

Mecano - Reload on Mecano query

- [Overview](#)
- [Talend Jobs](#)
 - [Source to ODS](#)
 - [ODS to DM](#)
- [Incremental load](#)
- [Reload](#)
 - [F04_QVMECANO_BW_QRY_MVPMNO04_0001](#)
 - [F05: QVMECANO_BW_QRY_MVPMNO04_0002](#)
 - [F06: QVMECANO_BW_QRY_MVPMCL01_0001](#)
 - [F07: QVMECANO_BW_QRY_MVPMOP04_0001](#)
 - [F08: QVMECANO_BW_QRY_MVPMOP04_0002](#)
 - [F09: QVMECANO_BW_QRY_MVPMOP04_0003](#)
 - [F11: QVMECANO_BW_QRY_MVPMOP04_0005](#)
 - [F12_QVMECANO_BW_QRY_MVPMOP04_0006](#)
 - [F13: QVMECANO_BW_QRY_MVPMOR04_0001](#)
 - [F14: QVMECANOBW_QRY_MVPMOR01_0002](#)
 - [F15: QV_BW_QRY_MVPMOP02_0010](#)
 - [F16: QVMECANO_BW_QRY_MVPMCO01_0001](#)

Overview

The variable to control the reload is `I_LOCAL_VAR_RECOVERY_MODE`

true = Reload

false = Incremental

This will control only PL_MECANO_DASH (weekly load).

There are 3 groups of loading variables

Day(yyyyMMdd,yyyyMMdd) : QVMECANOBW_QRY_MVPMOR01_0002

- `I_LOCAL_VAR_MVPMOR01_YYYYMMDD_FROM` eg. 20241001
- `I_LOCAL_VAR_MVPMOR01_YYYYMMDD_TO` ***Must NOT over current Monday

Edit variables for MVPMOR01/QVMECANOBW_QRY_MVPMOR01_0002

Name	Caption	Sel. Type	Sign	Option	Value [from]	Value [to]	Complex Edit
OL_DAYIN	Day Interval	Interval	Include	Between	YYYYMMDD_FROM	YYYYMMDD_TO	Edit

Week (yyyyww) : The rest of queries

- `I_LOCAL_VAR_HISTO_YEAR` eg. 2024
- `I_LOCAL_VAR_HISTO_WEEK_END` eg. 01

Edit variables for MVPMNO04/QVMECANO_BW_QRY_MVPMNO04_0001

Name	Caption	Sel. Type	Sign	Option	Value [from]	Value [to]	Complex Edit
V_AUT_C_PLANT_0001	Plant (Selection Option, Optional, Auth)	Complex		Between	A1	A1	Edit
OCALWEEK_0001	Calendar Year/Week (Select Option, Optional)	Complex		Equal	YYYYWW	A1	Edit

** This will apply to all the queries that use Week parameter.

Month(yyyy0MM,yyyy0MM) : QVMECANO_BW_QRY_MVPMCO01_0001

- `I_LOCAL_VAR_MVPMCO01_0001_HIST_K4YYYY0MM_PREV` eg. K42024009
- `I_LOCAL_VAR_MVPMCO01_0001_HIST_K4YYYY0MM_CURR` eg. K42024010

Edit variables for MVPMCO01/QVMECANO_BW_QRY_MVPMCO01_0001

Name	Caption	Sel. Type	Sign	Option	Value [from]	Value [to]	Complex Edit
V_C_AUTHMA_0004	Authorization Scope (Auth with input)	Complex		Between	A1	A1	Edit
* OS_FPER	Fiscal Year Selection Period	Complex		Between	K4YYYY0MM_PREV	K4YYYY0MM_CUR	Edit

There are 2 data flow of loading

1. PL_MECANO_DASH (F_MAIN_MECANO) load weekly on Monday
2. PL_MECANO_DAILY_LOAD Every Tue, Wed, Thu and Fri impact only query QVMECANO_BW_QRY_MVPMOP04_0006

** Remark: The common confusion

- Query QVMECANO_BW_QRY_MVPMOP04_0006 has the Week variable (yyyyww) but it uses the daily flow loading. Since this query needs to update every day.
- Query QVMECANO_BW_QRY_MVPMOR01_0002 has the Day variable (yyyyMMdd) but it is a weekly load.

Talend Jobs

To reload run job F_MAIN_MECANO, which cover source to DM.

Source to ODS

F100_MEC_EXTRACT_BW_QUERY_TO_GSC

List of query Args(tFixedFlowInput_1)

Schema	Number of rows	Mode	Use Single Table	Use Inline Table	Inline Table																																				
Built-In	1	Use Single Table	☐	☒	<table border="1"> <thead> <tr> <th>l_var_xtract_job</th> <th>l_var_file</th> <th>valid</th> <th>stg_filename</th> </tr> </thead> <tbody> <tr> <td>context.l_var_mecano_xtract_DBPM...</td> <td>context.l_var_mecano_xtract_DBPM...</td> <td>context.l_var_mecano_xtract_DBPM...</td> <td>"MEC_IT_0000_0000_F001_..."</td> </tr> <tr> <td>context.l_var_mecano_xtract_DBPM...</td> <td>context.l_var_mecano_xtract_DBPM...</td> <td>context.l_var_mecano_xtract_DBPM...</td> <td>"MEC_IT_0000_0000_F002_..."</td> </tr> <tr> <td>context.l_var_mecano_xtract_DBPM...</td> <td>context.l_var_mecano_xtract_DBPM...</td> <td>context.l_var_mecano_xtract_DBPM...</td> <td>"MEC_IT_0000_0000_F003_..."</td> </tr> <tr> <td>context.l_var_mecano_xtract_mvpm...</td> <td>context.l_var_mecano_xtract_mvpm...</td> <td>context.l_var_mecano_xtract_mvpm...</td> <td>"MEC_IT_0000_0000_F004_..."</td> </tr> <tr> <td>context.l_var_mecano_xtract_mvpm...</td> <td>context.l_var_mecano_xtract_mvpm...</td> <td>context.l_var_mecano_xtract_mvpm...</td> <td>"MEC_IT_0000_0000_F005_..."</td> </tr> <tr> <td>context.l_var_mecano_xtract_mvpm...</td> <td>context.l_var_mecano_xtract_mvpm...</td> <td>context.l_var_mecano_xtract_mvpm...</td> <td>"MEC_IT_0000_0000_F006_..."</td> </tr> <tr> <td>context.l_var_mecano_xtract_mvpm...</td> <td>context.l_var_mecano_xtract_mvpm...</td> <td>context.l_var_mecano_xtract_mvpm...</td> <td>"MEC_IT_0000_0000_F007_..."</td> </tr> <tr> <td>context.l_var_mecano_xtract_mvpm...</td> <td>context.l_var_mecano_xtract_mvpm...</td> <td>context.l_var_mecano_xtract_mvpm...</td> <td>"MEC_IT_0000_0000_F008_..."</td> </tr> </tbody> </table>	l_var_xtract_job	l_var_file	valid	stg_filename	context.l_var_mecano_xtract_DBPM...	context.l_var_mecano_xtract_DBPM...	context.l_var_mecano_xtract_DBPM...	"MEC_IT_0000_0000_F001_..."	context.l_var_mecano_xtract_DBPM...	context.l_var_mecano_xtract_DBPM...	context.l_var_mecano_xtract_DBPM...	"MEC_IT_0000_0000_F002_..."	context.l_var_mecano_xtract_DBPM...	context.l_var_mecano_xtract_DBPM...	context.l_var_mecano_xtract_DBPM...	"MEC_IT_0000_0000_F003_..."	context.l_var_mecano_xtract_mvpm...	context.l_var_mecano_xtract_mvpm...	context.l_var_mecano_xtract_mvpm...	"MEC_IT_0000_0000_F004_..."	context.l_var_mecano_xtract_mvpm...	context.l_var_mecano_xtract_mvpm...	context.l_var_mecano_xtract_mvpm...	"MEC_IT_0000_0000_F005_..."	context.l_var_mecano_xtract_mvpm...	context.l_var_mecano_xtract_mvpm...	context.l_var_mecano_xtract_mvpm...	"MEC_IT_0000_0000_F006_..."	context.l_var_mecano_xtract_mvpm...	context.l_var_mecano_xtract_mvpm...	context.l_var_mecano_xtract_mvpm...	"MEC_IT_0000_0000_F007_..."	context.l_var_mecano_xtract_mvpm...	context.l_var_mecano_xtract_mvpm...	context.l_var_mecano_xtract_mvpm...	"MEC_IT_0000_0000_F008_..."
l_var_xtract_job	l_var_file	valid	stg_filename																																						
context.l_var_mecano_xtract_DBPM...	context.l_var_mecano_xtract_DBPM...	context.l_var_mecano_xtract_DBPM...	"MEC_IT_0000_0000_F001_..."																																						
context.l_var_mecano_xtract_DBPM...	context.l_var_mecano_xtract_DBPM...	context.l_var_mecano_xtract_DBPM...	"MEC_IT_0000_0000_F002_..."																																						
context.l_var_mecano_xtract_DBPM...	context.l_var_mecano_xtract_DBPM...	context.l_var_mecano_xtract_DBPM...	"MEC_IT_0000_0000_F003_..."																																						
context.l_var_mecano_xtract_mvpm...	context.l_var_mecano_xtract_mvpm...	context.l_var_mecano_xtract_mvpm...	"MEC_IT_0000_0000_F004_..."																																						
context.l_var_mecano_xtract_mvpm...	context.l_var_mecano_xtract_mvpm...	context.l_var_mecano_xtract_mvpm...	"MEC_IT_0000_0000_F005_..."																																						
context.l_var_mecano_xtract_mvpm...	context.l_var_mecano_xtract_mvpm...	context.l_var_mecano_xtract_mvpm...	"MEC_IT_0000_0000_F006_..."																																						
context.l_var_mecano_xtract_mvpm...	context.l_var_mecano_xtract_mvpm...	context.l_var_mecano_xtract_mvpm...	"MEC_IT_0000_0000_F007_..."																																						
context.l_var_mecano_xtract_mvpm...	context.l_var_mecano_xtract_mvpm...	context.l_var_mecano_xtract_mvpm...	"MEC_IT_0000_0000_F008_..."																																						

