Technical Documenation - Quality Management Report

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
ZBI_RCS_QM_A02	Quality Results Application - End User role	Linked to analysis role ZBI_QM
ZBI_RCS_QM_A02_PAX	Quality Results Application - For PAX	Linked to analysis role ZBI_QM_PAX
ZP2_RCS_QM_A02	Quality Results Application - End User role	Linked to analysis role ZP2_QM
ZR_RCS_CA_M14	QM - Quality Management	Role menu for queries

Authorization Objects

List of authorization objects mandatory for the application.

Authorization object	Explanation
CPFCTR1_2; C_ZSECT	GBU, role: ZR_*_CA_P05
C_PLANT__C_AUTHMA	Authorization scope role: ZR_*_CA_P00
C_PLANT	Plant, role: ZR_*_CA_P02

BW role catalog: https://docs.google.com/spreadsheets/d/10GEfKYqrT1eeTO_uHYAheL1GX7L5y_pvH0KQU64qh5I/edit#gid=131158862

DataFlow

Overview

Use the google presentation below as a template. This google presentation must be saved in the Reporting GDrive folder under the corresponding application. Then post the link to the document here.

Reporting documentation drive folder:

https://drive.google.com/drive/folders/0B0qn89R0RGdqYkZZOFZyYXlXVkE

Exemple of dataflow overview :

Template Application name DataFlow

Technical Rules on Workbench

Mains abap rules in transformations:

MVQM01 cubes associated with no data or not loaded since 2019 or 2021 doc done in december 2023).

TRSF: DPQM09 -> DBQM04 (Rhodia)

Start routine: an internal table is filled with data from APQMNCPLT (QM - Novecare Plants) for each correspondance with source system and plant from data from DPQM09.

If this internal table is empty (it means no correspondance with APQMNCPLT) source data with C_MRMITNI (Result exists) not empty are deleted.

If this internal table is not empty, and if we don't have a correspondance AND if C_MRMITNI (Result exists) is not empty, then the records are deleted.

If field BATCH is empty, he takes value from C_LSCHARG (Batch Number (QALS)).

