

Technical Documentation - Sales & distribution - OTIF report

- 1 Access Management
- 2 DataFlow
 - 2.1 Overview
 - 2.1.1 Multi Provider MVSDSO50 - OTIF Global - DATA FLOW
 - 2.1.2 Multi Provider MVSDSO51 - OTIF Transfer Global - DATA FLOW
 - 2.1.3 Multi Provider MVSDSO60 - MP Sales & Transfer for OTIF with Order Book dates - DATA FLOW
 - 2.2 Technical Rules on Workbench
 - 2.2.1 Business Rules
 - 2.2.1.1 Calculation Rules (DBSDSO98 / DBSDSO91 / DBSDSO60 / DBSDSO20)
 - 2.3 Reporting
 - 2.4 Main queries
 - 2.5 Queries used to load QV applications
 - 2.6 Dependencies with other applications
- 3 Data Loading
 - 3.1 Info Providers and objects loaded
 - 3.1.1 1 - Main Process Chain
 - 3.1.2 2 - Propagation Layer
 - 3.1.3 3 - Business Layer
 - 3.1.4 4 - Reporting Level
- 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_M49	SD - Sales and Distribution	Role menu
ZBI_RCS_SD_A01	Application role - Sales & Distribution	Linked to authorization role ZBI_SD
ZP2_RCS_SD_A01	Application role - Sales & Distribution	Linked to authorization role ZP2_SD

Authorization Objects

List of authorization objects mandatory for the application.

Authorization object	Explanation
CPFCTR1_2	Global Business Unit role ZR_*_CA_P05
C_COMPCCDE	Company Code role ZR_*_CA_P01
C_SAL_OFF	Sales Office role ZR_*_CA_P14
C_COMPCCDE__C_AUTHMA	Authorization scope role ZR_*_CA_P10

See also file maintained by Authorization team :

- BW Catalog of Roles: [Authorization catalog](#)
- BW Authorization: [BW Authorizations](#)

DataFlow

Overview

Multi Provider MVSDSO50 - OTIF Global - DATA FLOW

This multi provider is used to manage the OTIF indicators for all sales processes (Solvay & Rhodia)

MP Details - characteristics

MultiProvider	Techn. name / val...	Dat...	L
MP: Sales - OTIF (Global Sales)			
Object Information			
Dimensions			
Data Package	MVSDSO50P		
Time	MVSDSO50T		
Unit	MVSDSO50U		
Delivery Dates	MVSDSO50D		
Actual goods issue date	DACT_GL_DTE	DATS	06
Actual GI date (Year/Month)	C_ACT_GI	NUMC	06
Actual GI date (Year)	C_ACT_GTY	NUMC	04
Actual Delivery Date	DACT_DL_DTE	DATS	06
Requested delivery date	EDSEL_DATE	DATS	06
Material availability date	DMATV_DATE	DATS	06
Planned Goods Issue Date of	DFLD_GL_DTE	DATS	06
Customer PO date	C_FODATE	DATS	06
Transfer order confirmation da	C_EBODA	DATS	06
Organization	MVSDSO50Z		
Source System	BLGSEYS	CHAR	10
Company code	C_COMPID6	CHAR	04
Plant	C_PLANT	CHAR	04
Material	MVSDSO503		
Material	C_MATNR2	CHAR	18
Material Plant	C_MATPLT2	CHAR	18
Material group	C_MAT_GRP	CHAR	09
Commercial Product / Material	C_PROD	CHAR	18
Customer	MVSDSO504		
Sold-to party	C_SOLDP	CHAR	10
Ship-to party	C_SHIP2D	CHAR	10
Customer (Sales View)	C_CUSTAL	CHAR	10
Reason/Types	MVSDSO505		
Sales document type	C_DOCTYP2	CHAR	04
Ship date type	C_DTTYFE	CHAR	03
Del date type	C_DTYFE2	CHAR	03
Reversal Indicator	OSTORH0	CHAR	01
Inco terms part (Core)	C_INCOTRM	CHAR	03
Non OTIF Reason	C_AUGRU	CHAR	03
Sales document item category	C_ETHCAT	CHAR	04
Shipment	MVSDSO506		
Order	MVSDSO507		
Sales Org/Office	MVSDSO508		
Sales Office	C_SAL_OFF	CHAR	04
Sales Organization	C_SAL_ORG	CHAR	04
Distribution Channel	C_DISTCHN	CHAR	02
Division	C_DIVISN	CHAR	02
GBU	MVSDSO509		
ERC GBU	OPFCRTR_2	CHAR	02
Controlling area	OCO_AREA	CHAR	04
Sub-Activity 2	C_SUBACT2	CHAR	10
Sub-activity	OG_OWVE01	CHAR	10
GBR	MVSDSO50A		
GBR: Customer end-use (Ship-t)	C_GBR5	CHAR	10
GBR: Customer gbu account by	C_GBR11	CHAR	10
GBR: Customer gbu account by	C_GBR12	CHAR	10
GBR: Customer gbu account by	C_GBR13	CHAR	10
GBR: Ship-to KA	C_GBR15	CHAR	10
GBR: Customer end-use Mat Gr	C_GBR14	CHAR	10
GBR: GBU Material Group	C_GBR18	CHAR	18
OTIF Groups	MVSDSO50B		
In Full Confirmed Detailed	C_INFULCD	CHAR	05
In Full Requested Detailed	C_INFULRD	CHAR	05
OTDOR Accuracy	C_OTDORAC	CHAR	05
OTDOR Delay Group	C_OTDORDG	CHAR	01
OTDFC Accuracy	C_OTDFCAC	CHAR	05
OTDFC Delay Group	C_OTDFODG	CHAR	01
OTDLC Accuracy	C_OTDLCAC	CHAR	05
OTDLC Delay Group	C_OTDLODG	CHAR	01
OTSCR Accuracy	C_OTSCRAC	CHAR	05
OTSCR Delay Group	C_OTSORDG	CHAR	01
OTSFC Accuracy	C_OTSFAC	CHAR	05
OTSFC Delay Group	C_OTSFODG	CHAR	01
OTSLC Accuracy	C_OTSLCAC	CHAR	05
OTSLC Delay Group	C_OTSLODG	CHAR	01

Details - Key Figures

Key Figures			
Tolerance			
• Late Tolerance in days	K_TOLDAY	INT4	04
• Early Tolerance in days	K_TOLDAY2	INT4	04
• Tolerance Limit for Over Delivery	SUPPR_BND	DEC	09
• Tolerance Limit for Under Delivery	SLOWR_BND	DEC	09
Quantity			
• Desired Delivery Quantity	BREQD_QTY	QUAN	09
• Confirmed quantity	K_CONF_QTY	QUAN	09
• Actual quantity delivered (in so)	K_IDLV_QTY	QUAN	09
• Cumulative order quantity in so	K_OML_OR_QTY	QUAN	09
• Gap Quantity (order requested)	K_GAPQTY	QUAN	09
• Gap Quantity (order confirmed)	K_GAPQTY2	QUAN	09
• Gap Quantity (order requested)	K_GAPQTY3	QUAN	09
• Cumulative confirmed quantity	K_CM_CQ_QTY	QUAN	09
OTIF			
• In full Requested (FR)	K_FRFULLR	INT4	04
• In full Confirmed (FC)	K_FRFULLC	INT4	04
• On Time Delivery Customer Req.	K_OTDCR	INT4	04
• On Time Delivery First Confirma	K_OTDFC	INT4	04
• On Time Delivery Last Confirma	K_OTDLC	INT4	04
• On Time Shipment Customer R	K_OTSCR	INT4	04
• On Time Shipment First Confir	K_OTSFC	INT4	04
• On Time Shipment Last Confir	K_OTSLC	INT4	04
Days			
• SO: Lead Time (Cust.PO x req.)	K_SOLEAD01	DEC	09
• SO: Lead Time (Cust.PO x ach)	K_SOLEAD02	DEC	09
• SO: Lead Time (Cust.PO x req.)	K_SOLEAD03	DEC	09
• SO: Lead Time (Cust.PO x ach)	K_SOLEAD04	DEC	09
• OTDCR Delay days	K_OTDCRDD	INT4	04
• OTDFC Delay days	K_OTDFCDD	INT4	04
• OTDLC Delay days	K_OTDLCDD	INT4	04
• OTSCR Delay days	K_OTSCRDD	INT4	04
• OTSFC Delay days	K_OTSFCDD	INT4	04
• OTSLC Delay days	K_OTSLCDD	INT4	04
Counter			
• Counter	K_COUNTER	DEC	09

Cube CRSDSO10 - OTIF Sales (Solvay)

SO: O.T.I.F. (Solvay)		CRSDSO10
Object Information		
• Version	◆ In Process	
• Save	■ Saved	
• Revised Version	■ Active Version	
• Object Status	● Active, executable	
Settings		
Dimensions		
• Data Package	CRSDSO10P	
• Time	CRSDSO10T	
• Unit	CRSDSO10U	
• Req.Del Date	CRSDSO101	
• Organization	CRSDSO102	
• Material	CRSDSO103	
• Reason / Type	CRSDSO104	
• Customer	CRSDSO105	
• Delivery date	CRSDSO106	
• Cust PO	CRSDSO107	
• Sales Order	CRSDSO108	
• Sales Order Line	CRSDSO109	
• Shipment	CRSDSO10A	
• Dates	CRSDSO10B	
• OTIF Groups	CRSDSO10C	
• Mat.Availability	CRSDSO10D	
Navigation Attributes		
Key Figures		

Cube CRSDSO50 - OTIF Sales (Rhodia)

SO: O.T.I.F. (Rhodia)		CRSDSO50
Object Information		
• Version	◆ In Process	
• Save	■ Saved	
• Revised Version	■ Active Version	
• Object Status	● Active, executable	
Settings		
Dimensions		
• Data Package	CRSDSO50P	
• Time	CRSDSO50T	
• Unit	CRSDSO50U	
• Req.Del Date	CRSDSO501	
• Organization	CRSDSO502	
• Material	CRSDSO503	
• Reason / Type	CRSDSO504	
• Customer	CRSDSO505	
• OTIF Groups	CRSDSO506	
• Delivery date	CRSDSO507	
• Sales Order	CRSDSO508	
• Order Line	CRSDSO509	
• Shipment	CRSDSO50A	
• Dates	CRSDSO50B	
• Cust PO	CRSDSO50C	
• Mat.Availability	CRSDSO50D	
Navigation Attributes		
Key Figures		

Multi Provider MVSDSO51 - OTIF Transfer Global - DATA FLOW

This multi provider is used to manage the OTIF indicators for all sales and transfer processes (Solvay & Rhodia)

Cubes:

CRSDSO54: PO: OTIF Transfer (Rhodia)

As part of the change # 4028391,excluded return items from this cube .Added filter in below DTP's to exclude Return Items.

DTP: DBSDSO91 -> CRSDSO54 (Rhodia)

DTP: DBSDSO91 -> DBSDSO9E (Rhodia)

DTP: DBSDSO98 -> CRSDSO24 (Solvay)

DTP: DBSDSO98 -> DBSDSO9F (Solvay)

CRSDSO24: PO: OTIF Transfer (Solvay)

As part of the change # 4028391,excluded return items from this cube .Added filter in below DTP's to exclude Return Items.