J201_STG_TO_ODS

dFlowInput_1)

File_name	Table_name	ODS_table_name	Valid	Separator	Escape_separator
"MEC_IT_0000_0000_F0...	"STG_BWH_0000_0000_...	"ODS_BWH_0000_F016_...	context.l_var_mecano...	"%"	"\"
"MEC_IT_0000_0000_F0...	"STG_BWH_0000_0000_...	"ODS_BWH_0000_F015_...	context.l_var_mecano...	"%"	"\"
"MEC_IT_0000_0000_F0...	"STG_BWH_0000_0000_...	"ODS_BWH_0000_F014_...	context.l_var_mecano...	"%"	"\"
"MEC_IT_0000_0000_F0...	"STG_BWH_0000_0000_...	"ODS_BWH_0000_F013_...	context.l_var_mecano...	"%"	"\"
"MEC_IT_0000_0000_F0...	"STG_BWH_0000_0000_...	"ODS_BWH_0000_F012_...	context.l_var_mecano...	"%"	"\"
"MEC_IT_0000_0000_F0...	"STG_BWH_0000_0000_...	"ODS_BWH_0000_F011_...	context.l_var_mecano...	"%"	"\"
"MEC_IT_0000_0000_F0...	"STG_BWH_0000_0000_...	"ODS_BWH_0000_F009_...	context.l_var_mecano...	"%"	"\"
"MEC_IT_0000_0000_F0...	"STG_BWH_0000_0000_...	"ODS_BWH_0000_F008_...	context.l_var_mecano...	"%"	"\"
"MEC_IT_0000_0000_F0...	"STG_BWH_0000_0000_...	"ODS_BWH_0000_F007_...	context.l_var_mecano...	"%"	"\"

These 2 jobs will control by following parameters in order to control the loading

Name of Xtract job parameter

It is used to control the name of Xtract name to run on the job F100_MEC_EXTRACT_BW_QUERY_TO_GSC

- l_VAR_mecano_xtract_DBPMMD01_0001_job
- l_VAR_mecano_xtract_DBPMMD02_0001_job
- l_VAR_mecano_xtract_DBPMMD05_0001_job
- l_VAR_mecano_xtract_MVPMCL01_0001_job

- I_VAR_mecano_xtract_mvpmco01_job
- I_VAR_mecano_xtract_MVPMNO04_0001_job
- I_VAR_mecano_xtract_MVPMNO04_0002_job
- I_VAR_mecano_xtract_MVPMOP02_0010_job
- I_VAR_mecano_xtract_MVPMOP04_0001_job
- I_VAR_mecano_xtract_MVPMOP04_0002_job
- I_VAR_mecano_xtract_MVPMOP04_0003_job
- I_VAR_mecano_xtract_MVPMOP04_0005_job
- I_VAR_mecano_xtract_MVPMOP04_0006_job
- I_VAR_mecano_xtract_MVPMOR01_0002_job
- I_VAR_mecano_xtract_MVPMOR04_0001_job

Valid parameter

It is used to control which query will be used to load by the job F100_MEC_EXTRACT_BW_QUERY_TO_GSC

- I_VAR_mecano_xtract_DBPMMD01_0001_valid
- I_VAR_mecano_xtract_DBPMMD02_0001_valid
- I_VAR_mecano_xtract_DBPMMD05_0001_valid
- I_VAR_mecano_xtract_MVPMCL01_0001_valid
- I_VAR_mecano_xtract_mvpmco01_valid
- I_VAR_mecano_xtract_MVPMNO04_0001_valid
- I_VAR_mecano_xtract_MVPMNO04_0002_valid
- I_VAR_mecano_xtract_MVPMOP02_0010_valid
- I_VAR_mecano_xtract_MVPMOP04_0001_valid
- I_VAR_mecano_xtract_MVPMOP04_0002_valid
- I_VAR_mecano_xtract_MVPMOP04_0003_valid
- I_VAR_mecano_xtract_MVPMOP04_0005_valid
- I_VAR_mecano_xtract_MVPMOP04_0006_valid
- I_VAR_mecano_xtract_MVPMOR01_0002_valid
- I_VAR_mecano_xtract_MVPMOR04_0001_valid

ODS to DM

F310_MEC_LOAD_FACT_MAINTENANCE_STRATEGIC (always delete the same ref_date on the last load from ODS in FACT before loading using WDL.
 FACT_maintenance_dash_strategic_cost_budget_tmp)
 F320_MEC_LOAD_FACT_MAINTENANCE_EFFECTIVENESS
 F330_MEC_LOAD_FACT_MAINTENANCE_EFFICIENCY

