000000000000139212	X	000000000000300001	PF1_020	000000000000030156

When in table table Z4R_MDG_T_TARSYS from MPG system we have only one line by MPG Material, it means there is no cross system.

i. Flat file for manual update.
It's possible to update directly data in aDSO ABMD001 to force some cross system. It may be necessary in cases where master data c_batch2 is not well completed.
For that it's necessary to create a flat file (xls fromat) with this columns:

A	B	C	D
Source system	Material	Target ERP System	Target ERP Material Number
PF1_020	30156	WP1_400	139212
WP1_400	139212	PF1_020	139212

Load the file in aDSO APMD002 with help of Infopackage IP: DTS_APMD002_0001 - PC_FILE. Then load aDSO ABMD01.

c. **Module Function**
A module function exists to know if a material is cross system or not:
Z_CROSS_SYSTEM_MATERIAL
It's necessary to have in enter parameters the logsys and material and also the target system of the cross system. needed.
If there is a cross system associatem, the module function will find it and tell us the orign of the cross system (MPG, old material or flat file).

```
Test for function group       ZBW_CROSS_SYSTEM
Function module               Z_CROSS_SYSTEM_MATERIAL
Uppercase/Lowercase           ☐

Runtime:         3.996 Microseconds
```

Import parameters	Value
I_P_LOGSYS	QF1_020
I_P_MATNR2	15589
I_P_LOGSYS_TARGET	WQ1_400

Export parameters	Value
LOGSYS_CROSS_SYST	WQ1_400
MATNR2_CROSS_SYST	139179
ORIGIN_CROSS_SYST	MPG

In case of multiple "origin" and the multiple cross system result, the priority is cross system from flat file, then from MPG system and at the end the cross system from old material number.

4. **Load PF1 (UPIS) Data - Propagation - aDSO APCOMS02**

MS: UPIS Stock Valuation (PF1)	APCOMS02
TRSF: DTS_Z1K_BATCH3 (PF1) -> APCOMS02	0SPFOBZXFGU7ZX8OKEVFXAHTFIN12HI2
UPIS Stock Valuation	DTS_Z1K_BATCH3
UPIS Stock Valuation - Full	ZPAK_5KY9PQR49SOJC3Q5QHXRQ0OQ2
UPIS Stock Valuation - Full - M-1	ZPAK_5LUKOR2PSKBMV4IYOXNNCPYJU
UPIS Stock Valuation - Init -> delta	ZPAK_5JCN9K8AMXQI34HKL8ZS7XLPM

Extraction from PF1 system table Z1K_BATCH3 (normally once a month, extraction of previous month in full mode).

5. **Calculate the cross sytem Model S - PF1 (UPIS data) - aDSO ABCOMS03**

MS: UPIS Stock Valuation (PF1) Cross System	ABCOMS03
TRSF: IFS_APCOMS02_ABCOMS03 -> ABCOMS03	0T3JTC2GVIDGATAZYWFN4CY0EXG2RJFV
Infosource APCOMS02 -> ABCOMS03	IFS_APCOMS02_ABCOMS03
TRSF: APCOMS02 -> IFS_APCOMS02_ABCOMS03	0KS08P2JNH8BEW3VXU2P0MSNGMTVD3YC
MS: UPIS Stock Valuation (PF1)	APCOMS02

The goal in this aDSO is to have the data considered as cross system. If a line is cross system, it means we need to take the cost unit not from the source (PF1) but from the cross system source system, so WP1 (ODS_PCP4).

In transformation before the infosource, we will extract from aDSO APCOMS02 the data where the C_BATCH2 is considered like cross system (C_FLGCROS = X) , and where the module function Z_CROSS_SYSTEM_MATERIAL find a cross system material in result.