DTP: DBSDSO91 -> CRSDSO54 (Rhodia)

DTP: DBSDSO91 -> DBSDSO9E (Rhodia)

Amount	AMOUNT		
Net value of the order item in document currency	NET_VALUE	CURR	69
Net value of rest to be delivered (SU)	K_TBD	CURR	69
Net value of rest excluded	K_NSOHEVL	CURR	69
Tolerance	TOLERANCE		
Late Tolerance in days	K_TOLDAV	3174	64
Early Tolerance in days	K_TOLDAY2	3174	64
Tolerance Limit for Over Delivery in %	OLUPR_BND	DEC	69
Tolerance Limit for Under Delivery in %	ULOWR_BND	DEC	69
Quantity	QUANTITY		
Actual quantity delivered (in sales units)	QDLV_QTY	QUAN	69
Cumulative order quantity in sales units	QDLV_OR_QTY	QUAN	69
Gap Quantity (order requested X delivery)	K_GAPQTY	QUAN	69
Gap Quantity (order confirmed X delivery)	K_GAPQTY2	QUAN	69
Gap Quantity (order requested X confirmed)	K_GAPQTY3	QUAN	69
Cumulative confirmed quantity in sales unit	QDLV_CF_QTY	QUAN	69
Rest to be delivered (in sales unit)	K_NORDML	QUAN	69
Not Scheduled Requested Quantity	K_NSOHERQ	QUAN	69
OTF	OTF		
In Full Requested (FR)	K_BRFULLR	3174	64
In Full Confirmed (FC)	K_BRFULLC	3174	64
On Time Delivery Customer Request (OTD CR) K_OTDCR		3174	64
On Time Delivery First Confirmation (OTD FC) K_OTDFC		3174	64
On Time Delivery Last Confirmation (OTD LC) K_OTDLC		3174	64
On Time Shipment Customer Request (OTS CR) K_OTSCR		3174	64
On Time Shipment First Confirmation (OTS FC) K_OTDFC		3174	64
On Time Shipment Last Confirmation (OTS LC) K_OTSLC		3174	64
OTD	OTD		
OTDCR Delay days	K_OTDCR00	3174	64
OTDFC Delay days	K_OTDFC00	3174	64
OTDLC Delay days	K_OTDLC00	3174	64
OTSCR Delay days	K_OTSCR00	3174	64
OTDFC Delay days	K_OTDFC00	3174	64
OTSLC Delay days	K_OTSLC00	3174	64
S0: Lead Time (Cut:PO x Req:Req:Avail)	K_S0LEAD1	DEC	69
S0: Lead Time (Cut:PO x achieved G)	K_S0LEAD2	DEC	69
S0: Lead Time (Cut:PO x Req: delivery)	K_S0LEAD3	DEC	69
S0: Lead Time (Cut:PO x achieved delivery)	K_S0LEAD4	DEC	69
S0: Lead Time Requested (Creation X bit cut)	K_S0LEAD5	DEC	69
S0: Lead Time PostComm (Creation X achieved)	K_S0LEAD6	DEC	69

Technical Rules on Workbench

Main documentation:

Preparation

Design

Deployment

Business Rules

Get the GBU code

For Rhodia

Using distribution channel and division from material, get the IECRA code in master data C_CDCA

```

** NEW for subactivity (IECRA)
UNASSIGN <fs_cdca>.
READ TABLE itb_cdca ASSIGNING <fs_cdca>
WITH KEY c_cdca = <result_fields>-/bic/c_distchn
              division = <result_fields>-/bic/c_divisn.
IF sy-subrc = 0
AND <fs_cdca> IS ASSIGNED.
  <result_fields>-g_cwwe01 = <fs_cdca>-g_cwwe01.
ENDIF.

```

Using IECRA code, get the sub-activity in master data g_cwwe01

```

UNASSIGN <fs_subact>.
READ TABLE itb_subact ASSIGNING <fs_subact>
WITH KEY g_cwwe01 = <result_fields>-g_cwwe01.
IF sy-subrc = 0
AND <fs_subact> IS ASSIGNED.
  <result_fields>-/bic/cpfctrl_2 = <fs_subact>-cpfctrl_2.
ENDIF.

```

For Solvay

using business area, get the technical business area in master data c_techba

```

READ TABLE itb_gbu INTO itb_gbu_w
WITH KEY c_techba = <result_fields>-/bic/c_techba.

IF sy-subrc EQ 0.
  IF itb_gbu_w-cpfctr1_2 IS NOT INITIAL.
    <result_fields>-/bic/cpfctr1_2 = itb_gbu_w-cpfctr1_2.
  ENDIF.

  IF itb_gbu_w-c_rest_ba IS NOT INITIAL.

    <result_fields>-g_cwwe01 = itb_gbu_w-c_rest_ba .

    CALL FUNCTION 'CONVERSION_EXIT_ALPHA_INPUT'
      EXPORTING
        input = itb_gbu_w-c_rest_ba
      IMPORTING
        output = <result_fields>-/bic/c_subact2.
  ENDIF.

```

The function above is used to fill zeros left

The code below takes the GBU code from master data sub-activity (g_cwwe01)

```

IF <result_fields>-g_cwwe01 IS NOT INITIAL.
  UNASSIGN <fs_subact>.
  READ TABLE itb_subact ASSIGNING <fs_subact>
  WITH KEY g_cwwe01 = <result_fields>-g_cwwe01.
  IF sy-subrc = 0
    AND <fs_subact> IS ASSIGNED.
    <result_fields>-/bic/cpfctr1_2 = <fs_subact>-cpfctr1_2.
  ENDIF.
ENDIF.

```

Calculation Rules (DBSDSO98 / DBSDSO91 / DBSDSO60 / DBSDSO20)

Define the ship end date type according to the content of fields actual shipment, planned shipment of GI date

```

IF <result_fields>-/bic/c_dtttype NE 'PGI'.
  " Pick Actual Good Issue

  IF v_shp_eact EQ '00000000'.
    v_shp_eact = <result_fields>-dtshp_eact.
    <result_fields>-/bic/c_dtttype2 = 'ASE'.
    " Actual Shipment End
  ENDIF.

  IF v_shp_eact EQ '00000000'.
    v_shp_eact = <result_fields>-dtshp_e_pl.
    <result_fields>-/bic/c_dtttype2 = 'PSE'.
    " Planned Shipment End
  ENDIF.

  IF v_shp_eact EQ '00000000'.
    v_shp_eact = v_gi_date.
    <result_fields>-/bic/c_dtttype2 = 'AGI'. " Actual Good Issue
  ENDIF.

ENDIF.

```

Define the good issue date reference

```

IF <result_fields>-act_gi_dte NE '00000000'.
  v_gi_date = <result_fields>-act_gi_dte.
ELSE.
  IF <result_fields>-pld_gi_dte NE '00000000'.
    v_gi_date = <result_fields>-pld_gi_dte.
  ELSE.
    v_gi_date = <result_fields>-gi_date.
  ENDIF.
ENDIF.

```

Redefine the good issue date reference when the order is a pickup

```

IF <result_fields>-/bic/c_pickflg = 'X' OR
  <result_fields>-/bic/c_doctyp2 = 'KE'.

```

```

  <result_fields>-/bic/c_dctype = 'PGI'. "Picking Good Issue
  <result_fields>-/bic/c_dctype2 = 'PGI'.

  IF <result_fields>-/bic/c_ersda NE '00000000'.
* Last Transfer Order Confirmation Date
    v_gi_date = <result_fields>-/bic/c_ersda.
    v_shp_eact = <result_fields>-/bic/c_ersda.
  ELSE.
* Last Requested Good Issue Date
*   v_gi_date = <result_fields>-/bic/c_lstrqgi.
*   v_shp_eact = <result_fields>-/bic/c_lstrqgi.
* Last Actual Good Issue Date
    v_gi_date = <result_fields>-act_gi_dte.
    v_shp_eact = <result_fields>-act_gi_dte.

  ENDIF.

```

Define the achieved delivery date and achieved shipment date

```

<result_fields>-/bic/c_achdlv = v_shp_eact.
<result_fields>-/bic/c_achship = v_gi_date.

```

Determine In Full requested considering the lower and upper tolerance

```

v_tol_uppr = <result_fields>-cml_or_qty *
  ( 100 + <result_fields>-uppr_bnd ) / 100.
v_tol_lowr = <result_fields>-cml_or_qty *
  ( 100 - <result_fields>-lowr_bnd ) / 100.

IF <result_fields>-dlv_qty BETWEEN v_tol_lowr AND v_tol_uppr.
  <result_fields>-/bic/k_infullr = 1.
ELSE.
  <result_fields>-/bic/k_infullr = 0.
ENDIF.

```

Generate the in full requested detailed with terms light, heavy or yes

```

IF <result_fields>-dlv_qty < v_tol_lowr.
  <result_fields>-/bic/c_infulrd = 'Light'.
ENDIF.
IF <result_fields>-dlv_qty > v_tol_uppr.
  <result_fields>-/bic/c_infulrd = 'Heavy'.
ENDIF.
IF <result_fields>-dlv_qty BETWEEN v_tol_lowr AND v_tol_uppr.
  <result_fields>-/bic/c_infulrd = 'Yes'.
ENDIF.

```

Determine In Full confirmed measure considering the lower and upper tolerance

```

v_tol_uppr = <result_fields>-cml_cf_qty *
            ( 100 + <result_fields>-uppr_bnd ) / 100.
v_tol_lowr = <result_fields>-cml_cf_qty *
            ( 100 - <result_fields>-lowr_bnd ) / 100.

IF <result_fields>-dlv_qty BETWEEN v_tol_lowr AND v_tol_uppr.
  <result_fields>-/bic/k_infullc = 1.
ELSE.
  <result_fields>-/bic/k_infullc = 0.
ENDIF.

```

Generate the in full confirmed detailed with terms light, heavy or yes

```

IF <result_fields>-dlv_qty < v_tol_lowr.
  <result_fields>-/bic/c_infulcd = 'Light'.
ENDIF.
IF <result_fields>-dlv_qty > v_tol_uppr.
  <result_fields>-/bic/c_infulcd = 'Heavy'.
ENDIF.
IF <result_fields>-dlv_qty BETWEEN v_tol_lowr AND v_tol_uppr.
  <result_fields>-/bic/c_infulcd = 'Yes'.
ENDIF.

```

Determine the differences between delivery quantity and requested quantity

```

<result_fields>-/bic/k_gapqty   = <result_fields>-dlv_qty -
                                <result_fields>-cml_or_qty.

<result_fields>-/bic/k_gapqty2  = <result_fields>-dlv_qty -
                                <result_fields>-cml_cf_qty.

<result_fields>-/bic/k_gapqty3  = <result_fields>-cml_or_qty -
                                <result_fields>-cml_cf_qty.

```

Generate on time delivery customer request (OTD CR)

```

IF v_lstreq      EQ '00000000' OR
   v_shp_eact    EQ '00000000'.
  v_dif_date = '999'.
ELSE.
  v_dif_date = v_shp_eact - v_lstreq .
ENDIF.

IF v_dif_date BETWEEN <result_fields>-/bic/k_tolday2 AND
  <result_fields>-/bic/k_tolday.
  <result_fields>-/bic/k_otdcr = 1. "on time delivey
  <result_fields>-/bic/c_otdcrac = 'OTD'.

ELSE.
  <result_fields>-/bic/k_otdcr = 0. "not on time delivey

  IF v_dif_date < <result_fields>-/bic/k_tolday2.
    <result_fields>-/bic/c_otdcrac = 'EARLY'.

    IF <result_fields>-/bic/c_pickflg = 'X'.
      <result_fields>-/bic/k_otdcr = 1. "on time delivey
      <result_fields>-/bic/c_otdcrac = 'OTD'.
    ENDIF.

  ENDIF.

  IF v_dif_date > <result_fields>-/bic/k_tolday.
    <result_fields>-/bic/c_otdcrac = 'LATE'.
  ENDIF.
ENDIF.

```

Consider special NON OTIF reasons as OTIF

```

READ TABLE itb_augru INTO itb_augru_w
WITH KEY logsys = <result_fields>-logsys
      c_augru = <result_fields>-/bic/c_augru.

IF sy-subrc NE 0.
  itb_augru_w-c_otifok = ''.
ENDIF.

IF itb_augru_w-c_otifok = 'X' .

  <result_fields>-/bic/k_otdcr = 1.
  <result_fields>-/bic/k_otdcrdd = 0.
  <result_fields>-/bic/c_otdcrac = 'OTD'.
  <result_fields>-/bic/c_otdcrdg = 'A'.

  <result_fields>-/bic/k_otdfc = 1.
  <result_fields>-/bic/k_otdfcdd = 0.
  <result_fields>-/bic/c_otdfcac = 'OTD'.
  <result_fields>-/bic/c_otdfcdg = 'A'.

  <result_fields>-/bic/k_otdlc = 1.
  <result_fields>-/bic/k_otdlcdd = 0.
  <result_fields>-/bic/c_otdlcac = 'OTD'.
  <result_fields>-/bic/c_otdlcdg = 'A'.

  <result_fields>-/bic/k_otscr = 1.
  <result_fields>-/bic/k_otscrdd = 0.
  <result_fields>-/bic/c_otscrac = 'OTS'.
  <result_fields>-/bic/c_otscrddg = 'A'.

  <result_fields>-/bic/k_otsfc = 1.
  <result_fields>-/bic/k_otsfcdd = 0.
  <result_fields>-/bic/c_otsfcac = 'OTS'.
  <result_fields>-/bic/c_otsfcddg = 'A'.

  <result_fields>-/bic/k_otslc = 1.
  <result_fields>-/bic/k_otslcdd = 0.
  <result_fields>-/bic/c_otslcac = 'OTS'.
  <result_fields>-/bic/c_otslcddg = 'A'.

ENDIF.