Incremental load

Variable = I_LOCAL_VAR_RECOVERY_MODE = false

Talend job = F_MAIN_MECANO

Talend sub job = F100_MEC_EXTRACT_BW_QUERY_TO_GSC

This job will determine

Day (yyyyMMdd,yyyyMMdd): get last Monday to current Monday by using custom routine

String to_week=R001_MECANO.getCurrentMonday(TalendDate.getCurrentDate());

String from_week=R001_MECANO.getpreviousMonday(TalendDate.getCurrentDate());

Week(yyyyww): Get the week of loading date - 1 day

int currentWeek = TalendDate.getPartOfDate("WEEK_OF_YEAR", TalendDate.getCurrentDate()) - 1;

String weekOfYear = (currentWeek > 0) ? String.format("%02d", currentWeek) : "52";

String param=TalendDate.getDate("YYYY")+ weekOfYear;

Month(yyyy0MM, yyyy0MM): get last month to current month based on loading date

String prev="K4"+TalendDate.formatDate("yyyy0MM", TalendDate.addDate(TalendDate.getCurrentDate(), -1,"MM"));

String cur="K4"+TalendDate.formatDate("yyyy0MM", TalendDate.getCurrentDate());

Reload

DAILY FLOW

It is not allowed to reload. It must use Weekly flow instead.

The daily flow will load only TALEND_QVMECANO_BW_QRY_MVPMOP04_0006

WEEKLY FLOW

Variable = I_LOCAL_VAR_RECOVERY_MODE = true

Talend job = F_MAIN_MECANO

Talend sub job = F100_MEC_EXTRACT_BW_QUERY_TO_GSC

This job will determine

Day (yyyyMMdd,yyyyMMdd):

String from_week=context.I_LOCAL_VAR_MVPMOR01__YYYYMMDD_FROM

String to_week=context.I_LOCAL_VAR_MVPMOR01__YYYYMMDD_TO

Remark:

- Must NOT over current Mon otherwise, it will be duplicate
- 1 Month will have size around 20 MB and 50k rows

Week(yyyyww):

yyyyww = I_LOCAL_VAR_HISTO_YEAR + I_LOCAL_VAR_HISTO_WEEK_END

Month(yyyy0MM, yyyy0MM):

String prev = context.I_LOCAL_VAR_MVPMCO01_0001_HIST_K4YYYY0MM_PREV;

String cur = context.I_LOCAL_VAR_MVPMCO01_0001_HIST_K4YYYY0MM_CURR;

F04_QVMECANO_BW_QRY_MVPMNO04_0001

Selection parameter = WEEK(yyyyww)

Fact table = FACT_maintenance_efficiency

This query will impact on kpi_no = 2.01/2.02/2.05_hist:

- nb_of_notification_created
- nb_of_notif_in_progress
- nb_of_notif_in_progress_without_work_order
- nb_of_notif_in_progress_with_delay
- nb_of_notif_in_progress_delay_1_to_30_days
- nb_of_notif_in_progress_delay_31_to_90_days
- nb_of_notif_in_progress_delay_more_than_91_days
- nb_of_notif_in_progress_no_required_end_date

Validation

select reference_week,

sum(nb_of_notification_created) nb_of_notification_created,

sum(nb_of_notif_in_progress) nb_of_notif_in_progress,

sum(nb_of_notif_in_progress_without_work_order) nb_of_notif_in_progress_without_work_order,

sum(nb_of_notif_in_progress_with_delay) nb_of_notif_in_progress_with_delay,

sum(nb_of_notif_in_progress_delay_1_to_30_days) nb_of_notif_in_progress_delay_1_to_30_days,

sum(nb_of_notif_in_progress_delay_31_to_90_days) nb_of_notif_in_progress_delay_31_to_90_days,

sum(nb_of_notif_in_progress_delay_more_than_91_days) nb_of_notif_in_progress_delay_more_than_91_days,

sum(nb_of_notif_in_progress_no_required_end_date) nb_of_notif_in_progress_no_required_end_date

from `prj-data-maintenance-dash-dev.DPL.DPL.V\_REP\_FACT\_maintenance\_dash`

where kpi_no = '2.01/2.02/2.05_hist'

group by reference_week
order by reference_week desc

Query results

SAVE RESULTS

JOB INFORMATION		RESULTS	CHART	JSON	EXECUTION DETAILS		EXECUTION GRAPH			
Row	reference_week	nb_of_notification_created	nb_of_notif_in_progress	nb_of_notif_in_progress_without_work_order	nb_of_notif_in_progress_with_delay	nb_of_notif_in_progress_delay_1_to_30_days	nb_of_notif_in_progress_delay_31_to_90_days	nb_of_notif_in_progress_delay_more_than_91_days	nb_of_notif_in_progress_no_required_end_date	nb_of_notif_in_progress_no_required_end_date
1	48.2024	8319	251027	42643	171801	12909	15233	143659	53204	53204
2	47.2024	8912	250383	42077	171043	13129	15166	142748	53140	53140
3	46.2024	8857	249616	42291	170260	12616	15289	142355	52697	52697
4	45.2024	9197	249606	42233	170292	12730	15183	142379	52798	52798
5	44.2024	8521	248886	41747	169306	12408	15159	141739	53166	53166
6	43.2024	8923	248592	41712	168192	12059	15216	140917	53390	53390
7	42.2024	8964	247009	41520	167615	12344	15027	140244	52670	52670
8	41.2024	8382	244911	41476	166342	12117	14818	139407	52351	52351

BW Result

Query Technical Name	QVMECANO_BW_QRY_MVPMNO04_0001								
Snapshot (Week)	41.2024 - 48.2024								
Planning Plant	WP1_400/GRU								
Snapshot (Week)	Nb of Notification Created	Nb of Notif. In Progress	Nb of Notif. In Progress without Work Order	Nb of Notif. In Progress with Delay	Nb of Notif. In Progress (Delay 1 to 30 days)	Nb of Notif. In Progress (Delay 31 to 90 days)	Nb of Notif. In Progress (Delay more than 91 days)		
41.2024	369	10,481	337	7,691	659	1,152	5,880		
42.2024	324	10,594	326	7,837	729	1,117	5,991		
43.2024	325	10,799	340	7,996	763	1,140	6,093		
44.2024	351	10,776	348	8,046	739	1,159	6,146		
45.2024	464	11,048	430	8,217	806	1,153	6,258		
46.2024	331	11,204	361	8,375	832	1,168	6,375		
47.2024	316	11,365	353	8,582	893	1,179	6,510		
48.2024	161	11,359	394	8,649	841	1,181	6,627		

Reload

For example to reload 41.2024

- Stop the loading in TMC.
- it is required to delete the data from DM.FACT_maintenance_efficiency in prj-data-dm-industrial-dev
delete from `DM.FACT\_maintenance\_efficiency`

where reference_week = '41.2024' and kpi_no = '2.01/2.02/2.05_hist'

- Change the variable

I_LOCAL_VAR_RECOVERY_MODE = true

I_LOCAL_VAR_HISTO_YEAR = 2024

I_LOCAL_VAR_HISTO_WEEK_END = 41

- Control the query to be run on the "valid" I_VAR_mecano_xtract_MVPMNO04_0001_valid = 1 and select the other to 0 in case we don't want to reload all the queries

F05: QVMECANO_BW_QRY_MVPMNO04_0002

Selection parameter = WEEK(yyyyww)