At this step, we have only data considered as cross system, we do a lookup in ods ODS_PCP4 with links:

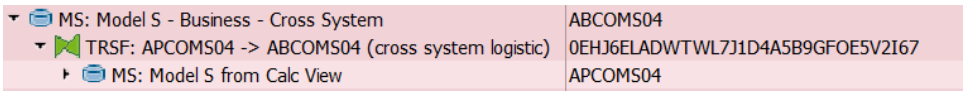
WP1 data (ods_pcp4)		PF1 data (APCOMS02)
material	=	material cross system
plant	=	manufacturing plant cross system
costing date	=	first day calmonth UPIS
cost var	<>	ZIP

If we found a correspondance in ODS_PCP4 we can continue, if not, the records will be not loaded in this aDSO.

In the second transformation (after the infosource) we will convert in EUR the cost units WP1 collected from ODS_PCP4. And then we can calculate the Integrated Margin (CP, CNP, AMO, Duty and Freight):

IM = Sotck from PF1 * cost unit from WP1.
Example File.

6. **Calculate the cross sytem Model S - WP1 (Logistic and Transit data) - aDSO ABCOMS04**



First, we need to define the records from APCOMS04 considered like cross system in function of C_BATCH2 (C_FLGCROS = X) and where the module function Z_CROSS_SYSTEM_MATERIAL return a valid result.
When we have a valid material cross system (PF1) we do a lookup in aDSO ABCOMS2 with links:

PF1 data (ABCOMS02)		WP1 data (APCOMS04)
material	=	material cross system
PLANT	=	manufacturing plant cross system
calmonth	=	Calmonth - 1

To take the stock and cost unit KPI from ABCOMS02 aggregated by Material, Plant and Month (if we don't find result, we don't load the record in ABCOMS04).
When it's done, we calculate the Integrated Margin KPI from PF1:

PF1_CP_IM = EUR_CONSO_VAR/ TOTQTY

PF1_CNP_IM =EUR_CONSO_FIX/ TOTQTY

PF1_AMO_IM = eur_CONSO_DEP / TOTQTY

PF1_FREIGHT_IM = EUR_DUTY / TOTQTY

PF1_DUTY_IM = EUR_TST / TOTQTY

Then, we can take the sotck from WP1 associated (with help of material cross system and manufacturing plant associated) to calculate the IM valuation:

Sotck from WP1 * IM from PF1.

Example file.

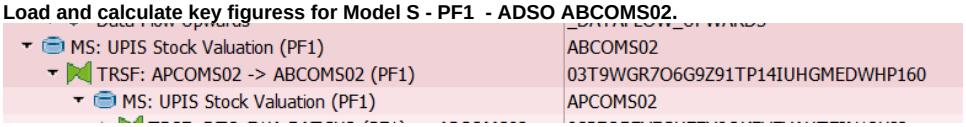
7. **Load and calculate key figuress for Model S - WP1 (Logistic and Transit data) - ADSO ABCOMS01.**
   Last step is to caculate all key figures for Model S report, It's done in transformation TRSF: APCOMS04 -> ADSO ABCOMS01.

   In start routine, we delete some lines according to their company code or plant. Company code and plant to exclude are defined in master data global filter (C_GLBFILT) in stream "MODEL S" and rule "PLANT" or "COMPCDE":

   During the loading, the transformation will convert the unit and currencies key figures with help of conversion type UOM_MATNR2 (units) and CTK_MODELS (currencies).

   The important point here is to normally, the cross system data (calculated in ABCOMS04) will be not loaded in this aDSO.