```

Generate group of delays for OTDCR

```

IF v_dif_date < 0.
  v_dif_date = v_dif_date * ( -1 ).
ENDIF.

SELECT /bic/c_delgrp
      INTO <result_fields>-/bic/c_otdcrdg
      FROM /bic/pc_delgrp
      WHERE /bic/k_lowdays <= v_dif_date
            AND /bic/k_higdays >= v_dif_date
            AND /bic/c_delgrp <> ''.
ENDSELECT.

```

Generate OTDCR delay days

```
<result_fields>-/bic/k_otdcrdd = v_dif_date.
```

The same rules are applied for OTDFC, OTDLC, OTSCR, OTSFC, OTSLC just changing the date to be considered, see below:

For OTDFC uses the dates $v_dif_date = v_fstatp - v_shp_eact$ (difference between max first ATP and shipment end date)

For OTDFC uses the dates $v_dif_date = v_lstatp - v_shp_eact$ (difference between max last ATP and shipment end date)

Define the dates used in OTS CR (ON TIME SHIPMENT Customer request)

```
IF <result_fields>-/bic/c_ddtype NE 'PGI'.  
  <result_fields>-/bic/c_ddtype = 'AGI'.  
ENDIF.
```

```
DATA: v_lstrqgi TYPE /bic/oic_lstrqgi,  
      v_firstgi TYPE /bic/oic_firstgi,  
      v_lastgi TYPE /bic/oic_lastgi,  
      v_endcomp TYPE /bic/oik_endcomp.
```

```
v_lstrqgi = <result_fields>-/bic/c_lstrqgi.  
v_firstgi = <result_fields>-/bic/c_firstgi.  
v_lastgi = <result_fields>-/bic/c_lastgi.
```

For OTSCR uses the dates $v_dif_date = v_lstrqgi - v_gi_date$ (difference between max actual GI date and last req delivery)

For OTSFC uses the dates $v_dif_date = v_firstgi - v_gi_date$ (difference between max actual GI date and first ATP GI)

For OTSLC uses the dates $v_dif_date = v_lastgi - v_gi_date$ (difference between max actual GI date and last ATP GI)

Generate OTIF necessary comment or reason

```
CLEAR v_otfndcd.  
  
IF <result_fields>-/bic/k_otdcr = 0 OR  
  <result_fields>-/bic/k_otdfc = 0 OR  
  <result_fields>-/bic/k_otdcl = 0 OR  
  <result_fields>-/bic/k_otscr = 0 OR  
  <result_fields>-/bic/k_otsfc = 0 OR  
  <result_fields>-/bic/k_otslc = 0 OR  
  <result_fields>-/bic/k_infullc = 0 OR  
  <result_fields>-/bic/k_infullr = 0.  
  
  v_otfndcd = 'X'.  
ENDIF.  
  
IF v_otfndcd = 'X' AND  
  <result_fields>-/bic/c_augru = ''.  
  
  <result_fields>-/bic/c_otfndcd = 'X'. "'REASON CODE NEEDED'.  
ENDIF.
```

Generate NON OTIF Type

```

CASE <result_fields>-/bic/c_augru.
  WHEN '740' OR '741' OR '742' OR '743' OR '744' OR '745'.
    <result_fields>-/bic/c_notityp = 'RED'.    "REDRESS".
  WHEN '750'.
    <result_fields>-/bic/c_notityp = 'REDF'.  "REDRESS FCA".
ENDCASE.

IF <result_fields>-/bic/c_notityp IS INITIAL.
  IF v_otfndcd = 'X'.
    <result_fields>-/bic/c_notityp = 'NOFP'.  "NON OTIF-Process".
  ELSE.
    <result_fields>-/bic/c_notityp = 'OTIFOK'. "OTIF-Process OK".
  ENDIF.
ENDIF.

```

Determine LEADTIME calculations (DBSDSO9B / DBSDSO9A / DBSDSO91 / DBSDSO60 / DBSDSO20).

Leadtime 1: difference between the customer PO Date (order line level - VBKD-BSTDK), and the requested material availability date

```

IF <result_fields>-/bic/c_podate EQ '00000000' OR
  <result_fields>-matav_date EQ '00000000'.
  <result_fields>-/bic/k_solead1 = '999'.
ELSE.
  <result_fields>-/bic/k_solead1 =
    <result_fields>-matav_date - <result_fields>-/bic/c_podate.
ENDIF.

```

Leadtime 2: difference between the customer PO Date (VBKD-BSTDK), and the achieved GI date (Max Actual GI Date)

```

IF <result_fields>-/bic/c_podate EQ '00000000' OR
  <result_fields>-act_gi_dte EQ '00000000'.
  <result_fields>-/bic/k_solead2 = '999'.
ELSE.
  <result_fields>-/bic/k_solead2 =
    <result_fields>-act_gi_dte - <result_fields>-/bic/c_podate.
ENDIF.

```

Leadtime 3: difference between the customer PO Date, and the requested delivery date (Max Last Req Delivery)

```

IF <result_fields>-/bic/c_podate EQ '00000000' OR
  <result_fields>-dsdel_date EQ '00000000'.
  <result_fields>-/bic/k_solead3 = '999'.
ELSE.
  <result_fields>-/bic/k_solead3 =
    <result_fields>-dsdel_date -
    <result_fields>-/bic/c_podate.
ENDIF.

```

Leadtime 4: difference between the customer PO Date and the achieved delivery date (Achieved Delivery Date)

```

IF <result_fields>-/bic/c_podate EQ '00000000' OR
  <result_fields>-act_dl_dte EQ '00000000'.
  <result_fields>-/bic/k_solead4 = '999'.
ELSE.
  <result_fields>-/bic/k_solead4 =
    <result_fields>-act_dl_dte - <result_fields>-/bic/c_podate.
ENDIF.

```

Leadtime 5 : LeadTime Requested = Order line creation date - order line last customer req Goods Issue date

```

IF <fs_order>-createdon EQ '00000000' OR
  <result_fields>-/bic/c_lstrggi EQ '00000000'.
  <result_fields>-/bic/k_solead5 = '999'.
ELSE.
  <result_fields>-/bic/k_solead5 =
    <result_fields>-/bic/c_lstrggi - <fs_order>-createdon .
ENDIF.

```

Leadtime 6 : LeadTime FirstCommitted = Order line creation date - schedule line 1st ATP (confirmed) GI date

```

IF <fs_order>-createdon EQ '00000000' OR
  <result_fields>-/bic/c_fstatp EQ '00000000'.
  <result_fields>-/bic/k_solead6 = '999'.
ELSE.
  <result_fields>-/bic/k_solead6 =
    <result_fields>-/bic/c_fstatp - <fs_order>-createdon.
ENDIF.

```

Leadtime 7 : LeadTime Actual = Order line creation date - actual GI date

```

IF <fs_order>-createdon EQ '00000000' OR
  v_gi_date EQ '00000000'.
  <result_fields>-/bic/k_solead7 = '999'.
ELSE.
  <result_fields>-/bic/k_solead7 =
    v_gi_date - <fs_order>-createdon.
ENDIF.

```

Leatime 8 : Last Req GI date - Customer PO

```

IF <result_fields>-/bic/c_podate EQ '00000000' OR
  <result_fields>-/bic/c_lstrqgi EQ '00000000'.
  <result_fields>-/bic/k_solead8 = '999'.
ELSE.
  <result_fields>-/bic/k_solead8 =
    <result_fields>-/bic/c_lstrqgi -
    <result_fields>-/bic/c_podate.
ENDIF.

```

Leatime 9 : First Req GI date - Customer PO

```

IF <result_fields>-/bic/c_podate EQ '00000000' OR
  <fs_order>-c_fstrqgi EQ '00000000'.
  <result_fields>-/bic/k_solead9 = '999'.
ELSE.
  <result_fields>-/bic/k_solead9 =
    <fs_order>-c_fstrqgi - <result_fields>-/bic/c_podate.
ENDIF.
ENDIF.

```

Reporting

Type	Tech Name	Description	Formula
CKF	BW_CKF_MVSDSO50_0001	Confirmed Qty (PUQ)	CML_CF_QTY
CKF	BW_CKF_MVSDSO50_0002	Order Qty (PUQ)	CML_OR_QTY
CKF	BW_CKF_MVSDSO50_0003	Delivery Qty (PUQ)	DLV_QTY
CKF	BW_CKF_MVSDSO50_0010	In Full	IF K_INFULLR = 1 AND K_INFULLC = 1, then K_COUNTER
CKF	BW_CKF_MVSDSO50_0011	Not In Full	IF K_INFULLR = 0 OR K_INFULLC = 0, then 0
CKF	BW_CKF_MVSDSO50_0012	Not in Full but OTDCR	IF K_INFULLR = 0 AND K_OTDCR = 1, then K_COUNTER
CKF	BW_CKF_MVSDSO50_0013	In Full & OTDCR	IF K_INFULLR = 1 AND K_OTDCR = 1, then K_COUNTER
CKF	BW_CKF_MVSDSO50_0014	In Full & OTDFC	IF K_INFULLR = 1 AND K_OTDFC = 1, then K_COUNTER
CKF	BW_CKF_MVSDSO50_0015	In Full & OTDLC	IF K_INFULLR = 1 AND K_OTDLC = 1, then K_COUNTER
CKF	BW_CKF_MVSDSO50_0016	Not In Full but OTDLC	IF K_INFULLR = 0 AND K_OTDLC = 1, then K_COUNTER
CKF	BW_CKF_MVSDSO50_0017	Not in Full but OTDFC	IF K_INFULLR = 0 AND K_OTDFC = 1, then K_COUNTER
CKF	BW_CKF_MVSDSO50_0018	In Full & OTSCR	IF K_INFULLR = 1 AND K_OTSCR = 1, then K_COUNTER
CKF	BW_CKF_MVSDSO50_0019	In Full & OTSFC	IF K_INFULLR = 1 AND K_OTSFC = 1, then K_COUNTER
CKF	BW_CKF_MVSDSO50_0020	In Full & OTSLC	IF K_INFULLR = 1 AND K_OTSLC = 1, then K_COUNTER
CKF	BW_CKF_MVSDSO50_0021	Not In Full but OTSCR	IF K_INFULLR = 0 AND K_OTSCR = 1, then K_COUNTER
CKF	BW_CKF_MVSDSO50_0022	Not In Full but OTSFC	IF K_INFULLR = 0 AND K_OTSFC = 1, then K_COUNTER
CKF	BW_CKF_MVSDSO50_0023	Not In Full but OTSLC	IF K_INFULLR = 0 AND K_OTSLC = 1, then K_COUNTER