Fact table = FACT_maintenance_efficiency

This query will impact on kpi_no = 2.01/2.02/2.05_det:

- nb_of_notification_created
- nb_of_notif_in_progress
- nb_of_notif_in_progress_without_work_order
- nb_of_notif_in_progress_with_delay
- nb_of_notif_in_progress_delay_1_to_30_days
- nb_of_notif_in_progress_delay_31_to_90_days
- nb_of_notif_in_progress_delay_more_than_91_days
- nb_of_notif_in_progress_no_required_end_date

Validation

select reference_week,

```

sum(nb_of_notification_created) nb_of_notification_created,
sum(nb_of_notif_in_progress) nb_of_notif_in_progress,
sum(nb_of_notif_in_progress_without_work_order) nb_of_notif_in_progress_without_work_order,
sum(nb_of_notif_in_progress_with_delay) nb_of_notif_in_progress_with_delay,
sum(nb_of_notif_in_progress_delay_1_to_30_days) nb_of_notif_in_progress_delay_1_to_30_days,
sum(nb_of_notif_in_progress_delay_31_to_90_days) nb_of_notif_in_progress_delay_31_to_90_days,
sum(nb_of_notif_in_progress_delay_more_than_91_days) nb_of_notif_in_progress_delay_more_than_91_days,
sum(nb_of_notif_in_progress_no_required_end_date) nb_of_notif_in_progress_no_required_end_date
from `prj-data-maintenance-dash-dev.DPL.V_REP_FACT_maintenance_dash`
where kpi_no = '2.01/2.02/2.05_det'
and reference_week between '41.2024' and '48.2024'
and planning_plant_key like '%GRU%'
group by reference_week
order by reference_week

```

BW Result

Query Technical Name	QVMECANO_BW_QRY_MVPMNO04_0002								
Snapshot (Week)	41.2024 - 48.2024								
PM planning plant	WP1_400/GRU								
Snapshot (Week)	Nb of Notification Created	Nb of Notif. In Progress	Nb of Notif. In Progress without Work Order	Nb of Notif. In Progress with Delay	Nb of Notif. In Progress (Delay 1 to 30 days)	Nb of Notif. In Progress (Delay 31 to 90 days)	Nb of Notif. In Progress (Delay more than 91 days)	Nb	
41.2024	369	10,481	337	7,691	659	1,152	5,880		
42.2024	324	10,594	326	7,837	729	1,117	5,991		
43.2024	325	10,799	340	7,996	763	1,140	6,093		
44.2024	351	10,776	348	8,046	739	1,159	6,140		
45.2024	464	11,040	430	8,217	806	1,153	6,250		
46.2024	331	11,204	361	8,375	832	1,168	6,375		
47.2024	316	11,365	352	8,582	893	1,179	6,510		
48.2024	161	11,359	394	8,649	841	1,181	6,627		

Reload

Normally this query will update the kpi_no = '2.01/2.02/2.05_det'

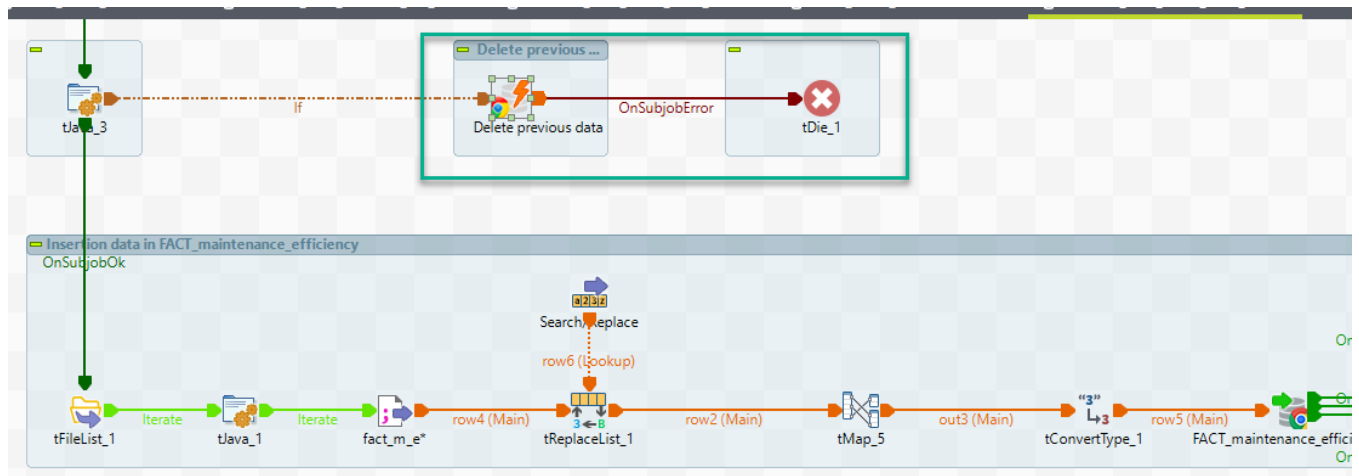
This data will be available on the DM.FACT_maintenance_efficiency only the latest week.

There is a section to delete the whole data from this query when I_LOCAL_VAR_RECOVERY_MODE == false (incremental load every Monday)

Therefore,

if we reload the data for the current week, just run the PL_MECANO_DASH with I_LOCAL_VAR_RECOVERY_MODE == false

if we reload the data for the previous weeks on DM.FACT_maintenance_efficiency, it is not required for this query.



However, if want to reload source to ODS_BWH_0000_F005_F_W_mec_mvpmno04_0002,

I_LOCAL_VAR_RECOVERY_MODE = true

I_LOCAL_VAR_HISTO_YEAR = 2024

I_LOCAL_VAR_HISTO_WEEK_END = 41

I_VAR_mecano_xtract_MVPMNO04_0002_valid = 1

To reload week 41.2024. It is required only job F100_MEC_EXTRACT_BW_QUERY_TO_GSC and J201_STG_TO_ODS. Then reload RELOAD [DM.FACT_notification_planning](#) by job F010_FACT_notification_planning_TO_DM

I_VAR_BWH_notification_planning_dm_reload_condition = and SnapShot__Week like '%2024' and substr(SnapShot__Week,1,2) >= '01' and substr(SnapShot__Week,1,2) <= '01' QUALIFY ROW_NUMBER() OVER (PARTITION BY Source_System_Key, Notification_key, Required_End_Date_Key, User_Status__Notification__List_Key, SnapShot__Week ORDER BY meta_business_date DESC) = 1

F06: QVMECANO_BW_QRY_MVPMCL01_0001

Selection parameter = WEEK(yyyyww)

Fact table = [FACT_maintenance_effectiveness](#)

This query will impact on kpi_no = 3.02

- planned_activity_nb_of_calls
- planned_activity_executed_nb_of_calls
- ongoing_activity_nb_of_calls
- activity_delayed_nb_of_calls
- activity_delayed_1_30_days_nb_of_calls
- activity_delayed_31_90_days_nb_of_calls
- activity_delayed_more_than_90_days_Nb_of_Calls
- planned_activity_nb_of_hours
- Planned_activity_executed_nb_of_hours