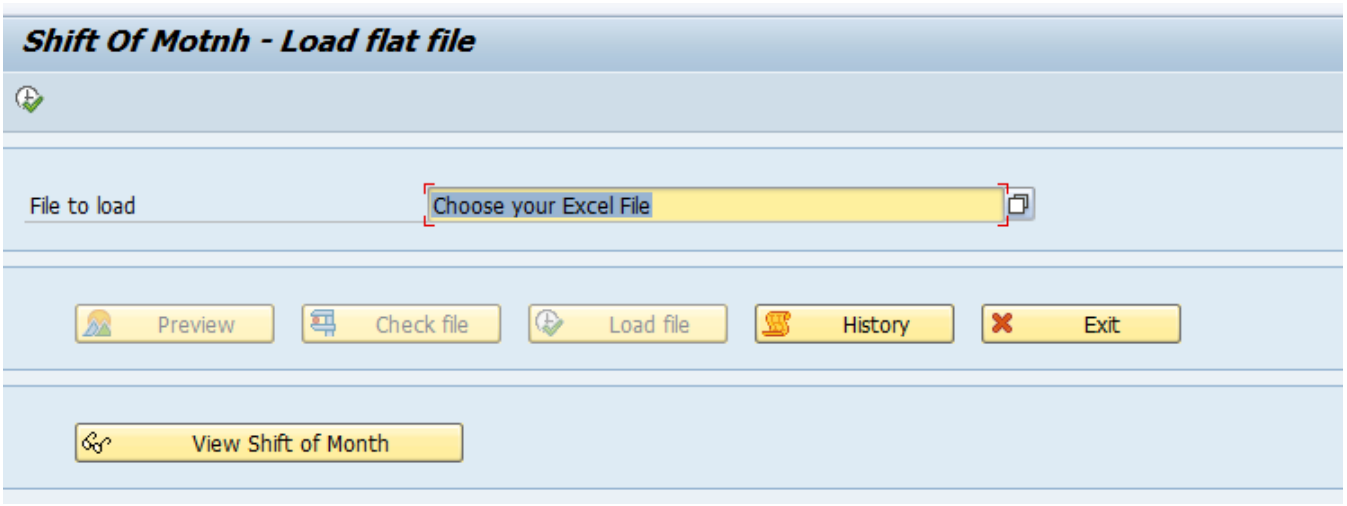
8. **Load and calculate key figuress for Model S - PF1 - ADSO ABCOMS02.**



   The important point here is to normally, the cross system data (calculated in ABCOMS03) will be not loaded in this aDSO.

Particular shift of month definition:

Transaction ZBW_SHIFT_MONTH (linked with program ZBW_LOAD_FLAT_FILE_MODELS):



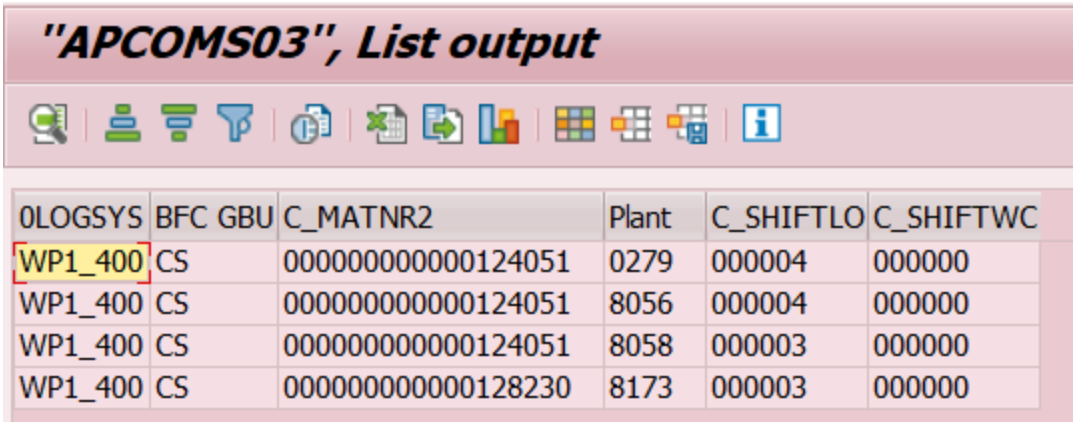
Shifts of month will be used in calculations views CV_FMCO_CO_MS_DBFIGL13_REFMONTH and CV_FMCO_CO_MS_APCOMS01_REFMONTH.

They are used to manage some exceptions when we define the month on which we will read the data to retrieve the data Transit and Logistic.