CKF	BW_CKF_MVSDSO50_0024	Not In Full Not OTDCR	IF K_INFULLR = 0 AND K_OTDCR = 0, then K_COUNTER
CKF	BW_CKF_MVSDSO50_0025	Not In Full Not OTDFC	IF K_INFULLR = 0 AND K_OTDFC = 0, then K_COUNTER
CKF	BW_CKF_MVSDSO50_0026	Not In Full Not OTDLC	IF K_INFULLR = 0 AND K_OTDLC = 0, then K_COUNTER
CKF	BW_CKF_MVSDSO50_0027	Not In Full Not OTSCR	IF K_INFULLR = 0 AND K_OTSCR = 0, then K_COUNTER
CKF	BW_CKF_MVSDSO50_0028	Not In Full Not OTSFC	IF K_INFULLR = 0 AND K_OTSFC = 0, then K_COUNTER
CKF	BW_CKF_MVSDSO50_0029	Not In Full Not OTSLC	IF K_INFULLR = 0 AND K_OTSLC = 0, then K_COUNTER
CKF	BW_CKF_MVSDSO50_0030	In Full but Not OTDCR	IF K_INFULLR = 1 AND K_OTDCR = 0, then K_COUNTER
CKF	BW_CKF_MVSDSO50_0031	In Full but Not OTDFC	IF K_INFULLR = 1 AND K_OTDFC = 0, then K_COUNTER
CKF	BW_CKF_MVSDSO50_0032	In Full but Not OTDLC	IF K_INFULLR = 1 AND K_OTDLC = 0, then K_COUNTER
CKF	BW_CKF_MVSDSO50_0033	In Full but Not OTSCR	IF K_INFULLR = 1 AND K_OTSCR = 0, then K_COUNTER
CKF	BW_CKF_MVSDSO50_0034	In Full but Not OTSFC	IF K_INFULLR = 1 AND K_OTSFC = 0, then K_COUNTER
CKF	BW_CKF_MVSDSO50_0035	In Full but Not OTSLC	IF K_INFULLR = 1 AND K_OTSLC = 0, then K_COUNTER
CKF	BW_CKF_MVSDSO50_0040	Lead Time 1 (Cust.PO X req.mat.avail)	K_SOLEAD1
CKF	BW_CKF_MVSDSO50_0041	Lead Time 2 (Cust.PO X achieved GI)	K_SOLEAD2
CKF	BW_CKF_MVSDSO50_0042	Lead Time 3 (Cust.PO X req.delivery)	K_SOLEAD3
CKF	BW_CKF_MVSDSO50_0043	Lead Time 4 (Cust.PO X achieved delivery)	K_SOLEAD4
CKF	BW_CKF_MVSDSO50_0044	Lead Time (Creation X Last Cust Req GI)	K_SOLEAD5
CKF	BW_CKF_MVSDSO50_0045	Lead Time (Creation X Sched line 1st ATP)	K_SOLEAD6
CKF	BW_CKF_MVSDSO50_0046	Lead Time (Creation X Actual GI date)	K_SOLEAD8
CKF	BW_CKF_MVSDSO50_0047	Rush Orders	IF K_SOLEAD5 < 2, THAN 1
CKF	BW_CKF_MVSDSO50_0050	OTD CR - Class 1	IF OTD CR - Class 1 = 1 and K_OTDCR = 0 THEN 1
CKF	BW_CKF_MVSDSO50_0051	OTD CR - Class 2	IF OTD CR - Class 2 = 1 and K_OTDCR = 0 THEN 1
CKF	BW_CKF_MVSDSO50_0052	OTD CR - Class 3	IF OTD CR - Class 3 = 1 and K_OTDCR = 0 THEN 1
CKF	BW_CKF_MVSDSO50_0053	OTD CR - Class 4	IF OTD CR - Class 4 = 1 and K_OTDCR = 0 THEN 1
CKF	BW_CKF_MVSDSO50_0054	OTD FC - Class 1	IF OTD FC - Class 1 = 1 and K_OTDFC = 0 THEN 1
CKF	BW_CKF_MVSDSO50_0055	OTD FC - Class 2	IF OTD FC - Class 2 = 1 and K_OTDFC = 0 THEN 1
CKF	BW_CKF_MVSDSO50_0056	OTD FC - Class 3	IF OTD FC - Class 3 = 1 and K_OTDFC = 0 THEN 1
CKF	BW_CKF_MVSDSO50_0057	OTD FC - Class 4	IF OTD FC - Class 4 = 1 and K_OTDFC = 0 THEN 1
CKF	BW_CKF_MVSDSO50_0058	OTS CR - Class 1	IF OTS CR - Class 1 = 1 and K_OTSCR = 0 THEN 1
CKF	BW_CKF_MVSDSO50_0059	OTS CR - Class 2	IF OTS CR - Class 2 = 1 and K_OTSCR = 0 THEN 1
CKF	BW_CKF_MVSDSO50_0060	OTS CR - Class 3	IF OTS CR - Class 3 = 1 and K_OTSCR = 0 THEN 1
CKF	BW_CKF_MVSDSO50_0061	OTS CR - Class 4	IF OTS CR - Class 4 = 1 and K_OTSCR = 0 THEN 1
CKF	BW_CKF_MVSDSO50_0062	OTD LC - Class 1	IF OTD LC - Class 1 = 1 and K_OTDLC = 0 THEN 1
CKF	BW_CKF_MVSDSO50_0063	OTD LC - Class 2	IF OTD LC - Class 2 = 1 and K_OTDLC = 0 THEN 1
CKF	BW_CKF_MVSDSO50_0064	OTD LC - Class 3	IF OTD LC - Class 3 = 1 and K_OTDLC = 0 THEN 1
CKF	BW_CKF_MVSDSO50_0065	OTD LC - Class 4	IF OTD LC - Class 4 = 1 and K_OTDLC = 0 THEN 1
CKF	BW_CKF_MVSDSO50_0066	OTS FC - Class	IF OTS FC - Class 1 = 1 and K_OTSFC = 0 THEN 1
CKF	BW_CKF_MVSDSO50_0067	OTS FC - Class 2	IF OTS FC - Class 2 = 1 and K_OTSFC = 0 THEN 1
CKF	BW_CKF_MVSDSO50_0068	OTS FC - Class 3	IF OTS FC - Class 3 = 1 and K_OTSFC = 0 THEN 1
CKF	BW_CKF_MVSDSO50_0069	OTS FC - Class 4	IF OTS FC - Class 4 = 1 and K_OTSFC = 0 THEN 1
CKF	BW_CKF_MVSDSO50_0070	OTS LC - Class	IF OTS LC - Class 1 = 1 and K_OTSLC = 0 THEN 1
CKF	BW_CKF_MVSDSO50_0071	OTS LC - Class 2	IF OTS LC - Class 2 = 1 and K_OTSLC = 0 THEN 1
CKF	BW_CKF_MVSDSO50_0072	OTS LC - Class 3	IF OTS LC - Class 3 = 1 and K_OTSLC = 0 THEN 1
CKF	BW_CKF_MVSDSO50_0073	OTS LC - Class 4	IF OTS LC - Class 4 = 1 and K_OTSLC = 0 THEN 1
RKF	BW_RKF_MVSDSO50_0001	# orderline with actual arrival date	C_DTTYE2 = ASE, KF=K_COUNTER
RKF	BW_RKF_MVSDSO50_0002	OTD CR - Class 1	C_OTDCRDG = A, KF=OTD CR - Class 1
RKF	BW_RKF_MVSDSO50_0003	OTD CR - Class 2	C_OTDCRDG = B, KF=OTD CR - Class 2

RKF	BW_RKF_MVSDSO50_0004	OTD CR - Class 3	C_OTDCRDG = C, KF=OTD CR - Class 3
RKF	BW_RKF_MVSDSO50_0005	OTD CR - Class 4	C_OTDCRDG = D, KF=OTD CR - Class 4
RKF	BW_RKF_MVSDSO50_0006	OTD FC - Class 1	C_OTDFCDG = A, KF=OTD CR - Class 1
RKF	BW_RKF_MVSDSO50_0007	OTD FC - Class 2	C_OTDFCDG = B, KF=OTD CR - Class 2
RKF	BW_RKF_MVSDSO50_0008	OTD FC - Class 3	C_OTDFCDG = C, KF=OTD CR - Class 3
RKF	BW_RKF_MVSDSO50_0009	OTD FC - Class 4	C_OTDFCDG = D, KF=OTD CR - Class 4
RKF	BW_RKF_MVSDSO50_0010	OTS CR - Class 1	C_OTSCRDG = A, KF=OTS CR - Class 1
RKF	BW_RKF_MVSDSO50_0011	OTS CR - Class 2	C_OTSCRDG = B, KF=OTS CR - Class 2
RKF	BW_RKF_MVSDSO50_0012	OTS CR - Class 3	C_OTSCRDG = C, KF=OTS CR - Class 3
RKF	BW_RKF_MVSDSO50_0013	OTS CR - Class 4	C_OTSCRDG = D, KF=OTS CR - Class 4
RKF	BW_RKF_MVSDSO50_0014	OTD LC - Class 1	C_OTDLCDG = A, KF=OTD LC - Class 1
RKF	BW_RKF_MVSDSO50_0015	OTD LC - Class 2	C_OTDLCDG = B, KF=OTD LC - Class 2
RKF	BW_RKF_MVSDSO50_0016	OTD LC - Class 3	C_OTDLCDG = C, KF=OTD LC - Class 3
RKF	BW_RKF_MVSDSO50_0017	OTD LC - Class 4	C_OTDLCDG = D, KF=OTD LC - Class 4
RKF	BW_RKF_MVSDSO50_0018	OTS FC - Class 1	C_OTSFCDG = A, KF=OTS FC - Class 1
RKF	BW_RKF_MVSDSO50_0019	OTS FC - Class 2	C_OTSFCDG = B, KF=OTS FC - Class 2
RKF	BW_RKF_MVSDSO50_0020	OTS FC - Class 3	C_OTSFCDG = C, KF=OTS FC - Class 3
RKF	BW_RKF_MVSDSO50_0021	OTS FC - Class 4	C_OTSFCDG = D, KF=OTS FC - Class 4
RKF	BW_RKF_MVSDSO50_0022	OTS LC - Class 1	C_OTSLCDG = A, KF=OTS LC - Class 1
RKF	BW_RKF_MVSDSO50_0023	OTS LC - Class 2	C_OTSLCDG = B, KF=OTS LC - Class 2
RKF	BW_RKF_MVSDSO50_0024	OTS LC - Class 3	C_OTSLCDG = C, KF=OTS LC - Class 3
RKF	BW_RKF_MVSDSO50_0025	OTS LC - Class 4	C_OTSLCDG = D, KF=OTS LC - Class 4

PS: same calculated and restricted key figures were created for MVSDSO51 and MVSDSO60 multi providers.

Main queries

Query	Description
BW_QRY_MVSDSO50_0001	BW Sales - OTIF - Details (core query)
BW_QRY_MVSDSO50_0002	BW Sales - OTIF dashboard (core query)
BW_QRY_MVSDSO50_0003	BW Sales - OTIF - Summary (core query)
BW_QRY_MVSDSO51_0001	BW OTIF SO&PO - Details (core query)
BW_QRY_MVSDSO51_0002	BW OTIF SO&PO - dashboard (core query)
BW_QRY_MVSDSO51_0003	BW OTIF SO&PO - Summary (core query)
BW_QRY_MVSDSO60_0001	BW Sales - OTIF & Orderbook - Details (core query)
BW_QRY_MVSDSO60_0002	BW Sales/Transfer - OTIF & Orderbook - Details (core query)
BW_QRY_MVSDSO60_0002_BCAST	Non OTIF Order-Lines of the Month (OTDCR)
BW_QRY_MVSDSO60_0003	BW Sales/Transfer - OTIF & Dynasis Query
DI_BW_QRY_MVSDSO50_0001	BW Sales - OTIF - Details (core query)
DI_BW_QRY_MVSDSO50_9001	BW Sales - OTIF - Details (Dataiku Forecast)
QV_BW_QRY_MVSDSO60_0002	QV -BW Sales/Transfer - OTIF & Orderbook - Details -
QVSCE_BW_QRY_MVSDSO50_0001	BW Sales - OTIF QV (core query)
QVSCE_BW_QRY_MVSDSO51_0001	BW OTIF (Sales & Transfer) QV (core query)

Workbook		Query
OTIF & Order Book: Sales & Purchase Order	BW_WBK_MVSDSO60_0002	BW Sales/Transfer - OTIF & Orderbook - Details (core query) BW_QRY_MVSDSO60_0002
OTIF & Order Book: Sales Order	BW_WBK_MVSDSO60_0001	BW Sales - OTIF & Orderbook - Details (core query) BW_QRY_MVSDSO60_0001
OTIF: Sales & Purchase Order	BW_WBK_MVSDSO51_0001	BW OTIF SO&PO - Details (core query) BW_QRY_MVSDSO51_0001
OTIF: Sales Order	BW_WBK_MVSDSO50_0004	BW Sales - OTIF - Details (core query) BW_QRY_MVSDSO50_0001
OTIF: Sales & Purchase Order (Summary)	BW_WBK_MVSDSO51_0003	BW Transfer & Sales - OTIF - Summary (core query) BW_QRY_MVSDSO51_0003
OTIF: Sales & Purchase Order (Synthesis)	BW_WBK_MVSDSO51_0002	BW Transfer PO OTIF - dashboard (core query) BW_QRY_MVSDSO51_0002
OTIF: Sales Order (Synthesis)	BW_WBK_MVSDSO50_0001	BW Sales - OTIF dashboard (core query) BW_QRY_MVSDSO50_0002
OTIF: Sales Order (Summary)	BW_WBK_MVSDSO50_0003	BW Sales - OTIF - summary (core query) BW_QRY_MVSDSO50_0003

Queries used to load QV applications

InfoProvider	Query	
MVSDSO50	QVBW_QRY_MVSDSO50_0001	BW Sales - OTIF details (qv query)
MVSDSO51	QVBW_QRY_MVSDSO51_0001	BW Sales - OTIF Transfer (qv query)
MVSDSO51	QVSCE_BW_QRY_MVSDSO51_0001	BW OTIF (Sales & Transfer) QV (core query)
MVSDSO50_	QVSCE_BW_QRY_MVSDSO50_0001	BW Sales - OTIF QV (core query)

SCE defines 6 different OTIF indicators:

- **OTIFD CR** : On Time Delivery in Full (Customer Request) : Concatenates In full Requested and OTD CR
- **OTIFD FC** : On Time Delivery in Full (1st Confirmed) : Concatenates In full Confirmed and OTD FC
- **OTIFD LC** : On Time Delivery in Full (Last Confirmed) : Concatenates In full Confirmed and OTD LC
- **OTIFS CR** : On Time Shipment in Full (Customer Request) : Concatenates In full Requested and OTS CR
- **OTIFS FC** : On Time Shipment in Full (1st Confirmed) : Concatenates In full Confirmed and OTS FC
- **OTIFS LC** : On Time Shipment in Full (Last Confirmed) : Concatenates In full Confirmed and OTS LC

All calculation in this query is managed at Order line level. When several schedule lines matching with one same order lines we consider the sum for quantities and maximum for dates.