Internal table itb_h_dpqm09 is filled with multiples rules:

```
V_DPQM09-INSP_LOT       = <SOURCE_FIELDS>-INSP_LOT.
V_DPQM09-/BIC/C_INSP_CH = <SOURCE_FIELDS>-/BIC/C_INSP_CH.
V_DPQM09-PLAN_TYPE      = <SOURCE_FIELDS>-PLAN_TYPE.
V_DPQM09-PLNNR          = <SOURCE_FIELDS>-PLNNR.
V_DPQM09-/BIC/C_PLANNOD = <SOURCE_FIELDS>-/BIC/C_PLANNOD.
V_DPQM09-/BIC/C_PHYS_SP = <SOURCE_FIELDS>-/BIC/C_PHYS_SP.
V_DPQM09-/BIC/C_DETAILE = <SOURCE_FIELDS>-/BIC/C_DETAILE.
V_DPQM09-/BIC/C_PROBENR = <SOURCE_FIELDS>-/BIC/C_PROBENR.
V_DPQM09-LOGSYS         = <SOURCE_FIELDS>-LOGSYS.

IF <SOURCE_FIELDS>-/BIC/C_SEMESNI = 'X'
  OR <SOURCE_FIELDS>-/BIC/C_SEMESNI = '*'
  OR NOT <SOURCE_FIELDS>-/BIC/C_SECODE1 IS INITIAL.

  V_DPQM09-/BIC/C_ATTRI   = <SOURCE_FIELDS>-/BIC/C_SEATTRI.
  V_DPQM09-/BIC/C_MBETG   = <SOURCE_FIELDS>-/BIC/C_SEMBETG.
  V_DPQM09-/BIC/C_PRUDT   = <SOURCE_FIELDS>-/BIC/C_SEPRUDT.
  V_DPQM09-/BIC/C_PRUET   = <SOURCE_FIELDS>-/BIC/C_SEPRUET.
  V_DPQM09-/BIC/C_PRUDV   = <SOURCE_FIELDS>-/BIC/C_SEPRUDV.
  V_DPQM09-/BIC/C_PRUTV   = <SOURCE_FIELDS>-/BIC/C_SEPRUTV.
  V_DPQM09-/BIC/C_MASCHIN = <SOURCE_FIELDS>-/BIC/C_SEMASC.
ELSE.

  IF <SOURCE_FIELDS>-/BIC/C_SRMITNI = 'X'
    OR <SOURCE_FIELDS>-/BIC/C_SRMITNI = '*'
    OR NOT <SOURCE_FIELDS>-/BIC/C_SRCODE1 IS INITIAL.

    V_DPQM09-/BIC/C_ATTRI   = <SOURCE_FIELDS>-/BIC/C_SRATTRI.
    V_DPQM09-/BIC/C_MBETG   = <SOURCE_FIELDS>-/BIC/C_SRMBETG.
    V_DPQM09-/BIC/C_PRUDT   = <SOURCE_FIELDS>-/BIC/C_SRPRUDT.
    V_DPQM09-/BIC/C_PRUET   = <SOURCE_FIELDS>-/BIC/C_SRPRUET.
    V_DPQM09-/BIC/C_PRUDV   = <SOURCE_FIELDS>-/BIC/C_SRPRUDV.
    V_DPQM09-/BIC/C_PRUTV   = <SOURCE_FIELDS>-/BIC/C_SRPRUTV.
    V_DPQM09-/BIC/C_MASCHIN = <SOURCE_FIELDS>-/BIC/C_SRMASC.
  ELSE.
```

```
    V_DPQM09-/BIC/C_ATTRI   = <SOURCE_FIELDS>-/BIC/C_MRATTRI.
    V_DPQM09-/BIC/C_MBETG   = <SOURCE_FIELDS>-/BIC/C_MRMBETG.
    V_DPQM09-/BIC/C_PRUDT   = <SOURCE_FIELDS>-/BIC/C_MRPRUDT.
    V_DPQM09-/BIC/C_PRUET   = <SOURCE_FIELDS>-/BIC/C_MRPRUET.
    V_DPQM09-/BIC/C_PRUDV   = <SOURCE_FIELDS>-/BIC/C_MRPRUDV.
    V_DPQM09-/BIC/C_PRUTV   = <SOURCE_FIELDS>-/BIC/C_MRPRUTV.
    V_DPQM09-/BIC/C_MASCHIN = <SOURCE_FIELDS>-/BIC/C_MRMASC.
  ENDIF.
ENDIF.

IF <SOURCE_FIELDS>-/BIC/C_SRMITNI = 'X'
  OR <SOURCE_FIELDS>-/BIC/C_SRMITNI = '*'