Flat file template (xls format):

Source system	GBU	Plant	Material	Shift of month logisitic	Shift of month Transit
WP1_400	CS	8058	124051	3	0
WP1_400	CS	8173	128230	3	0
WP1_400	CS	0279	124051	4	0
WP1_400	CS	8056	124051	4	0

Data in production:



0LOGSYS	BFC GBU	C_MATNR2	Plant	C_SHIFTLO	C_SHIFTWC
WP1_400	CS	000000000000124051	0279	000004	000000
WP1_400	CS	000000000000124051	8056	000004	000000
WP1_400	CS	000000000000124051	8058	000003	000000
WP1_400	CS	000000000000128230	8173	000003	000000

At least Source system, GBU, shift of month Logistic and shift of month Transit are mandatory.

Procedure to use the transaction:

Intermediate Flag

An "intermediate flag" object was added in Model S data flow.

The goal is to from manufacturing plant / material and Reference Month Logistic or Reference Month Transit, search if the company code associated with manufacturing plant is different from at least one of company code of the plants from the BOM component.

If at least one component has a different company code compared to the finish material, intermediate flag = X.

For that, it's necessary to retrieve the BOM associated with the Material and Manufacturing plant from the Model S query.

The BOM information comes from provider ABPURM02.

Due to the rule to calculate the BOM explosion, there can be some case where component are not listed in ABPURM02, if no intermediate flag found when we used ABPURM02, we will read ODS_PCP1 for same material and plant for the lookup, and in addition where vendor is empty.

If in ODS_PCP1 we have for one finish material & plant, one component with empty vendor and with a company code different from the finish material, intermediate flag = X.

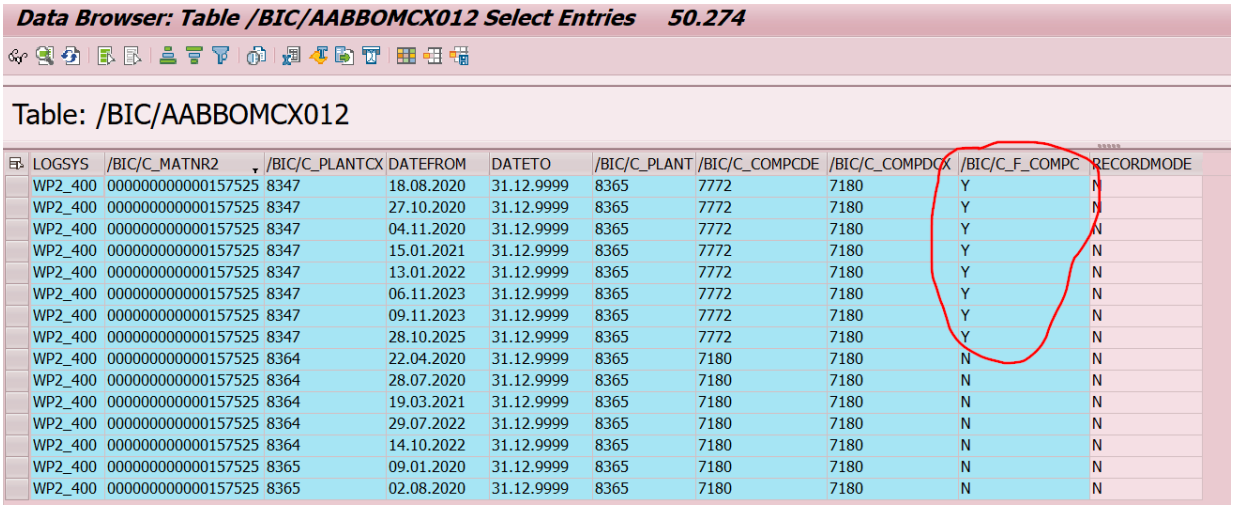
If in ODS_PCP1 we have for one finish material & plant, one component with empty vendor, but we don't find component different from the finish material, is not possible to know if we have intermediate flag, so we created field "potential intermediate flag" = X.