Validation

DPL

```
select reference_week,

sum(planned_activity_nb_of_calls) planned_activity_nb_of_calls,

sum(planned_activity_executed_nb_of_calls) planned_activity_executed_nb_of_calls,

sum(ongoing_activity_nb_of_calls) ongoing_activity_nb_of_calls,

sum(activity_delayed_nb_of_calls) activity_delayed_nb_of_calls,

sum(activity_delayed_1_30_days_nb_of_calls) activity_delayed_1_30_days_nb_of_calls,

sum(activity_delayed_31_90_days_nb_of_calls) activity_delayed_31_90_days_nb_of_calls,

sum(activity_delayed_more_than_90_days_Nb_of_Calls) activity_delayed_more_than_90_days_Nb_of_Calls,

sum(planned_activity_nb_of_hours) planned_activity_nb_of_hours,

sum(planned_activity_executed_nb_of_hours) planned_activity_executed_nb_of_hours

from `prj-data-maintenance-dash-dev.DPL.V_REP_FACT_maintenace_dash`

where kpi_no = '3.02'

and reference_week between '41.2024' and '48.2024'

and planning_plant_key like '%GRU%'

group by reference_week

order by reference_week
```

ODS

```
SELECT meta_run_id,SUM(SAFE_CAST(Planned_Activity__Nb_of_Calls as float64)) Planned_Activity__Nb_of_Calls FROM `prj-data-dm-industrial-dev.ODS.ODS_BWH_0000_F006_F_W_mec_mvpmcl01_0001`
```

where Snapshot_Week_Key = '41.2024'

and Planning_Plant_Key like '%GRU%'

group by meta_run_id

meta_run_id ▾	Planned_Activity_Nb
eba8adb1-4d31-4682-9a9f-46d586b38cbb	288.0

BW Result

	Planned Activity (Nb of Calls)	Planned Activity Executed (Nb of Calls)	Ongoing Activity (Nb of Calls)	Activity Delayed (Nb of Calls)	Activity Delayed (1-30 days) (Nb of Calls)	Activity Delayed (30-90 days) (Nb of Calls)
Snapshot Week						
41.2024	284	57	2,207	2,207		295
42.2024	328	88	2,306	2,306		277
43.2024	217	118	2,404	2,404		357
44.2024	265	136	2,384	2,384		351
45.2024	252	79	2,446	2,446		316
46.2024	272	148	2,535	2,535		297
47.2024	350	119	2,580	2,580		281
48.2024	295	106	2,760	2,760		375
5						
6						
7						
8	reference_week	planned_activity_nb_of_calls	planned_activity_executed_nb_of_calls	activity_delayed_nb_of_calls	activity_delayed_1_30_days_nb_of_calls	activity_delayed_30_90_days_nb_of_calls
9	41.2024	288	57	2,207.00	2207	295
0	42.2024	328	88	2,306.00	2306	277
1	43.2024	217	118	2,404.00	2404	357
2	44.2024	265	136	2,384.00	2384	351
3	45.2024	252	79	2,446.00	2446	316
4	46.2024	272	148	2,535.00	2535	297
5	47.2024	350	119	2,580.00	2580	281
6	48.2024	295	106	2,760.00	2760	375
7						
8		4	0	0	0	0
9		0	0	0	0	0
0		0	0	0	0	0
1		0	0	0	0	0
2		0	0	0	0	0
3		0	0	0	0	0
4		0	0	0	0	0
5		0	0	0	0	0

Reload

For example to reload 41.2024

1. Stop the loading in TMC.
2. it is required to delete the data from DM.FACT_maintenance_effectiveness in prj-data-dm-industrial-dev
delete from `DM.FACT\_maintenance\_effectiveness`
where reference_week = '41.2024' and kpi_no = '3.02'

3. Change the variable

I_LOCAL_VAR_RECOVERY_MODE = true

I_LOCAL_VAR_HISTO_YEAR = 2024

I_LOCAL_VAR_HISTO_WEEK_END = 41

4. Control the query to be run on the "valid" I_VAR_mecano_xtract_MVPMCL01_0001_valid = 1 and select the other to 0 in case we don't want to reload all the queries.

F07: QVMECANO_BW_QRY_MVPMOP04_0001

Selection parameter = WEEK(yyyymmww)

Fact table = FACT_maintenance_efficiency

This query will impact on kpi_no = 2.04

- nb_operation_planned
- nb_operation_planned_good_day
- nb_operation_reactives

Validation

```
select reference_week,
sum(nb_operation_planned) nb_operation_planned,
sum(nb_operation_planned_good_day) nb_operation_planned_good_day,
sum(nb_operation_reactives) nb_operation_reactives,
from 'prj-data-maintenance-dash-dev.DPL.V_REP_FACT_maintenance_dash'
where kpi_no = '2.06/2.07/2.09'
and reference_week between '26.2024' and '52.2024'
and planning_plant_key like '%GRU%'
group by reference_week
order by reference_week
```

Reload

For example to reload 41.2024

1. Stop the loading in TMC.

2. it is required to delete the data from DM.FACT_maintenance_efficiency in prj-data-dm-industrial-dev

delete from `DM.FACT\_maintenance\_efficiency`

where reference_week = '41.2024' and kpi_no = '2.06/2.07/2.09'

3. Change the variable

I_LOCAL_VAR_RECOVERY_MODE = true

I_LOCAL_VAR_HISTO_YEAR = 2024

I_LOCAL_VAR_HISTO_WEEK_END = 41

4. Control the query to be run on the "valid" I_VAR_mecano_xtract_MVPMOP04_0001_valid = 1 and select the other to 0 in case we don't want to reload all the queries.

F08: QVMECANO_BW_QRY_MVPMOP04_0002

Selection parameter = WEEK(yyyyww)

Fact table = FACT_maintenance_efficiency

This query will impact on kpi_no = 2.06/2.07/2.09

- nb_operation_scheduled_kpi6
- nb_operation_well_confirmed_kpi6
- nb_operations_in_delay
- remaining_work_backlog_h
- remaining_work_backlog_man_days

Validate

select reference_week,

sum(nb_operation_scheduled_kpi6) nb_operation_scheduled_kpi6,

sum(nb_operation_well_confirmed_kpi6) nb_operation_well_confirmed_kpi6,

sum(nb_operations_in_delay) nb_operations_in_delay,

sum(remaining_work_backlog_h) remaining_work_backlog_h,

sum(remaining_work_backlog_man_days) remaining_work_backlog_man_days,

from `prj-data-maintenance-dash-dev.DPL.V\_REP\_FACT\_maintenance\_dash`

where kpi_no = '2.04'

and reference_week between '41.2024' and '48.2024'

and planning_plant_key like '%GRU%'

group by reference_week

order by reference_week

Reload

For example to reload 41.2024

1. Stop the loading in TMC.

2. it is required to delete the data from DM.FACT_maintenance_efficiency in prj-data-dm-industrial-dev

delete from `DM.FACT\_maintenance\_efficiency`

where reference_week = '41.2024' and kpi_no = '2.04'

3. Change the variable

I_LOCAL_VAR_RECOVERY_MODE = true

I_LOCAL_VAR_HISTO_YEAR = 2024

I_LOCAL_VAR_HISTO_WEEK_END = 41

4. Control the query to be run on the "valid" I_VAR_mecano_xtract_MVPMOP04_0002_valid = 1 and select the other to 0 in case we don't want to reload all the queries.