  OR NOT <SOURCE_FIELDS>-/BIC/C_SRCODE1 IS INITIAL.

  V_DPQM09-/BIC/C_SAT2S   = <SOURCE_FIELDS>-/BIC/C_SRSAT2S.
ELSE.

  V_DPQM09-/BIC/C_SAT2S   = <SOURCE_FIELDS>-/BIC/C_MRSAT2S.
ENDIF.

CLEAR V_MITTEL.
CONDENSE <SOURCE_FIELDS>-/BIC/C_SEORIGI NO-GAPS.

IF <SOURCE_FIELDS>-/BIC/C_SEMESNI = 'X'
  OR <SOURCE_FIELDS>-/BIC/C_SEMESNI = '*'.

  V_DPQM09-/BIC/K_MITTEL = <SOURCE_FIELDS>-/BIC/K_SEMESRT.
  V_MITTEL = <SOURCE_FIELDS>-/BIC/C_SEORIGI.
ELSE.

  IF <SOURCE_FIELDS>-/BIC/C_SRMITNI = 'X'
    OR <SOURCE_FIELDS>-/BIC/C_SRMITNI = '*'.

    V_DPQM09-/BIC/K_MITTEL = <SOURCE_FIELDS>-/BIC/K_MITTESR.
    V_MITTEL = <SOURCE_FIELDS>-/BIC/C_SRORIGI.
  ELSE.
```

```
      IF <SOURCE_FIELDS>-/BIC/C_MRMITNI = 'X'
        OR <SOURCE_FIELDS>-/BIC/C_MRMITNI = '*'.

        V_DPQM09-/BIC/K_MITTEL = <SOURCE_FIELDS>-/BIC/K_MITTEMR.
        V_MITTEL = <SOURCE_FIELDS>-/BIC/C_MRORIGI.
      ELSE.

        CLEAR: V_DPQM09-/BIC/K_MITTEL,
               V_MITTEL.
      ENDIF.
    ENDIF.
  ENDIF.

  IF V_DPQM09-/BIC/K_MITTEL = 0.
    CASE V_DPQM09-/BIC/C_MBETG.
      WHEN 'A'.
        V_DPQM09-/BIC/K_MITTEL = 1.
      WHEN 'R'.
        V_DPQM09-/BIC/K_MITTEL = 0.
      WHEN OTHERS.
        V_DPQM09-/BIC/K_MITTEL = ''.
    ENDCASE.
  ENDIF.

  IF NOT <SOURCE_FIELDS>-/BIC/C_KURZTXT IS INITIAL.
    V_DPQM09-/BIC/C_RESULT = <SOURCE_FIELDS>-/BIC/C_KURZTXT.
  ELSE.
    V_DPQM09-/BIC/C_RESULT = V_MITTEL.
  ENDIF.
```

```
IF NOT <SOURCE_FIELDS>-/BIC/C_SEMASC IS INITIAL.
  V_DPQM09-/BIC/C_POSTE = <SOURCE_FIELDS>-/BIC/C_SEMASC.
ELSE.
  IF NOT <SOURCE_FIELDS>-/BIC/C_SRMASC IS INITIAL.
    V_DPQM09-/BIC/C_POSTE = <SOURCE_FIELDS>-/BIC/C_SRMASC.
  ELSE.
    IF NOT <SOURCE_FIELDS>-/BIC/C_MRMASC IS INITIAL.
      V_DPQM09-/BIC/C_POSTE = <SOURCE_FIELDS>-/BIC/C_MRMASC.
    ELSE.
      V_DPQM09-/BIC/C_POSTE = <SOURCE_FIELDS>-/BIC/C_WORKCTR.
    ENDIF.
  ENDIF.
ENDIF.

IF <SOURCE_FIELDS>-/BIC/C_SRMITNI = 'X'
  OR <SOURCE_FIELDS>-/BIC/C_SRMITNI = '*'
    OR NOT <SOURCE_FIELDS>-/BIC/C_SRCODE1 IS INITIAL.
  V_DPQM09-/BIC/C_QERGDAT = <SOURCE_FIELDS>-/BIC/C_SRQERGD.
ELSE.
  IF NOT <SOURCE_FIELDS>-/BIC/C_MRQERGD IS INITIAL.
    V_DPQM09-/BIC/C_QERGDAT = <SOURCE_FIELDS>-/BIC/C_MRQERGD.
  ENDIF.
ENDIF.

INSERT V_DPQM09 INTO TABLE ITB_H_DPQM09.
```