Quick explanation of intermediate flag calculation:

1. Find the material, manufacturing plant and date to use to make the lookup.
    a. Manufacturing plant: MANUFACTURING_PLANT
    b. Material: C_MATNR2
    c. Date: CALMON_SHI_L or CALMON_SHI_W (calmonth shift logistic or working cap, only one is filled, each time choose the one not empty.
2. Convert CALMON_SHI_L or CALMON_SHI_W into fiscal period
3. Select from BW BOM Table ABPURM02 records where
    a. Source logsys = abpurm02 logsys
    b. Source c_matnr2 = abpurm02 c_matnrp
    c. Source manufacturing_plant <> abpurm02 c_rwplt2 (not necessary to take records with same plant, we want only records with not same plant companies codes, if plants are the same, companies will be the same)
    d. Source manufacturing_plant = abpurm02 c_plantp
    e. Source fiscper (calmon_shi_l or calmon_shi_w) = abpurm02 fiscper
4. Find the company code associated to the source manufacturing plant and to the raw material plant (C_RWPLT2) from ABPURM02 table with C_PLANT master data information.
5. Compare the companies found, if there are different, intermediate flag (C_INTM) = X.

We also take into account CPX intermediate with help of flat file transferred from CPX to OBP: ABBOMCX01 "Intermediates plants and companies from CPX".

In this table (loaded by flat file "OH_SME" in /exploit/bw/integrated_margin) we have the list of finish material, plant, manufacturing plant, and companies associated, and a field to know if the finish material in CPX contain component with different company code.



Data Browser: Table /BIC/AABBOMCX012 Select Entries 50.274

Table: /BIC/AABBOMCX012

LOGSYS	/BIC/C_MATNR2	/BIC/C_PLANTCX	DATEFROM	DATETO	/BIC/C_PLANT	/BIC/C_COMPCDE	/BIC/C_COMPDCX	/BIC/C_F_COMPC	RECORDMODE
WP2_400	000000000000157525	8347	18.08.2020	31.12.9999	8365	7772	7180	Y	N
WP2_400	000000000000157525	8347	27.10.2020	31.12.9999	8365	7772	7180	Y	N
WP2_400	000000000000157525	8347	04.11.2020	31.12.9999	8365	7772	7180	Y	N
WP2_400	000000000000157525	8347	15.01.2021	31.12.9999	8365	7772	7180	Y	N
WP2_400	000000000000157525	8347	13.01.2022	31.12.9999	8365	7772	7180	Y	N
WP2_400	000000000000157525	8347	06.11.2023	31.12.9999	8365	7772	7180	Y	N
WP2_400	000000000000157525	8347	09.11.2023	31.12.9999	8365	7772	7180	Y	N
WP2_400	000000000000157525	8347	28.10.2025	31.12.9999	8365	7772	7180	Y	N
WP2_400	000000000000157525	8364	22.04.2020	31.12.9999	8365	7180	7180	N	N
WP2_400	000000000000157525	8364	28.07.2020	31.12.9999	8365	7180	7180	N	N
WP2_400	000000000000157525	8364	19.03.2021	31.12.9999	8365	7180	7180	N	N
WP2_400	000000000000157525	8364	29.07.2022	31.12.9999	8365	7180	7180	N	N
WP2_400	000000000000157525	8364	14.10.2022	31.12.9999	8365	7180	7180	N	N
WP2_400	000000000000157525	8365	09.01.2020	31.12.9999	8365	7180	7180	N	N
WP2_400	000000000000157525	8365	02.08.2020	31.12.9999	8365	7180	7180	N	N

We also used this file, and if one finish material & manuf plant are in this list + C_F_COMP = 'Y', then we put the intermediate flag = X.

Reporting

BW_QRY_ABMMIC01_0001 - BW - Material Stock for extract model S (Core Query)

BW_QRY_CPCOMS01_0001 - Stock margin elimination report (Core Query)

BW_QRY_CPCOMS01_0002 - Stock Margin Elimination report 2 (Core Query)

Dependencies with other applications

Mains data sources:

Logistic data: ADSO ABMMIC01 - "Material stock".

Transit data: DBFIGL13 - "IM from FIGL: Line Items for G/L Acct (S) - Rhodia Level 2".