F09: QVMECANO_BW_QRY_MVPMOP04_0003

Selection parameter = WEEK(yyyyww)

Fact table = FACT_maintenance_efficiency

This query will impact on kpi_no = 2.06

- nb_operation_unscheduled
- nb_operation_planned
- nb_operation_planned_good_day
- nb_operation_planned_good_week
- nb_operation_reactives
- hours_of_operations_planned
- hours_of_operations_planned_good_day
- hours_of_operations_planned_good_week
- hours_of_operations_reactives

Validate

```
select reference_week,
sum(nb_operation_unscheduled) nb_operation_unscheduled,
sum(nb_operation_planned) nb_operation_planned,
sum(nb_operation_planned_good_day) nb_operation_planned_good_day,
sum(nb_operation_planned_good_week) nb_operation_planned_good_week,
sum(nb_operation_reactives) nb_operation_reactives,
sum(hours_of_operations_planned) hours_of_operations_planned,
sum(hours_of_operations_planned_good_day) hours_of_operations_planned_good_day,
sum(hours_of_operations_planned_good_week) hours_of_operations_planned_good_week,
sum(hours_of_operations_reactives) hours_of_operations_reactives,
from `prj-data-maintenance-dash-dev.DPL.V_REP_FACT_maintenance_dash`
where kpi_no = '2.06'
and reference_week between '41.2024' and '48.2024'
and planning_plant_key like '%GRU%'
group by reference_week
order by reference_week
```

Reload

For example to reload 41.2024

1. Stop the loading in TMC.
2. it is required to delete the data from DM.FACT_maintenance_efficiency in prj-data-dm-industrial-dev

```
delete from `DM.FACT_maintenance_efficiency`
```

where reference_week = '41.2024' and kpi_no = '2.06'

3. Change the variable

I_LOCAL_VAR_RECOVERY_MODE = true

I_LOCAL_VAR_HISTO_YEAR = 2024

I_LOCAL_VAR_HISTO_WEEK_END = 41

4. Control the query to be run on the "valid" I_VAR_mecano_xtract_MVPMOP04_0003_valid = 1 and select the other to 0 in case we don't want to reload all the queries.

F11: QVMECANO_BW_QRY_MVPMOP04_0005

Selection parameter = WEEK(yyyymmww)

Fact table = FACT_maintenance_efficiency

This query will impact on kpi_no = 2.08

- remaining_work_hours
- available_capacity
- available_capacity_unit

Validate

DPL

with temp as(

select reference_week, work_center,

sum(remaining_work_hours) remaining_work_hours,

max(available_capacity) available_capacity,

from `prj-data-maintenance-dash-dev.DPL.V\_REP\_FACT\_maintenance\_dash`

where kpi_no = '2.08'

and reference_week between '26.2024' and '50.2024'

and planning_plant_key like '%GRU%'

group by reference_week,work_center)

select reference_week,

sum(remaining_work_hours) remaining_work_hours,

sum(available_capacity) available_capacity,

from temp

group by reference_week

order by reference_week

ODS

select Snapshot__Week__Key, meta_run_id, sum(safe_cast(Remaining_Work__h as float64)), sum(safe_cast(Available_capacity as float64))

from `prj-data-dm-industrial-dev.ODS.ODS\_BWH\_0000\_F011\_F\_W\_mec\_mvpmop04\_0005`

where Planning_Plant_Key like '%GRU%'

and Snapshot__Week__Key like '%2024%'

group by Snapshot__Week__Key, meta_run_id

order by Snapshot__Week__Key

Query Technical Name	QVMECANO_BW_QRY_MVPMOP04_0005							
SnapShot (Week)	26.2024 - 50.2024							
Planning Plant	WP1_400/GRU							
Snapshot Frequency	WEEKLY							
	Remaining Work (h)	Available capacity				CHECK		
SnapShot (Week)	H	H	reference_week	remaining_work_h	available_capac	remaining_work_hours	available_capacity	
26.2024	2,001.500	2,790.000	26.2024	2001.5	2790	0.000	0.000	
27.2024	1,361.000	2,600.000	27.2024	1361	2600	0.000	0.000	
28.2024	1,605.000	2,310.000	28.2024	1605	2310	0.000	0.000	
29.2024	1,667.000	2,550.000	29.2024	1667	2550	0.000	0.000	
30.2024	1,788.000	2,160.000	30.2024	1788	2160	0.000	0.000	
31.2024	1,713.000	2,290.000	31.2024	1713	2290	0.000	0.000	
32.2024	1,924.000	2,600.000	32.2024	1924	2600	0.000	0.000	
33.2024	2,443.000	2,730.000	33.2024	2443	2730	0.000	0.000	
34.2024	1,739.000	2,310.000	34.2024	1739	2310	0.000	0.000	
35.2024	1,536.500	2,410.000	35.2024	1536.5	2410	0.000	0.000	
36.2024	1,388.000	2,690.000	36.2024	1388	2690	0.000	0.000	
37.2024	1,648.000	2,600.000	37.2024	1648	2600	0.000	0.000	
38.2024	1,560.000	2,600.000	38.2024	1560	2600	0.000	0.000	
39.2024	1,493.000	2,600.000	39.2024	1493	2600	0.000	0.000	
40.2024	1,797.000	2,360.000	40.2024	1797	2360	0.000	0.000	
41.2024	1,665.000	2,680.000	41.2024	1665	2680	0.000	0.000	
42.2024	1,586.500	2,680.000	42.2024	1586.5	2680	0.000	0.000	
43.2024	2,026.500	2,600.000	43.2024	2026.5	2600	0.000	0.000	
44.2024	1,843.000	2,310.000	44.2024	1843	2310	0.000	0.000	
45.2024	1,905.000	2,790.000	45.2024	1905	2790	0.000	0.000	
46.2024	1,832.000	2,680.000	46.2024	1832	2680	0.000	0.000	
47.2024	2,318.000	3,120.000	47.2024	2318	3120	0.000	0.000	
48.2024	799.500	2,650.000	48.2024	799.5	2650	0.000	0.000	
49.2024	1,749.500	2,720.000	49.2024	1749.5	2720	0.000	0.000	
50.2024	2,051.000	3,180.000	50.2024	2051	3180	0.000	0.000	

Reload

For example to reload 41.2024

1. Stop the loading in TMC.
2. it is required to delete the data from DM.FACT_maintenance_efficiency in prj-data-dm-industrial-dev

delete from `DM.FACT\_maintenance\_efficiency`

where reference_week = '41.2024' and kpi_no = '2.08'

3. Change the variable

I_LOCAL_VAR_RECOVERY_MODE = true

I_LOCAL_VAR_HISTO_YEAR = 2024

I_LOCAL_VAR_HISTO_WEEK_END = 41

4. Control the query to be run on the "valid" I_VAR_mecano_xtract_MVPMOP04_0005_valid = 1 and select the other to 0 in case we don't want to reload all the queries.