1 - Rules for OTIF Calculation

1.1 - On Time Delivery "On Time Delivery: it means that the calculation is based on date of arrival, they compare an achieved arrival date with a target arrival date. Achieved Delivery Date:

Achieved Delivery Date

=If([Transportation mode]="PICK UP" Or [Order Document Type Code]="KE";[V_Max Actual GI Date];If(IsNull([V_Max End of Shipment Calculated]);[V_Max Actual GI Date];[V_Max End of Shipment Calculated]))

For "PICK UP" transportation mode, Rhodia (WP1) : Last transfer order confirmation date for the order line --> FIELD LTAP-QDATU else Last Good Issue date

For "PICK UP" transportation mode, Solvay (PF1) : Last Good Issue date

For any other transportation mode : The last "Actual Shipment end" if it exists else the last "planned shipment end" if it exists else the last "Actual good Issue"

OTD CR: If Difference between Achieved delivery date and Order line requested delivery Date is lower than – Early tolerance then "EARLY" else if Difference between Achieved delivery date and Order line requested delivery Date is greater than Late tolerance then "LATE" else "OTD"

=If([Transportation mode]="PICK UP";If(DaysBetween(Last Req Delivery);[Achieved Delivery Date])<0;0;DaysBetween(Last Req Delivery);[Achieved Delivery Date]))

OTD FC: If Difference between Achieved delivery date and Orderline 1st confirmed delivery Date is lower than – Early tolerance then "EARLY" else if Difference between Achieved delivery date and Order line 1st conf delivery Date is greater than Late tolerance then "LATE" else "OTD"

=If([Transportation mode]="PICK UP";If(DaysBetween([V_Max 1st ATP];[Achieved Delivery Date])<0;0;DaysBetween([V_Max 1st ATP];[Achieved Delivery Date]))

OTD LC: If Difference between Achieved delivery date and Order line last confirmed delivery Date is lower than – Early tolerance then "EARLY" else if Difference between Achieved delivery date and Order line last conf delivery Date is greater than Late tolerance then "LATE" else "OTD"

=If([Transportation mode]="PICK UP";If(DaysBetween([V_Max Last ATP];[Achieved Delivery Date])<0;0;DaysBetween([V_Max Last ATP];[Achieved Delivery Date]))

1.2 - On Time Shipment "On Time Shipment: it means that the calculation is based on date of good issue, they compare an achieved shipment date with a target shipment date (calculated in SAP from requested delivery date, 1 confirmed delivery date and last confirmed delivery date). Achieved Shipment Date: For "PICK UP" transportation mode : Last transfer order confirmation date for the order line --> FIELD LTAP-QDATU else Last Good Issue date For other transportation mode : The last "Actual GI" date

Achieved Shipment Date

=If([Transportation mode]="PICK UP" Or [Order Document Type Code]="KE";[V_Max Actual GI Date])

OTS CR: If Difference between Achieved shipment date and Order line requested GI Date is lower than – Early tolerance then "EARLY" else if Difference between Achieved shipment date and Order line requested GI Date is greater than Late tolerance then "LATE" else OTS.

=If([Transportation mode]="PICK UP";If(DaysBetween([V_Max Last Req GI]; "[V_Max Actual GI Date])<0;0;DaysBetween([V_Max Last Req GI]; "[V_Max Actual GI Date]

OTS FC: If Difference between Achieved shipment date and Order line 1st confirmed GI Date is lower than – Early tolerance then "EARLY" else if Difference between Achieved shipment date and Order line 1st Conf GI Date is greater than Late tolerance then "LATE" else "OTS"

=If([Transportation mode]="PICK UP";If(DaysBetween([V_Max 1st ATP GI]; "[V_Max Actual GI Date] <0;0;DaysBetween([V_Max 1st ATP GI]; [V_Max Actual GI Date])))

OTS LC: If Difference between Achieved shipment date and Order line last confirmed GI Date is lower than – Early tolerance then "EARLY" else if Difference between Achieved shipment date and Order line Last confirmed GI Date is greater than Late tolerance then "LATE" else "OTS"

=If([Transportation mode]="PICK UP";If(DaysBetween([V_Max 1st ATP GI]; "[V_Max Actual GI Date] <0;0;DaysBetween([V_Max 1st ATP GI]; [V_Max Actual GI Date])))

1.3 - Tolerances:

"Number of days based on Transportation Mode (Order Shipping Condition) OCEAN = 7 RAIL = 3 BARGE = 3 INTERMODAL = 2 ROAD = 1 AIR = 0 OTHER = 0 "

DAYS DELAY NET (APPLY THE TOLERANCE)

TOLERANCES by Shipping Conditions

AB";"AC";"AE";"AF => 7 days (ocean)

AJ => 1 day (LTL)

AK AH AI => 0 (FTL)

AU AP => 3 (Barge / Rail)

AV => 2 (multimodal)

"AL" "AW" "AX" => 0 999 days) (Pick Up)

=If([OTD CR - Delay Days]=999;999;If([OTD CR - Delay Days]=0;If(Abs([OTD CR - Delay Days])<=[Tolerance Days];0;If([OTD CR - Delay Days]<0; [Tolerance Days]+[OTD CR - Delay Days];[OTD CR - Delay Days]-[Tolerance Days]))))

1.4 - Exceptions:

"Customer Pick Up: uses Last transfer order confirmation date in calculation Customer Pick Up Early = OTIF Order Non OTIF Reason code starting with 7 = OTIF "

1.5 - IN FULL

In Full Calculation "In Full Request Order Quantity fits with Delivery Quantity In Full Confirmed Scheduled Line Confirmed Quantity fits with Delivery Quantity "

IN FULL REQUESTED – IFR

(Order line qty – Sum Delivery quantity) + tolerance that comes from the sales order

IN FULL CONFIRMED – IFC

(Sum Scheduled Confirmed Qty by sales order item – Sum Delivery quantity) + tolerance that comes from the sales order

1.6 - LEAD TIMES

7 lead-times have been defined by SCE (to be challenged for rationalisation)

Leadtime 1 : The Lead-time is the difference, in days, between the customer PO Date (order line level - VBKD-BSTDK), and the requested material availability date (Max in the scheduled line level).

=DaysBetween([V_Max Cust PO];[V_Max Req Material Availability])

Leadtime 2 : The Lead-time is the difference, in days, between the customer PO Date (VBKD-BSTDK), and the achieved GI date

=DaysBetween([V_Max Cust PO];[Max Actual GI Date])

Leadtime 3 : The Lead-time is the difference, in days, between the customer PO Date, and the requested delivery date.

=DaysBetween([V_Max Cust PO];[V_Max Last Req Delivery])

Leadtime 4 : The Lead-time is the difference, in days, between the customer PO Date, and the achieved delivery date

=DaysBetween([V_Max Cust PO];[Achieved Delivery Date])

Leadtime 5 : The Lead-time is the difference, in days, between the creation date and the last customer req Goods Issue date

Leadtime 6 : The Lead-time is the difference, in days, between the creation date and the schedule line 1st ATP (confirmed) GI date

Leadtime 7 : The Lead-time is the difference, in days, between the creation date and the actual GI date

1.7 - DOCUMENTS TYPES

The document types to be considered are:

GCTA	Standard Order
KE	Consig issue
SB	SB Third-p.dir order
SO	SO Rush Order
TA	Standard Order
TAF	Standard Order (FPI)
TAM	Delivery Order
TAV	Standard Order (VMI)
TD	TD Standard Order
TDIV	TDIV Misc Op
TSA	Telesales
ZITI	Sales Order ITI BR

ZORB	Standard Order BR
ZOUT	Other Outputs BR
ZPVA	ZPVA Ord.immed.ship.
ZVEX	Fut. Dely Invoice BR
ZVEY	Fut. Dely Shipmnt BR
NB	Standard PO
UB	Stock transport ord.

Dependencies with other applications

- It was created some master data in order to have the attributes shared with several applications: Sales Order line (C_ITM_SD), PO Line (C_ITM_GS)
- Data is consumed by Qlikview dashboard. The loadings in QV are daily. It runs a fixed scheduled, the loadings must be finished when the next start.
- There is an integration with SPRINT (purchase schedule lines) in order to take the PO schedule lines and identify all the transfers
- There is a link with TIERS in order to get the main shipment for each delivery

Data Loading

Info Providers and objects loaded

Detail of process chain, list + link between or special event done for the loading

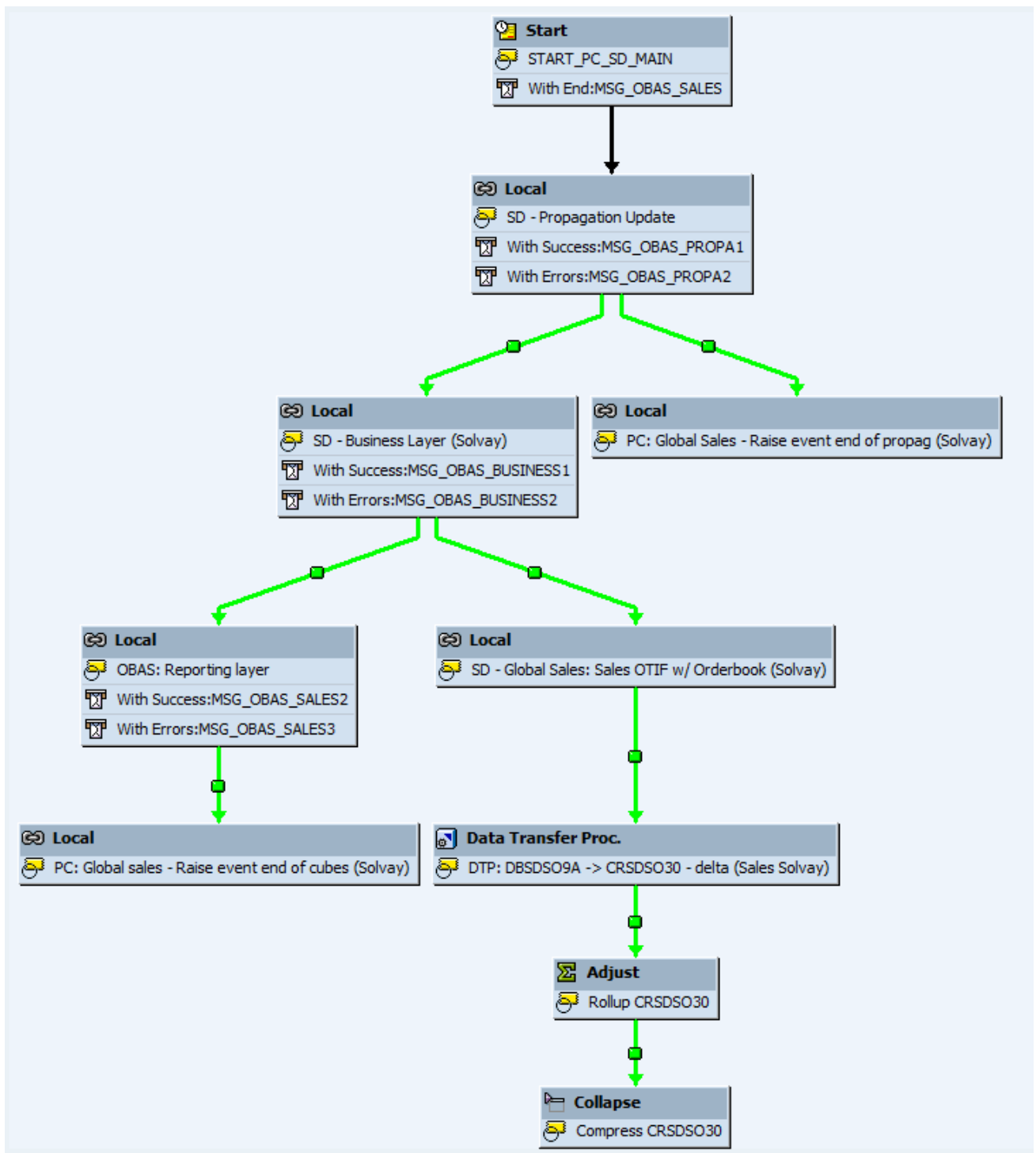
Main Process Chain	Final Provider Loading	Frequency	Time start
Global Sales: CRSDSO50 (OTIF Sales) (Rhodia) PC_GL_SALES_CRSDSO50	CRSDSO50	Daily, not weekend	Arround 05:00 am Meta chain PC_GL_SALES_REP02
Global Sales: CRSDSO10 (OTIF Sales) (Solvay) PC_GL_SALES_CRSDSO10	CRSDSO10	Daily, not weekend	Arround 02:45 am Meta chain PC_OBAS_REPORTING
OBAS: Reporting Layer Transfers Orders PC_OBAS_TRANSFERS_ODB	CRSDSO24 CRSDSO12 CRSDSO11 CRSDSO04 CRSDSO17	Daily, not weekend	Arround 06:15 am Meta chain PC_OBAS_TRANSFERS_MAIN
Global Sales: Transfers Reporting layer (Rhodia) PC_GL_SALES_PO_REPORTING	CRSDSO54 CRSDSO52 CRSDSO51 CRSDSO57 CRSDSO47 CRSDSO44	Daily, not weekend	Arround 06:45 am Meta chain PC_GL_SALES_MAIN_PO