Integrated Margin data: ODS_PCP4 - "CCR: Integrated margin".

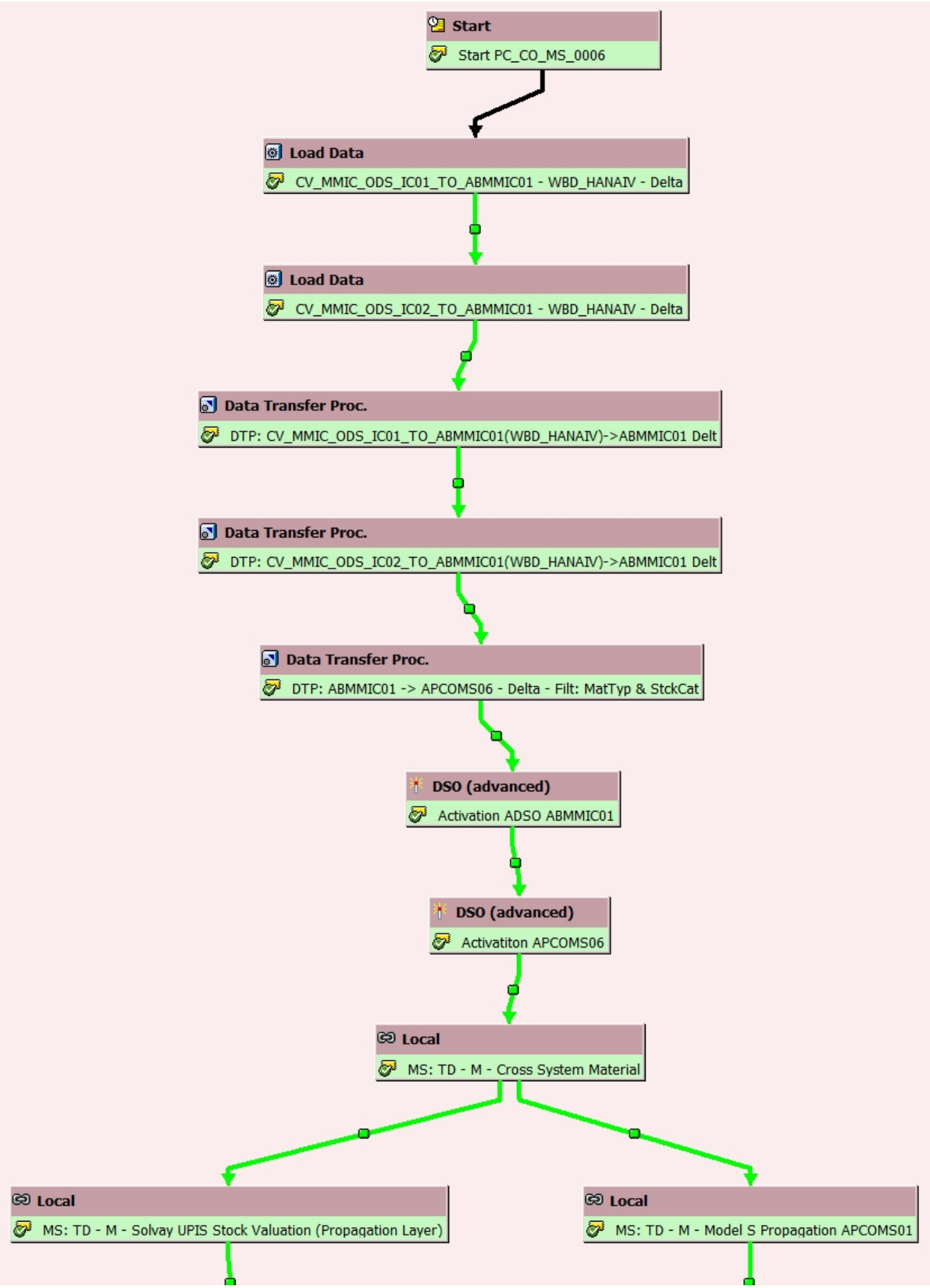
Data Loading

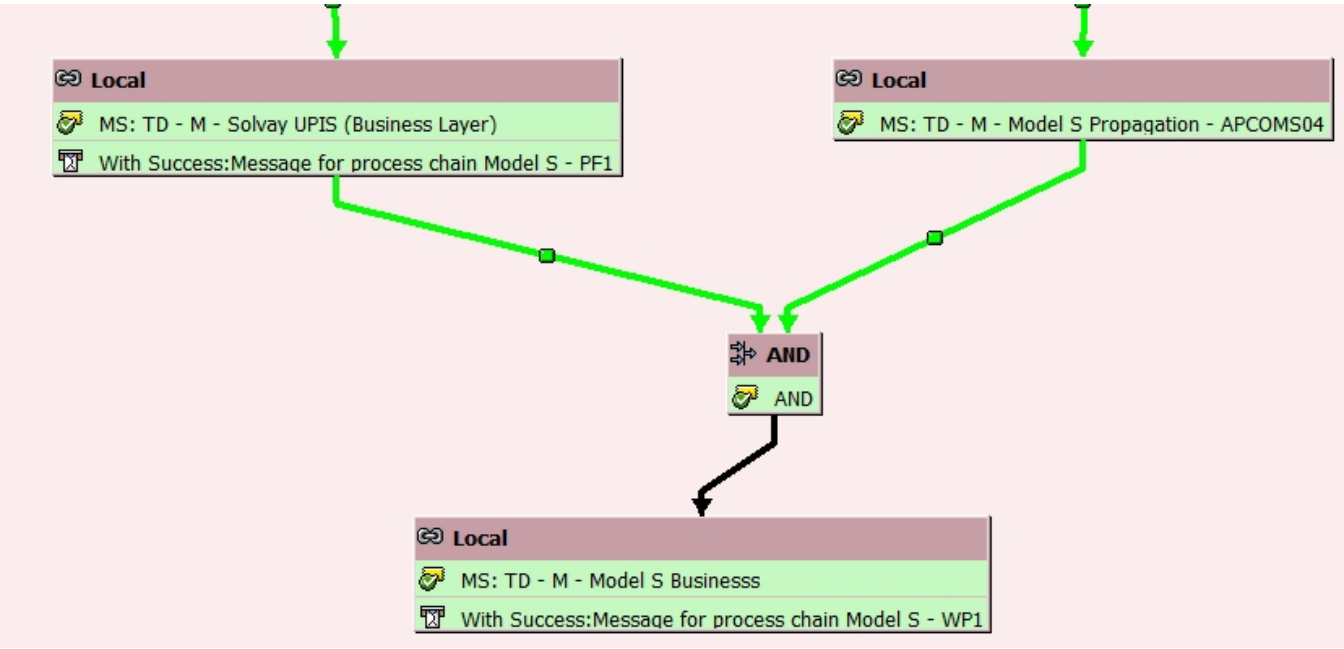
Info Providers and objects loaded

Process Chain - Technical Name	Process Chain - Description	Starter	Comment
PC_CO_MS_0006	MS: TD - M - Model S - Global	Event: Z_EVT_PC_CO_MS_0006 triggered in process chain PC_INT_MARGIN	Starts arround 3:30 AM during working day D+3.
PC_CO_MS_0008	MS: TD - M - Cross System Material	External start only (via API or metachain) Triggered in PC_CO_MS_0006	
PC_CO_MS_0002	MS: TD - M - Solvay UPIS Stock Valuation (Propagation Layer)	External start only (via API or metachain) Triggered in PC_CO_MS_0006	
PC_CO_MS_0007	MS: TD - M - Solvay UPIS (Business Layer)	External start only (via API or metachain) Triggered in PC_CO_MS_0006	
PC_CO_MS_0003	MS: TD - M - Model S Propagation APCOMS01	External start only (via API or metachain)	Load ADSO APCOMS01 with query BW_QRY_ABMMIC01_0001 in source.