Internal table itb_C_GLBFILT take values from master data global filter for stream 'QM' and rule = 'INSP_TEXT"

Table: /BIC/PC_GLBFILT

/BIC/C_STREAM	/BIC/C_RULE	/BIC/C_GLBFILT	OBJVE...	CHANGED	/BIC/C_DESC	/BIC/C_SIGN	/BIC/C_OPTION	/BIC/C_LOW	/BIC/C_HIGH	/BIC/C_ACTIVE
QM	INSP_TEXT	001	A		List of Inspection Text Key Word	I	EQ	ADD		Y
QM	INSP_TEXT	002	A		List of Inspection Text Key Word	I	EQ	EXTRA		Y
QM	INSP_TEXT	003	A		List of Inspection Text Key Word	I	EQ	ADJUSTMENTS		Y

Fields routine:

Fields:

K_MITTEL (Quantitative Result) / C_SATZS (Results Record Status (Global)) / C_PRUDV (Inspection Start Date (Global)) / C_RESULT (Result) / C_PRUDT (End Date of the Inspection (Global)) / C_PRUET (End Time of the Inspection (Global)) / C_MBETG (Inspection Result Valuation (Global)) / C_PRUTV (Starting Time of the Inspection (Global)) / C_POSTE (Work Center (real) = General information if exists) / C_ATTRI (Attribute of the Results Record (Valid, Invalid,..) (Global)) / C_MASCHIN (General Information (Global)) / C_QERGDAT Origin of Results Data (Global)

come from itb_h_dpqm09 (key fields insp_lot / c_insp_ch / plan_type / plnnr / c_plannod / c_phys_sp / c_detaile / c_probenr / logsys).

0CO_AREA comes from master data PLANT.

End routine:

Fields C_FTR / C_TOT_SL / C_TOT_OFF / C_FSR contains TRUE or FALSE with a comparaison between the records of fields C_PHYS_SP / C_LTXA1 and INSP_LOT (see end routine to more details).

TRSF: DPQM10 -> DBQM04 (Rhodia)

Except for the selective deletion in start routine not present here, the logic and rules are the same as transofrmation TRSF: DPQM09 -> DBQM04 (Rhodia)

TRSF: IH_QM_2LIS_05_QE1_RCS -> DPQM09

Key figure K_MINUTES comes from dso DPQM09 in case of same records was already loaded in the past with K_MINUTES greater than 0.

C_SHIFTDT (Shift work day) and C_SHIFT (Work shift) are determined in function of workctr (work center), c_qzeitv (Starting Time of the Inspection (QAMR)).

TRSF: DPQM07 -> DBQM03 (Solvay)

Start routine: If field BATCH is empty, he takes value from C_LSCHARG (Batch Number (QALS)).

Internal table itb_h_dpqm07 is filled with same rales as itb_h_dpqm09 in transformation TRSF: DPQM09 -> DBQM04 (Rhodia)

Fields:

K_MITTEL (Quantitative Result) / C_SATZS (Results Record Status (Global)) / C_PRUDV (Inspection Start Date (Global)) / C_RESULT (Result) / C_PRUDT (End Date of the Inspection (Global)) / C_PRUET (End Time of the Inspection (Global)) / C_MBETG (Inspection Result Valuation (Global)) / C_PRUTV (Starting Time of the Inspection (Global)) / C_POSTE (Work Center (real) = General information if exists) / C_ATTRI (Attribute of the Results Record (Valid, Invalid,..) (Global)) / C_MASCHIN (General Information (Global)) / C_QERGDAT Origin of Results Data (Global)

come from itb_h_dpqm07 (key fields insp_lot / c_insp_ch / plan_type / plnnr / c_plannod / c_phys_sp / c_detaile / c_probenr / logsys / recordmode).

TRSF: 2LIS_05_QE2 - Solvay -> IH_QM_2LIS_05_QE2

0CO_AREA comes from master data PLANT.

TRSF: DPQM08 -> DBQM03 (Solvay)

Data with C_MRMITNI (Result exists) is empty are deleted.

Internal table itb_h_dpqm08 is filled with same rales as itb_h_dpqm08 in transformation TRSF: DPQM09 -> DBQM04 (Rhodia) and same fields are loaded.

0CO_AREA comes from master data PLANT.

TRSF: IH_QM_2LIS_05_QE1_SOLVAY -> DPQM08

C_SHIFTDT (Shift work day) and C_SHIFT (Work shift) are determined in function of workctr (work center), c_qzeitv (Starting Time of the Inspection (QAMR)).

MVQM02 cubes associated with no data or not loaded since 06.2020 (doc done in december 2023).