<p>DPS Dynasys: META - D - 6.Reporting DynaSys (Dynamic KPIs)</p> <p>PC_DPS_D YNASYS_15</p>	<p>DPDYN04 DBDYN20 CRDYN11</p>	<p>Hourly, everyday with decision.</p>	<p>Hourly</p> <p>Decision linked to values in master data global filter</p> <p>Table: /BIC/PC_GLBFLT</p> <table border="1"> <thead> <tr> <th>OBJVERS</th> <th>CHANGED</th> <th>/BIC/C_DESC</th> <th>/BIC/C_SIGN</th> <th>/BIC/C_OPTION</th> <th>/BIC/C_LOW</th> <th>/BIC/C_HIGH</th> <th>/BIC/C_ACTIVE</th> </tr> </thead> <tbody> <tr><td>001 A</td><td></td><td>PC_DPS_DYNASYS_15 => 1st load (HHMMSS)</td><td>I</td><td>BT</td><td>040000</td><td>041500</td><td>N</td></tr> <tr><td>002 A</td><td></td><td>PC_DPS_DYNASYS_15 => 2nd load (HHMMSS)</td><td>I</td><td>BT</td><td>070000</td><td>071500</td><td>Y</td></tr> <tr><td>003 A</td><td></td><td>PC_DPS_DYNASYS_15 => 3rd load (HHMMSS)</td><td>I</td><td>BT</td><td>140000</td><td>141500</td><td>Y</td></tr> <tr><td>004 A</td><td></td><td>PC_DPS_DYNASYS_15 => 4th load (HHMMSS)</td><td>I</td><td>BT</td><td>180000</td><td>181500</td><td>Y</td></tr> <tr><td>005 A</td><td></td><td>PC_DPS_DYNASYS_15 => 5th load (HHMMSS)</td><td>I</td><td>BT</td><td>230000</td><td>231500</td><td>Y</td></tr> <tr><td>999 A</td><td></td><td>PC_DPS_DYNASYS_15 => For tests only !!!!</td><td>I</td><td>BT</td><td>070000</td><td>180000</td><td>N</td></tr> <tr><td>001 A</td><td></td><td>Bypass decision bloc 1 in chain PC_DPS_DYNASYS_15</td><td>I</td><td>EQ</td><td>N</td><td>N</td><td>N</td></tr> <tr><td>000 A</td><td></td><td>Sub Chain PC_DPS_DYNASYS_33 => 1st load (HHMMSS)</td><td>I</td><td>BT</td><td>080000</td><td>081500</td><td>Y</td></tr> <tr><td>001 A</td><td></td><td>Sub Chain PC_DPS_DYNASYS_33 => 1st load (HHMMSS)</td><td>I</td><td>BT</td><td>090000</td><td>091500</td><td>Y</td></tr> <tr><td>002 A</td><td></td><td>Sub Chain PC_DPS_DYNASYS_33 => 1st load (HHMMSS)</td><td>I</td><td>BT</td><td>100000</td><td>101500</td><td>Y</td></tr> <tr><td>003 A</td><td></td><td>Sub Chain PC_DPS_DYNASYS_33 => 1st load (HHMMSS)</td><td>I</td><td>BT</td><td>110000</td><td>111500</td><td>Y</td></tr> <tr><td>004 A</td><td></td><td>Sub Chain PC_DPS_DYNASYS_33 => 2nd load (HHMMSS)</td><td>I</td><td>BT</td><td>140000</td><td>141500</td><td>Y</td></tr> <tr><td>005 A</td><td></td><td>Sub Chain PC_DPS_DYNASYS_33 => 3rd load (HHMMSS)</td><td>I</td><td>BT</td><td>180000</td><td>181500</td><td>Y</td></tr> <tr><td>006 A</td><td></td><td>Sub Chain PC_DPS_DYNASYS_33 => 4th load (HHMMSS)</td><td>I</td><td>BT</td><td>230000</td><td>231500</td><td>Y</td></tr> </tbody> </table>	OBJVERS	CHANGED	/BIC/C_DESC	/BIC/C_SIGN	/BIC/C_OPTION	/BIC/C_LOW	/BIC/C_HIGH	/BIC/C_ACTIVE	001 A		PC_DPS_DYNASYS_15 => 1st load (HHMMSS)	I	BT	040000	041500	N	002 A		PC_DPS_DYNASYS_15 => 2nd load (HHMMSS)	I	BT	070000	071500	Y	003 A		PC_DPS_DYNASYS_15 => 3rd load (HHMMSS)	I	BT	140000	141500	Y	004 A		PC_DPS_DYNASYS_15 => 4th load (HHMMSS)	I	BT	180000	181500	Y	005 A		PC_DPS_DYNASYS_15 => 5th load (HHMMSS)	I	BT	230000	231500	Y	999 A		PC_DPS_DYNASYS_15 => For tests only !!!!	I	BT	070000	180000	N	001 A		Bypass decision bloc 1 in chain PC_DPS_DYNASYS_15	I	EQ	N	N	N	000 A		Sub Chain PC_DPS_DYNASYS_33 => 1st load (HHMMSS)	I	BT	080000	081500	Y	001 A		Sub Chain PC_DPS_DYNASYS_33 => 1st load (HHMMSS)	I	BT	090000	091500	Y	002 A		Sub Chain PC_DPS_DYNASYS_33 => 1st load (HHMMSS)	I	BT	100000	101500	Y	003 A		Sub Chain PC_DPS_DYNASYS_33 => 1st load (HHMMSS)	I	BT	110000	111500	Y	004 A		Sub Chain PC_DPS_DYNASYS_33 => 2nd load (HHMMSS)	I	BT	140000	141500	Y	005 A		Sub Chain PC_DPS_DYNASYS_33 => 3rd load (HHMMSS)	I	BT	180000	181500	Y	006 A		Sub Chain PC_DPS_DYNASYS_33 => 4th load (HHMMSS)	I	BT	230000	231500	Y
OBJVERS	CHANGED	/BIC/C_DESC	/BIC/C_SIGN	/BIC/C_OPTION	/BIC/C_LOW	/BIC/C_HIGH	/BIC/C_ACTIVE																																																																																																																				
001 A		PC_DPS_DYNASYS_15 => 1st load (HHMMSS)	I	BT	040000	041500	N																																																																																																																				
002 A		PC_DPS_DYNASYS_15 => 2nd load (HHMMSS)	I	BT	070000	071500	Y																																																																																																																				
003 A		PC_DPS_DYNASYS_15 => 3rd load (HHMMSS)	I	BT	140000	141500	Y																																																																																																																				
004 A		PC_DPS_DYNASYS_15 => 4th load (HHMMSS)	I	BT	180000	181500	Y																																																																																																																				
005 A		PC_DPS_DYNASYS_15 => 5th load (HHMMSS)	I	BT	230000	231500	Y																																																																																																																				
999 A		PC_DPS_DYNASYS_15 => For tests only !!!!	I	BT	070000	180000	N																																																																																																																				
001 A		Bypass decision bloc 1 in chain PC_DPS_DYNASYS_15	I	EQ	N	N	N																																																																																																																				
000 A		Sub Chain PC_DPS_DYNASYS_33 => 1st load (HHMMSS)	I	BT	080000	081500	Y																																																																																																																				
001 A		Sub Chain PC_DPS_DYNASYS_33 => 1st load (HHMMSS)	I	BT	090000	091500	Y																																																																																																																				
002 A		Sub Chain PC_DPS_DYNASYS_33 => 1st load (HHMMSS)	I	BT	100000	101500	Y																																																																																																																				
003 A		Sub Chain PC_DPS_DYNASYS_33 => 1st load (HHMMSS)	I	BT	110000	111500	Y																																																																																																																				
004 A		Sub Chain PC_DPS_DYNASYS_33 => 2nd load (HHMMSS)	I	BT	140000	141500	Y																																																																																																																				
005 A		Sub Chain PC_DPS_DYNASYS_33 => 3rd load (HHMMSS)	I	BT	180000	181500	Y																																																																																																																				
006 A		Sub Chain PC_DPS_DYNASYS_33 => 4th load (HHMMSS)	I	BT	230000	231500	Y																																																																																																																				
<p>DPS Dynasys: META - M - 6.Reporting DynaSys (Snapshots)</p> <p>PC_DPS_D YNASYS_09</p>	<p>DPDYN01 DBDYN13 CRDYN13</p>	<p>1 time per month the 6th day of month</p>	<p>06:00 am</p>																																																																																																																								
<p>Global Sales: CRSDSO30 (SO OTIF w /order book) (Solvay)</p> <p>PC_GL_SA LES_CRSD SO30</p> <p>SD: SO - D - Main Process Chain - Update</p> <p>PC_SD_MA IN</p>	<p>CRSDSO30</p>	<p>Daily, not weekend</p>	<p>Around 06:25 am</p> <p>Meta chain</p> <p>PC_OBAS_TRANSFERS_MAIN</p> <p>Around 01:55 am</p> <p>Meta chain</p> <p>RSP_DAILY</p>																																																																																																																								
<p>Global Sales: CRSDSO70 (SO OTIF w /order book) (Rhodia)</p> <p>PC_GL_SA LES_CRSD SO70</p>	<p>CRSDSO70</p>	<p>Daily, not weekend</p>	<p>Around 05:00 am & 07:00 am</p> <p>Meta chains</p> <p>PC_GLOBAL_SALES_MAIN</p> <p>PC_GL_SALES_MAIN_PO</p>																																																																																																																								