PC_CO_MS_0004	MS: TD - M - Model S Propagation - APCOMS04	External start only (via API or metachain) Triggered in PC_CO_MS_0006	
PC_CO_MS_0005	MS: TD - M - Model S Businesss	External start only (via API or metachain) Triggered in PC_CO_MS_0006	Trigerred when PC_CO_MS_0005 AND PC_CO_MS_0007 are finished.
PC_CO_MS_0001	MS: TD - D - Shift of Months: Models S (Propagation Layer)	External start only (via API or metachain)	
PC_CO_MS_0009	MS - TD - M - Solvay UPIS - Monthly	Starts at 7:00 AM , 2:00 PM and 4:00 PM (french time) during working day D+3. The starter in defined at 7:00 AM and the job is job scheduling is repeated at 2:00 PM (see more details here). One more load added to start at 4:00 PM	Independant Process Chain, not in PC_CO_MS_0006. Used to take new lines cretaed in PF1 during the morning of working day D+3.

PC_CO_MS_0006 - MS: TD - M - Model S - Global:


Start
Start PC_CO_MS_0006
Load Data
CV_MMIC_ODS_IC01_TO_ABMMIC01 - WBD_HANAIV - Delta
Load Data
CV_MMIC_ODS_IC02_TO_ABMMIC01 - WBD_HANAIV - Delta
Data Transfer Proc.
DTP: CV_MMIC_ODS_IC01_TO_ABMMIC01(WBD_HANAIV)->ABMMIC01 Delt
Data Transfer Proc.
DTP: CV_MMIC_ODS_IC02_TO_ABMMIC01(WBD_HANAIV)->ABMMIC01 Delt
Data Transfer Proc.
DTP: ABMMIC01 -> APCOMS06 - Delta - Filt: MatTyp & StckCat
DSO (advanced)
Activation ADSO ABMMIC01
DSO (advanced)
Activatiton APCOMS06
Local
MS: TD - M - Cross System Material
Local
MS: TD - M - Solvay UPIS Stock Valuation (Propagation Layer)
Local
MS: TD - M - Model S Propagation APCOMS01



Loading frequency

The loading is done after the process chain PC_INT_MARGIN - "Integrated Margin 2" normally once a month in the 3th workday.

Points to be careful during the loading:

- **Be carrefull ABCOMMIC01 has key figure in summation mode. If there is an error during the calculation of Model S and the issue not comes from stocks, don't re-execute all the process chain but just the part needed. For example just UPIS part & Model S propagation and the Model S business part.**

- If there are errors during process chain integrated margin, the model s process chain will not starts, for example if PC_INT_MARGIN not scheduled (error in CPX system).
- Extraction of stock with query BW_QRY_ABMMIC01_0001 takes time.
- Infopackages to extract data from calculation view to load ADSO APCOMS04 can dump because of the calculation views which take too long or which have too large volumes. It's why there are several IP for logistic data. In case of errors it will be necessary to extract the data by smaller groups.

Average performance

The process chain PC_CO_MS_0006 tooks 2 hours (including one hour and a half to charge the adso APCOMS01 (extraction of stock with query from adso ABMMIC01).

She starts arround 3:30 am and finish at 5:30 am (French time)

Data Quality Control

Data come from SAP system. To compare data between BW and sources systems, check propagation layers.

Operational Documentation

Procedures

<Describe the recurring procedures needed to operate the application (eg. start/pause/terminate/restart the app processes, data preparation, data ingestion, ETL, data visualization, data export, other manual activities)>

Scheduling

<Describe the scheduling in place for the application (eg. existing jobs, trigger time/event based, dependencies)>

Monitoring

<Describe the monitoring checks to confirm the application is performing well (eg. check the overall status, check performance metrics like runtime /data volume/memory/disk/CPU, maintain and react to alerts/notifications)>

Error Handling

<Describe how to handle errors (eg. error codes, description and respective resolution, alert users)>

Known Bugs

<List the existing bugs, its criticity, workarounds and resolution plan.>

Roadmap

<List past & future evolutions for the application (including links to MED/FSD/TSD)>