F12_QVMECANO_BW_QRY_MVPMOP04_0006

Selection parameter = WEEK(yyyyww) / **Need to delete max reference date**

Fact table = FACT_maintenance_efficiency

This query will impact on kpi_no = 2.13

- nb_operation_planned
- nb_operation_planned_good_day
- nb_operation_reactives

Reload 1 week file size around 12 MB

Loading time around 6 min

Validation

select reference_week,

sum(nb_operation_planned) nb_operation_planned,

sum(nb_operation_planned_good_day) nb_operation_planned_good_day,

sum(nb_operation_reactives) nb_operation_reactives,

from `prj-data-maintenance-dash-dev.DPL.V\_REP\_FACT\_maintenance\_dash`

where kpi_no = '2.13'

and reference_week between '41.2024' and '48.2024'

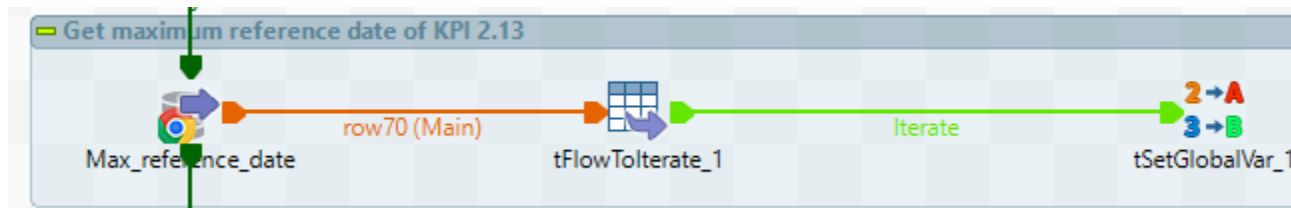
and planning_plant_key like '%GRU%'

group by reference_week

order by reference_week

The loading will get the data from ODS_BWH_0000_F012_F_W_mec_mvmpop04_0006 only the reference_date > max(reference_date) from DM.FACT_maintenance_efficiency. Therefore, it will load only the new data by using component below

SELECT max(reference_date) AS max_reference_date FROM DM.FACT_maintenance_efficiency WHERE kpi_no='2.13'



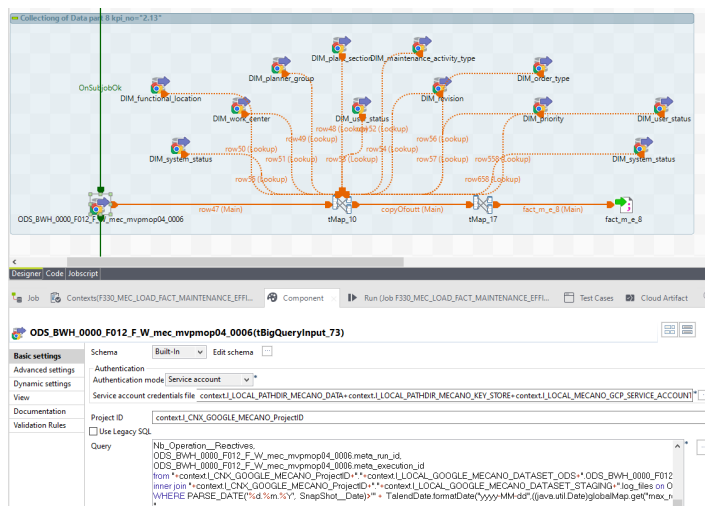
Select xxxx from ODS.ODS_BWH_0000_F012_F_W_mec_mvmpop04_0006

inner join STG.log_files on ODS_BWH_0000_F012_F_W_mec_mvmpop04_0006.meta_run_id=log_files.meta_run_id

and log_files.meta_file_name='MEC_IT_0000_0000_F012_'+context.Business_date+'_0000_F_W_MEC_MVPMOP04_0006.csv'

and log_files.meta_status in ('OK','NOK')

WHERE PARSE_DATE('%d.%m.%Y', SnapShot__Date)>' + TalendDate.formatDate("yyyy-MM-dd",((java.util.Date)globalMap.get("max_ref_date")))



4. Control the query to be run on the “valid” I_VAR_mecano_xtract_MVPMOP04_0006_valid = 1 and select the other to 0 in case we don't want to reload all the queries.

Before

Untitled query

RUN

SAVE

DOWNLOAD

SHARE

SCHEDULE

OPEN IN

MORE

```
1 select reference_date,meta_run_id,sum(nb_operation_reactives) from `prj-data-dm-industrial-dev.DM_FACT_maintenance_efficiency`
2 where reference_date >= '2024-11-01'
3 and kpi_no = '2.13'
4 --QUALITY ROW_NUMBER() OVER (PARTITION BY SnapShot__Date__Key ORDER BY meta_ods_insert_date DESC) = 1
5 group by reference_date,meta_run_id
6 order by reference_date
7 limit 100
```

Query results

JOB INFORMATION

RESULTS

CHART

JSON

EXECUTION DETAILS

EXECUTION GRAPH

Row	reference_date	meta_run_id	f0_
1	2024-11-01	51050799-6bc8-418c-a24b-fa5617c92408	541
2	2024-11-04	f2c274e8-4b15-4ba9-8eca-56573316706c	1114
3	2024-11-05	a7eaf9a2-c21f-492a-b406-48a301f6660e	929
4	2024-11-06	a7eaf9a2-c21f-492a-b406-48a301f6660e	993
5	2024-11-07	8effcc05-7ec7-4696-9c0a-7812be88c420	1025
6	2024-11-11	daeef80f-6ba0-40e8-a0d4-77b1d4d61dc4	909
7	2024-11-12	1c365062-fa21-4e04-93b2-c1905c31031b	1015
8	2024-11-13	1c365062-fa21-4e04-93b2-c1905c31031b	19
9	2024-11-14	6ed696ae-c202-4104-9d0f-f6ae3d74414e	1175
10	2024-11-18	9196ae15-9d7c-4209-903e-b5d7c1dc8cbf	920
11	2024-11-19	23e905a3-9928-4aed-8098-10e380f65c86	1057
12	2024-11-20	23e905a3-9928-4aed-8098-10e380f65c86	23
13	2024-11-21	24eacd72-5a64-4c74-9883-822c79fe4430	1050
14	2024-11-22	f9b0f9bc-1365-4602-b971-2ad6876a2021	1185
15	2024-11-25	12ae0296-4f95-4b0c-9fcd-28ae7a329109	1168
16	2024-11-26	71039db6-4806-4c14-bbc3-3d6107c5cc5c	1240

After

```
select SnapShot__Date__Key,meta_run_id,sum(safe_cast(Nb.Operation__Reactivities as float64))
from `prj-data-dm-industrial-dev.ODS.ODS_BWH_0000_F012_F_W_mec_mvpmop04_0006`
where meta_run_id = '6662b4b8-4cae-411b-9b3c-966dbe996bb8'
group by SnapShot__Date__Key,meta_run_id
order by SnapShot__Date__Key
limit 100
```

Query results

INFORMATION	RESULTS	CHART	JSON	EXECUTION DETAILS	EXECUTION GRA
SnapShot__Date__Key	meta_run_id	f0_			
04.11.2024	6662b4b8-4cae-411b-9b3c-966...	1114.0			
05.11.2024	6662b4b8-4cae-411b-9b3c-966...	929.0			
06.11.2024	6662b4b8-4cae-411b-9b3c-966...	993.0			
07.11.2024	6662b4b8-4cae-411b-9b3c-966...	1025.0			
08.11.2024	6662b4b8-4cae-411b-9b3c-966...	1288.0			

BW query result.

Query Technical Name	QVMECANO_BW_QRY_MVPMOP04_0006				
SnapShot (Week)	45.2024 - 48.2024				
SnapShot Frequency	DAILY				
SnapShot (Week)	45.2024				
SnapShot (Date)	04.11.2024	05.11.2024	06.11.2024	07.11.2024	08.11.2024
	Nb Operation (Reactivities)	Nb Operation (Reactivities)	Nb Operation (Reactivities)	Nb Operation (Reactivities)	Nb Operation (Reactivities)
	1,114	929	993	1,025	1,288

5. Repeat changing parameter on point 3 until current week in this case 48.204

⚠ This query can't select loading in between since it is required to delete all new data first.