CPQM01

Based on CVs CV_QM_INPDVL_PF1 & /CV_QM_INPDVL_WP1.
Theses CVs use dso DBQM03 & DBQM04 (transformations associtaed already documented), DPSD001 and DSO_DLV1.

RSDS DTS_BW_LIPS_LIKP PF1_020 -> ODSO DPSD001

In start routine an internal table itb_dpsd001 is updated with fiscvarnt, delnum2 (delivery number) and delitm2 (delivery item) from dpsd001 (only for data from package with same c_delnum2 in DSO).

This table is usend in end routine to fill fiscvarnt.

TRSF: 2LIS_12_VCITM -> DSO_DLV1

0ITEM_CATEG (Sales document item category) comes from:

dso_sd02 in function of vgbel (reference doc) and vgpos (reference item) and if the two firsts characters from vgbel (reference doc) not equals to 45.

If vgbel equals to 45, default value = NLC.

Else, 0ITEM_CATEG equal source field pstyv (item category)

CPQM02:

Based on aDSO: ABQM02, APQM01, ABQM04, ABQM01 and ABQM03.

ABQM02 & ABQM04 have a source DBQM03 (transformations associated already documented) with direct mapping.

ABQM01 & ABQM03 have a source DBQM04 (transformations associated already documented) with direct mapping.

No specials rules for APQM01.

CPQM03:

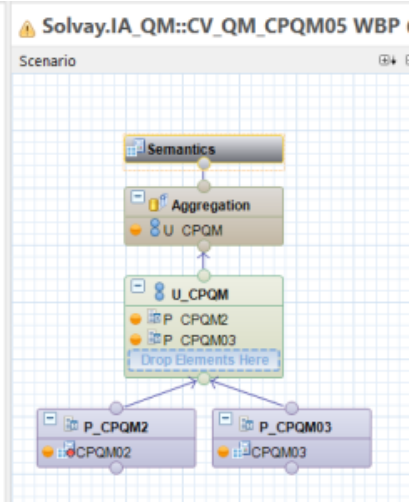
No speciales rules in transofrmations asscoiated

CPQM04:

No speciales rules in transofrmations asscoiated

CPQM05:

CV CV_QM_CPQM05 based on CPQM02 and CPQM03



Propagation layer-APQM02

This ADSO extracts data from the following two data sources from WP1 and PF1

DTS_BW_QM_ATTRIBUT(WP1_400) APQM02

DTS_BW_QM_ATTRIBUT(PF1_020) APQM02

Both the Data sources created based on Function Modules(***ZFM_BW_QM_ATTRIBUT*** *(PF1+WP1*) and implemented the same business logic in both the systems.

Function module extracts Inspections lot with ' SATZSTATUS -5 (Record status completed) and ATTRIBUT type Invalid results Only .

Source for this function module is QASR,QALS,QAMV,QAPP and PLKO tables in SAP in order to find the required field

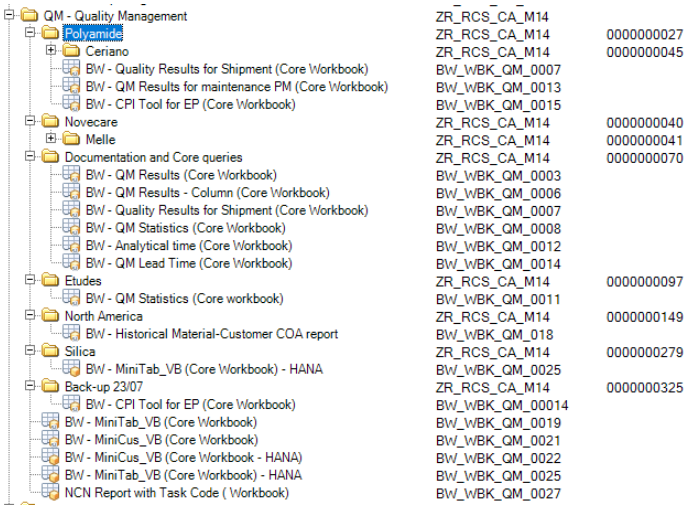
From source to PSA data load will be triggered based on created on and changed on date in two different Info packages. There is a code at Info package level to extract last four data.