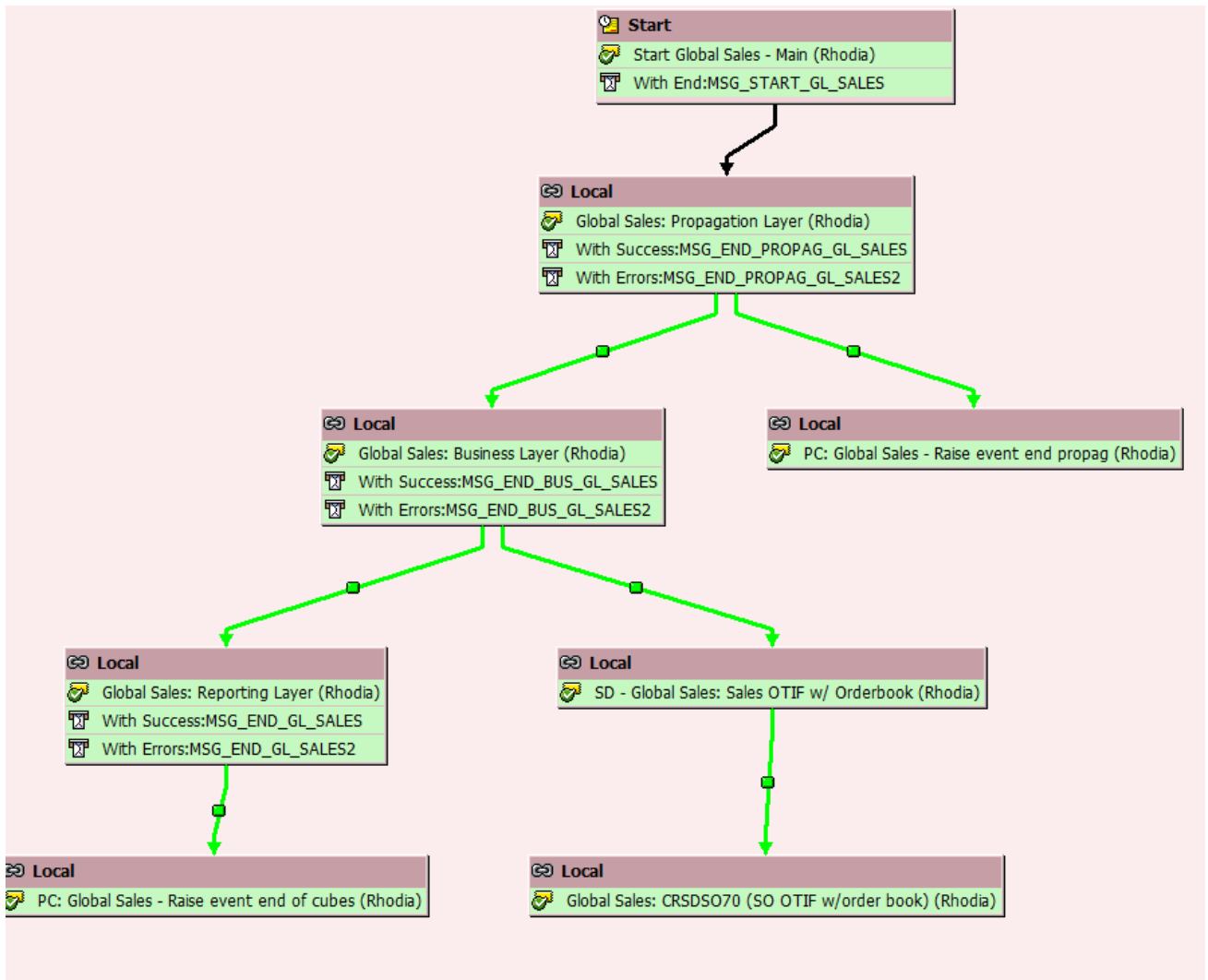
1 - Main Process Chain

This process chain is responsible to trigger and synchronize all Global Sales processes chains for Solvay and Rhodia. It is scheduled to run every workday inside the Daily process chain (RSP_DAILY) after general master data PC, around 3AM (French time) and TIERS.

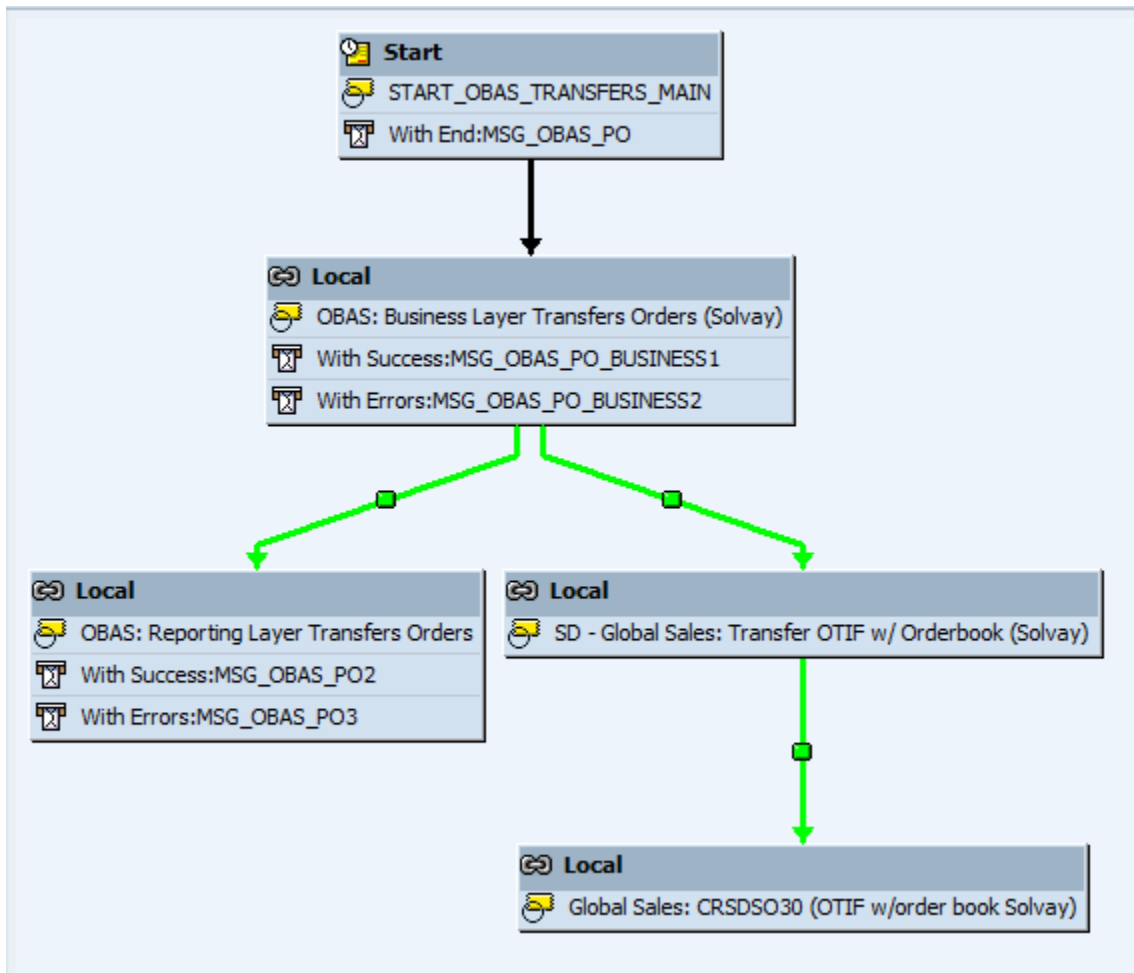
Sales - Part Solvay: PC_SD_MAIN (SD - Main Process Chain - UPTDATE DATA -)



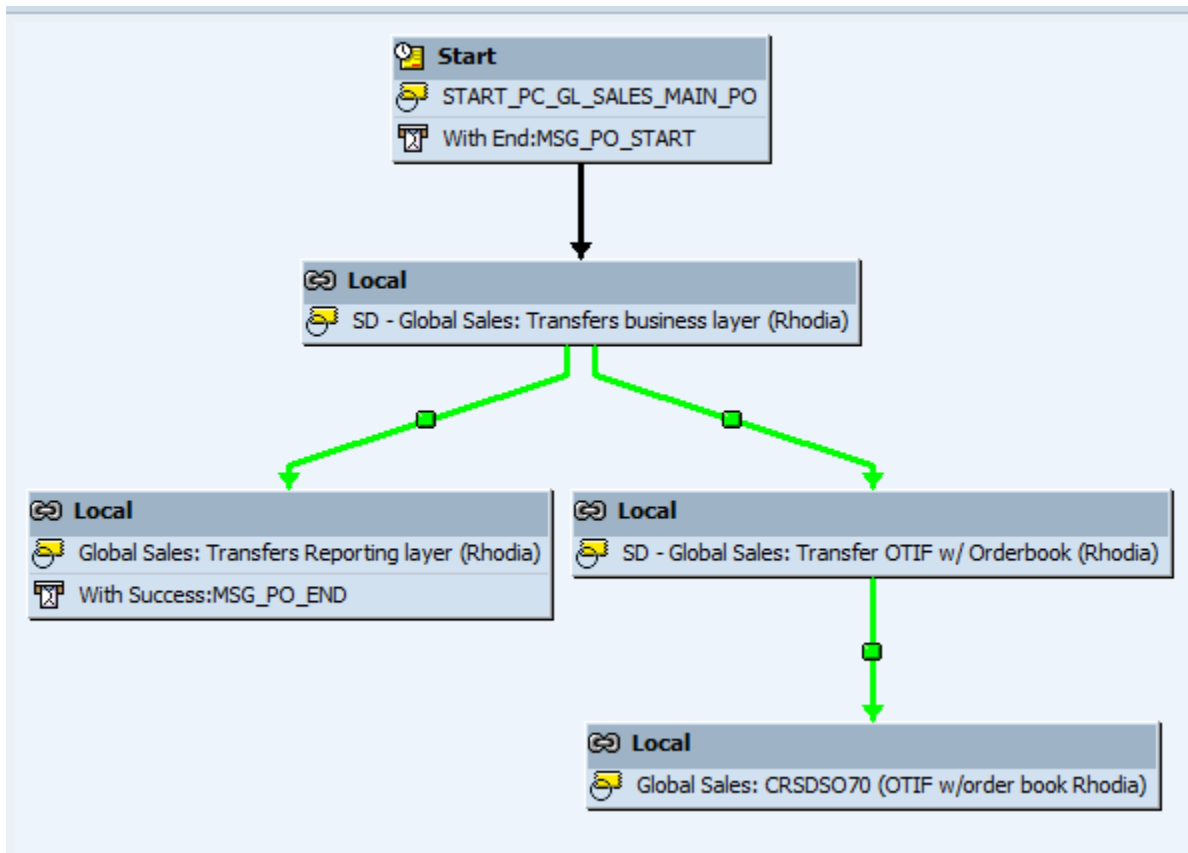
Sales - Part Rhodia : PC_GLOBAL_SALES_MAIN (PC: Global Sales: Main (Rhodia))



Transfer - Part Solvay: PC_OBAS_TRANSFERS_MAIN (OBAS: Transfers Orders Main chain)



Transfer - Part Rhodia: PC_GL_SALES_MAIN_PO (PC: Global Sales: Main transfers (Rhodia))



2 - Propagation Layer

This process chain is responsible to run the loads between source system and first layer in BW.

For OTIF process, all the processes in propagation layer are run in general Global Sales Process Chain

3 - Business Layer

This process chain is responsible to load the second and third layers from propagation layer.

Main Processes for OTIF

PC_GL_SALES_DBSDSO20 (SD - Global Sales: OTIF (Solvay)) and PC_GL_SALES_DBSDSO60 (SD - Global Sales: OTIF (Rhodia))

Start
START PC_GL_SALES_DBSDSO20 (Solvay)

Data Transfer Proc.
DTP: DBSDSO 11 -> DBSDSO20 - delta (Solvay) TAS

DataStore Data
Activate DBSDSO20

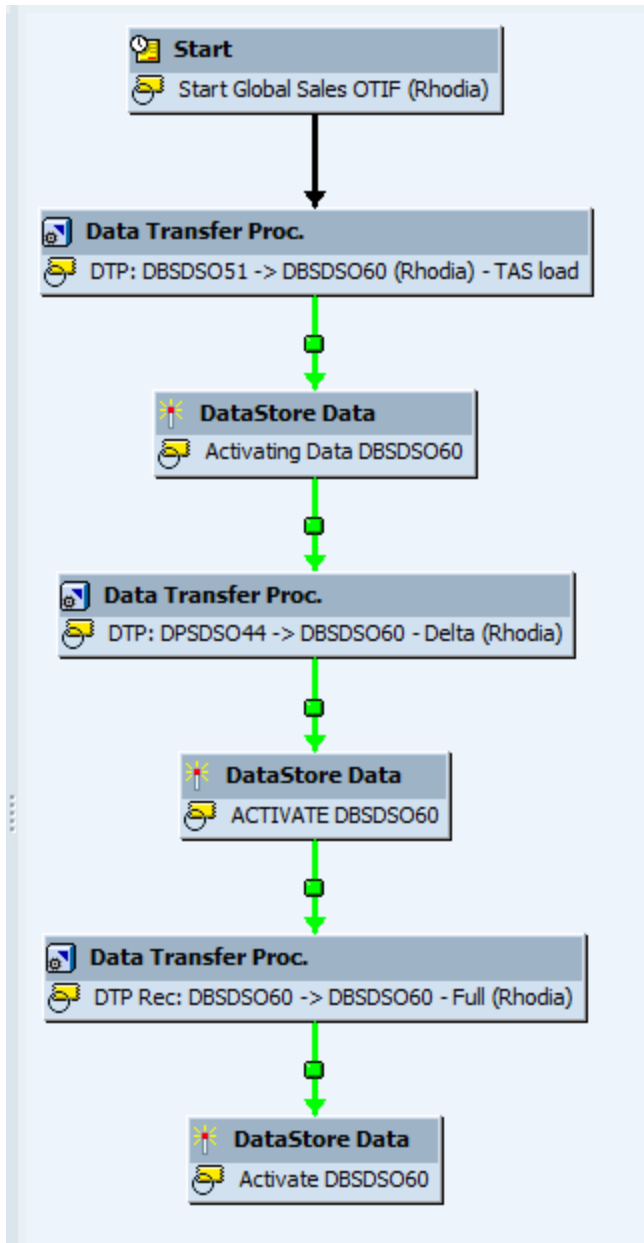
Data Transfer Proc.
DTP: DPSDSO04 -> DBSDSO20 - Delta (Solvay)