F13: QVMECANO_BW_QRY_MVPMOR04_0001

Selection parameter = WEEK(yyyyww)

Fact table = FACT_maintenance_efficiency

This query will impact on kpi_no = 2.12/2.03/2.14

- nb_of_orders_in_progress
- kpi_2_11_nb_of_orders_executed_with_a_required_end_date
- kpi_2_11_nb_of_orders_executed_before_required_date
- kpi_2_11_nb_of_orders_executed_without_a_required_end_date
- kpi_2_11_number_of_orders_executed

Validate

```
select reference_week,
sum(nb_of_orders_in_progress) nb_of_orders_in_progress,
sum(kpi_2_11_nb_of_orders_executed_with_a_required_end_date) kpi_2_11_nb_of_orders_executed_with_a_required_end_date,
sum(kpi_2_11_nb_of_orders_executed_before_required_date) kpi_2_11_nb_of_orders_executed_before_required_date,
sum(kpi_2_11_nb_of_orders_executed_without_a_required_end_date) kpi_2_11_nb_of_orders_executed_without_a_required_end_date,
sum(kpi_2_11_number_of_orders_executed) kpi_2_11_number_of_orders_executed,
from `prj-data-maintenance-dash-dev.DPL.V_REP_FACT_maintenance_dash`
where kpi_no = '2.12/2.03/2.14'
and reference_week between '41.2024' and '48.2024'
and planning_plant_key like '%GRU%'
group by reference_week
order by reference_week
```

Reload

For example to reload 41.2024

1. Stop the loading in TMC.
2. it is required to delete the data from DM.FACT_maintenance_efficiency in prj-data-dm-industrial-dev

```
delete from `DM.FACT_maintenance_efficiency`
```

```
where reference_week = '41.2024' and kpi_no = '2.12/2.03/2.14'
```

3. Change the variable

```
I_LOCAL_VAR_RECOVERY_MODE = true
```

```
I_LOCAL_VAR_HISTO_YEAR = 2024
```

```
I_LOCAL_VAR_HISTO_WEEK_END = 41
```

4. Control the query to be run on the "valid" I_VAR_mecano_xtract_MVPMOR04_0001_valid = 1 and select the other to 0 in case we don't want to reload all the queries

F14: QVMECANOBW_QRY_MVPMOR01_0002

Selection parameter = DAY(yyyyMMdd,yyyyMMdd)

Fact table = FACT_maintenance_effectiveness

This query will impact on kpi_no = 3.03

- work_orders

Validate

DPL

```

select reference_date,
sum(work_orders) work_orders,
from `prj-data-maintenance-dash-dev.DPL.V_REP_FACT_maintenance_dash`
where kpi_no = '3.03'
and reference_date between '2024-11-07' and '2024-12-01'
and planning_plant_key like '%GRU%'
group by reference_date
order by reference_date

```

⚠ reference_date in this query was changed mapping to created_on

ODS

```

SELECT Reference_date_Key, meta_business_date, sum(safe_cast(Number_of_Work_Orders as float64)) Number_of_Work_Orders
FROM `prj-data-dm-industrial-dev.ODS.ODS_BWH_0000_F014_F_W_mec_mvpmor01_0002`
where PM_planning_plant_Key like '%GRU%'
and Created_on like '%2024%'
group by Created_on, meta_business_date
order by Created_on

```

Reload

For example to reload 41.2024

1. Stop the loading in TMC.
2. it is required to delete the data from DM.FACT_maintenance_effectiveness in prj-data-dm-industrial-dev

```
delete from `DM.FACT_maintenance_effectiveness`
```

where reference_day **between** '2024-10-07' to '2024-10-11' and kpi_no = '3.03'

3. Change the variable

```
I_LOCAL_VAR_RECOVERY_MODE = true
```

```
I_LOCAL_VAR_MVPMOR01_YYYYMMDD_FROM = 20241007
```

```
I_LOCAL_VAR_MVPMOR01_YYYYMMDD_TO = 20241011 ***Must NOT over current Monday
```

4. Control the query to be run on the "valid" I_VAR_mecano_xtract_MVPMOR01_0002_valid = 1 and select the other to 0 in case we don't want to reload all the queries.

F15: QV_BW_QRY_MVPMOP02_0010

Selection parameter = WEEK(yyyyww)

Fact table = FACT_maintenance_efficiency

This query will impact on kpi_no = 2.11

- barcode_scan_rate
- nb_operations_well_scanned
- nb_operations_completed

Validate

```

select reference_week,
sum(barcode_scan_rate) barcode_scan_rate,
sum(nb_operations_well_scanned) nb_operations_well_scanned,
sum(nb_operations_completed) nb_operations_completed
from `prj-data-maintenance-dash-dev.DPL.V_REP_FACT_maintenance_dash`
where kpi_no = '2.11'
and reference_week between '41.2024' and '48.2024'
and planning_plant_key like '%GRU%'
group by reference_week
order by reference_week

```

Reload

For example to reload 41.2024

1. Stop the loading in TMC.
2. it is required to delete the data from DM.FACT_maintenance_efficiency in prj-data-dm-industrial-dev

delete from `DM.FACT\_maintenance\_efficiency`

where reference_week = '41.2024' and kpi_no = '2.11'

3. Change the variable

I_LOCAL_VAR_RECOVERY_MODE = true

I_LOCAL_VAR_HISTO_YEAR = 2024

I_LOCAL_VAR_HISTO_WEEK_END = 41

4. Control the query to be run on the "valid" I_VAR_mecano_xtract_MVPMOP02_0010_valid = 1 and select the other to 0 in case we don't want to reload all the queries

F16: QVMECANO_BW_QRY_MVPMCO01_0001

Selection parameter = MONTH(yyyy0MM,yyyy0MM)

Fact table = FACT_maintenance_dash_strategic

This query will impact on kpi_no = 1.10

- actual_amount
- plan_amount

This query will select by month and the job will load the new data from ODS to WDL dataset in temp table first. Then, it will select the date from WDP to delete on the FACT_maintenance_dash_strategic. Therefore, no need to select delete.

DELETE FROM DM.FACT_maintenance_strategic

WHERE ref_date IN (SELECT DISTINCT CAST(ref_date as Date format 'MON YYYY') as ref_date FROM WDL.
FACT_maintenance_dash_strategic_cost_budget_tmp)

⚠ WDL.FACT_maintenance_dash_strategic_cost_budget_tmp table will have data from the last loading from ODS (truncate)

Validate

select reference_date,currency_key,currency_type_key,

sum(actual_amount) actual_amount,

sum(plan_amount) plan_amount,

from `prj-data-maintenance-dash-dev.DPL.V\_REP\_FACT\_maintenance\_dash`

where kpi_no = '1.10'

and reference_date between '2024-09-01' and '2024-12-01'

and planning_plant_key like '%GRU%'

group by reference_date,currency_key,currency_type_key

order by reference_date,currency_type_key

Reload

For example, need to reload Oct 2024

1. Stop the loading in TMC.
2. Change the variable

I_LOCAL_VAR_RECOVERY_MODE = true

I_LOCAL_VAR_MVPMCO01_0001_HIST_K4YYYY0MM_PREV = 2024010

I_LOCAL_VAR_MVPMCO01_0001_HIST_K4YYYY0MM_CURR = 2024010

3. Control the query to be run on the "valid" I_VAR_mecano_xtract_mvpmco01_valid = 1 and select the other to 0 in case we don't want to reload all the queries