Reporting

Query	Description
BW_QRY_CPQM01_001	BW - Quality Results for Deliveries (core query-HANA)
BW_QRY_CPQM02_0001	BW - QM - Results (Core Query) - HANA
BW_QRY_CPQM02_0002	BW-QM Statistics (Core Query)
BW_QRY_CPQM02_0003	BW-QM Statistics (PpK and CpK)- (Core Query)
BW_QRY_CPQM02_0005	BW - QM Results - Column (Core Query) - HANA
BW_QRY_CPQM02_0006	BW - Analytical time (Core Query) - HANA
BW_QRY_CPQM02_0007	QM - Equip - Maintenance (Corr/Prev/Adj) Core Query - HANA
BW_QRY_CPQM02_0011	BW - QM - Inspection Results by Lot (Core Query) - HANA
BW_QRY_CPQM02_0013	BW SILICA - Résultats QM (Core Query) - HANA
BW_QRY_CPQM02_9003	QM - Laboratory Billing (by Determinations) Core Query- HANA
BW_QRY_CPQM02_9004	QM - Laboratory Productivity (by Workcenter)Core Query- HANA
BW_QRY_CPQM02_9005	QM - Laboratory Productivity (by Analyst) Core Query - HANA
BW_QRY_CPQM04_0001	QM : Inspection Lot/Usage (Core Query)
BW_QRY_CPQM05_0001	BW-QM:Inspection Results & Invalid Results(Core Query)-HANA
BW_QRY_MVQM01_0001	BW - QM - Results (Core Query)

BW_QRY_MVQM01_0003	BW - QM - Results - Column (Core Query)
BW_QRY_MVQM01_0004	BW SILICA - Résultats QM
QV_BW_QRY_CPQM02_0001	QVBW - QM - Results (Core Query) - HANA
QV_BW_QRY_CPQM02_0013	QV - BW SILICA - Result QM - HANA (QV)
QV_BW_QRY_CPQM02_0014	BW - QM Statistics (PpK and CpK)
QV_BW_QRY_CPQM03_0001	QVBW - QM: Inspection Invalid Results(Core Query) - HANA

Workbooks:



Dependencies with other applications

We should have the information where the application is sending or receiving information (e.g. APD open hub)

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	Duration
PC_QM_16 TD - D - QM - RL - Inspection Results - Solvay	ABQM04	Daily (not weekend)	3:23 am (triggered by PC_QM_34) 08:20 am (triggered by PC_QM_36) 01:04 pm (triggered by PC_QM_36) 05:03 pm (triggered by PC_QM_36)	1 min
PC_QM_13 TD - D - QM - RL - Inspection Results - Rhodia	ABQM03	Daily (not weekend)	3:23 am (triggered by PC_QM_34) 08:20 am (triggered by PC_QM_36) 01:04 pm (triggered by PC_QM_36) 05:03 pm (triggered by PC_QM_36)	1 min
PC_QM_29 TD - D - QM - PL - Inspection Results (Rhodia)	APQM01 ABQM05	Daily (not weekend)	3:08 am (triggered by PC_QM_02) 3:20 am (triggered by PC_QM_02) 08:20 am (triggered by PC_QM_36) 01:04 pm (triggered by PC_QM_36) 05:03 pm (triggered by PC_QM_36)	2 mins

PC_QM_40 TD - D - QM - BL - Inspection Lot/Usage Decision (RCS)	ABQM07	Daily (not weekend)	Arround 3:00 am (triggered by PC_QM_00)	1 mins
PC_QM_39 TD - D - QM - BL - Inspection Lot/Usage Decision (RCS)	ABQM06	Daily (not weekend)	Arround 3:00 am (triggered by PC_QM_00)	1 mins

Data Quality Control

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 criticality, workarounds and resolution plan.>

Roadmap

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