DataStore Data
ACTIVATE DBSDSO20

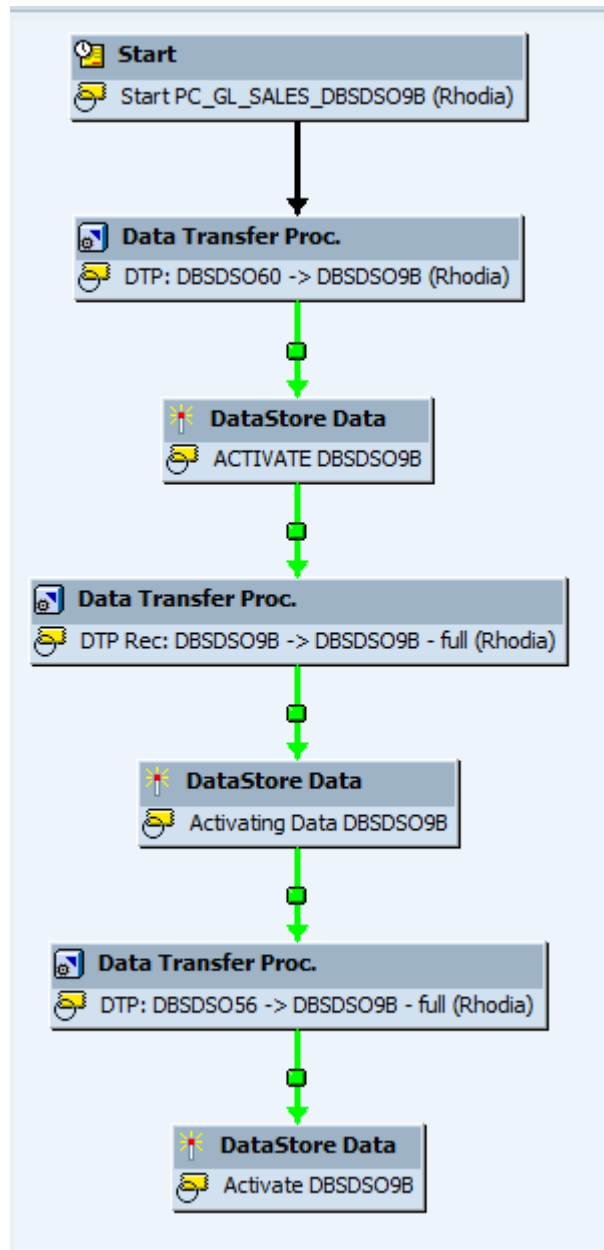
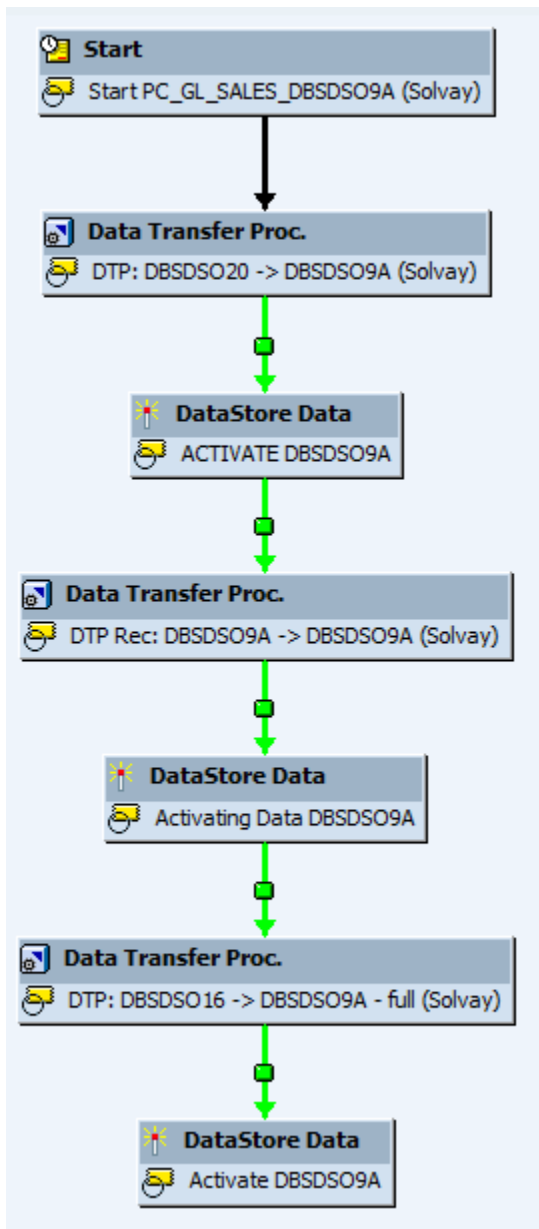
Data Transfer Proc.
DTP Rec: DBSDSO20 -> DBSDSO20 - Full (Solvay)

DataStore Data
Activating Data DBSDSO20

...



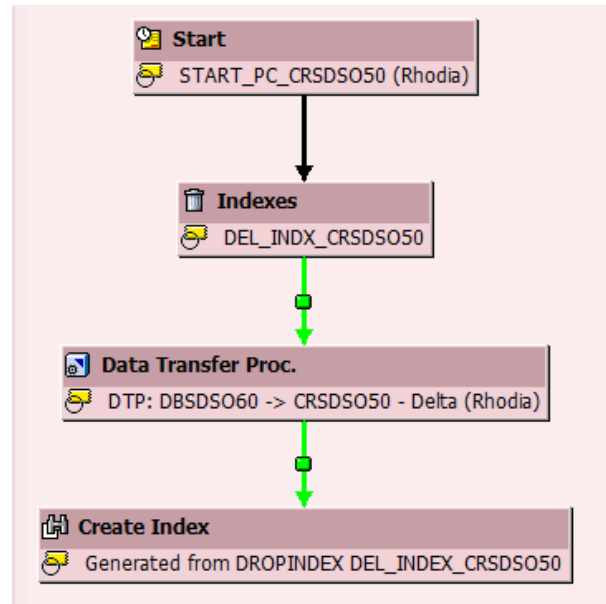
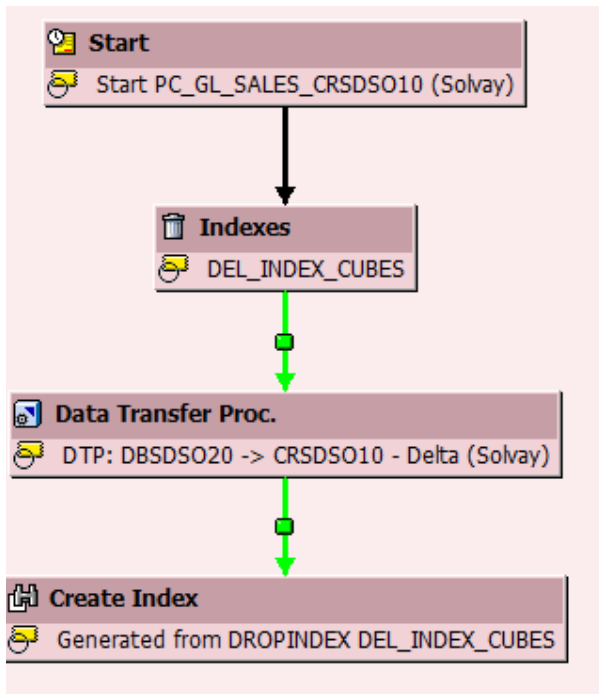
PC_GL_SALES_DBSDSO9A (SD - Global Sales: Sales OTIF w/ Orderbook (Solvay)) and PC_GL_SALES_DBSDSO9B (SD - Global Sales: Sales OTIF w/ Orderbook (Rhodia))



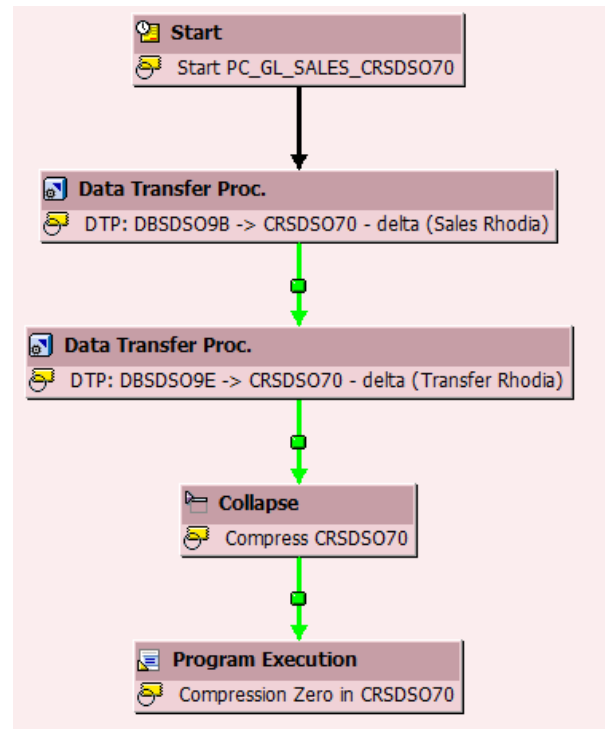
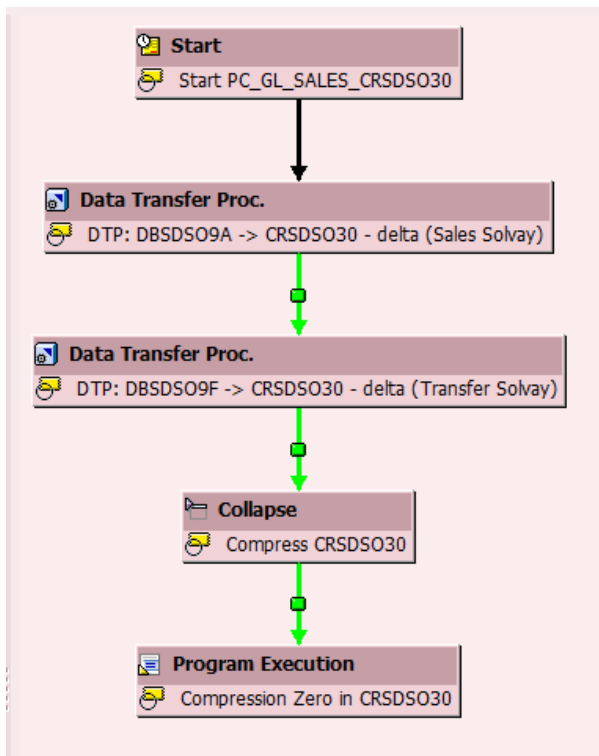
4 - Reporting Level

This process chain is responsible to load all the cubes

The most important ones are: PC_GL_SALES_CRSDSO10 (SD - Global Sales: Cube CRSDSO10 OTIF Sales (Solvay)) and PC_GL_SALES_CRSDSO50 (SD - Global Sales: Cube CRSDSO50 OTIF Sales (Rhodia))



PC_GL_SALES_CRSDSO30 (Global Sales: CRSDSO30 (OTIF w/order book Solvay)) and PC_GL_SALES_CRSDSO70 (Global Sales: CRSDSO70 (OTIF w/order book Rhodia))



Data Quality Control

Transaction	Description	Corresponding table	Comments
-------------	-------------	---------------------	----------

VT03N	Shipments	VTTK, VTTS	in ERP
VI03	Shipment Costs	VFKK, VF KP	in ERP
VL03N	Deliveries	LIKP, LIPS	in ERP
VA03	Sales Orders	VBAK, VBAP	in ERP
ME23N	Purchase Orders	EKKO, EKPO, EKBE	in ERP
MM03	Material maintenance	MARA	in ERP
RSUOM	Unit conversion	T005	in BW
RSCUR	Currency conversion	TCURR	in BW

<https://wiki.solvay.com/display/TECHREP/BW+Authorizations?src=contextnavpagetreemode>

Operational Documentation

Procedures

- When the process chain fails, the process type must be analyzed and repeated if possible.
- Sometimes a full load by period is required by user in order to synchronize all global sales data.

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

- Sometimes the unit conversion is not done in query due the unit is not mapped correctly in table UOMCMAT2 by material.

Roadmap

- The Global Sales is deployment phase and sometimes the GBU key user requests additional informations to be added in project.
- The New field ZI Partner will be available for reporting in the queries: Queries- BW_QRY_MVSDSO51_0001 and QVSCE_BW_QRY_MVSDSO51_0001 with the requirement in the FD 4699751.