

Purchasing Dashboard 4. Data Loading with Talend

4.1 - Talend Integration

Source data integration with Talend ETL tool

- **Xtract Server (WBP):**
 - Talend connects to the Xtract server (*ACEW1PQLIKAPP02*) trigger extraction job using HTTP calls.
 - Xtract generates results as CSV files stored in memory.
 - Talend retrieves files, apply some transformation and then loads them into Google Cloud Storage.
- **Google Sheets (Hedges, wap solid fuels, CO2):**
 - Talend integrates with Google Sheets to extract some static files that will be use to apply mappings or lookup to BW tables.
 - It retrieves this data from Google Sheets.

Data Transformation and Loading to Google BigQuery:

- Once data from all sources is available in Google Cloud Storage as files, Talend proceeds with data transformation and loading.
- Talend performs data transformations as needed, including cleansing, mapping, and structuring the data for consistency.
- The transformed data is loaded into various stages, operational data stores (ODS), and data mart tables within Google BigQuery.
- These tables are organized to facilitate efficient querying and reporting for energy optimization purposes.

By utilizing Talend for data extraction, transformation, and loading (ETL), the web app ensures that data from diverse sources is collected, processed, and structured for analysis and reporting within Google BigQuery, enabling users to make informed decisions based on up-to-date and accurate data.

4.2 - Source Data Extraction

- 4.1 - Talend Integration
- 4.2 - Source Data Extraction
 - F100_PUR_DATA_EXTRACT
 - J100_PURCHASING_EXTRACT_SOURCE_3
 - J100_PURCHASING_EXTRACT_SOURCE_6
 - J100_PURCHASING_EXTRACT_SOURCE_7
 - J100_PURCHASING_EXTRACT_SOURCE_11
 - J100_PURCHASING_EXTRACT_SOURCE_13
 - F300_PUR_STG_TO_ODS
 - J300_PUR_STG_TO_ODS
 - F400_PUR_PREP_DATA
 - F410_PURCHASING_STEP
 - F420_PURCHASING_STEP
- 4.4 - Load to DM (calculations and transformations)
 - F500_PUR_DATA_TO_DTM
 - J500_PURCHASING_PIVOT_TO_DTM
 - J520_PURCHASING_SPEND_TO_DTM
- 4.5 - Scheduling and Automation
- 4.6 - Data Validation

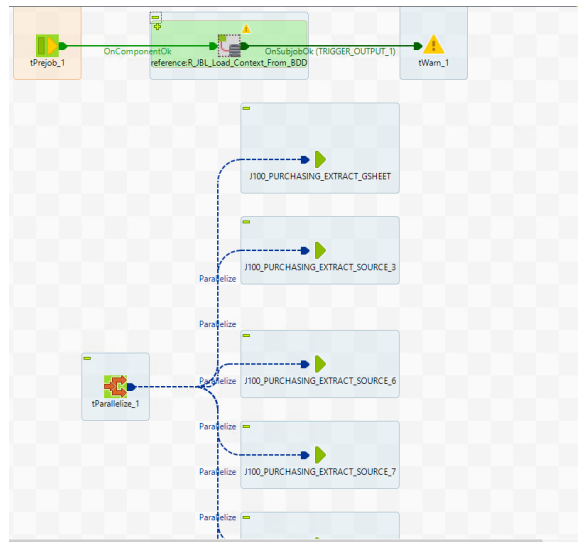
Responsible & contact points:

- Application Owner - Sylvie Severini
- Delivery Manager - Donia Rachdi
- Project Manager - Vitaly Altukhov
- Data Architect - Joao Fonseca
- Tableau Dev - Maxime Marboeuf
- Data Engineer - Virgil Lissassi; replaced by Matteo Menghetti

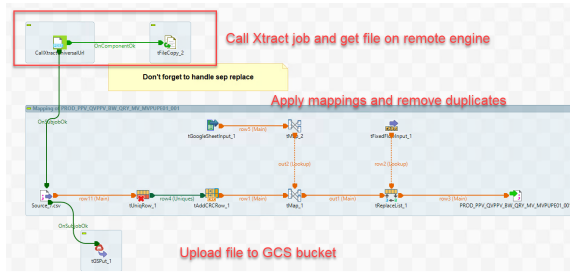
| | | |
|--|---|--|
| <p>Main jobs for source extraction</p> | <ul style="list-style-type: none"> • F100_P UR_DA TA_EX TRACT • J100_P PURCH ASING _EXTR ACT_S OURCE _3 • J100_P PURCH ASING _EXTR ACT_S OURCE _6 • J100_P PURCH ASING _EXTR ACT_S OURCE _7 • J100_P PURCH ASING _EXTR ACT_S OURCE _11 • J100_P PURCH ASING _EXTR ACT_S OURCE _13 | <p style="text-align: center;">--to the top --</p> |
| | <p>Job description by steps</p> | <p>Job design</p> |

1. In the job F100_PUR_DATA_EXTRA_CT:
a. A parallel execution of all data sources is launched

We will analyze in detail on job but the logic in the other is similar

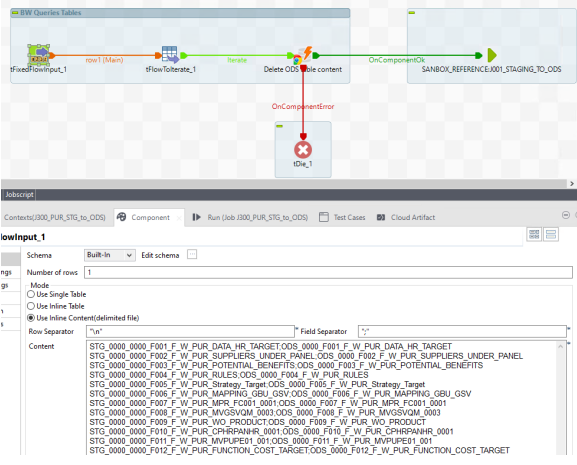


1. The jobs triggers the extract universal jobs. In some cases we trigger loop over a list of parameters and trigger multiple extractions
2. We read the CSV file obtained by extract and we :
 - a. Remove duplicates
 - b. Apply mappings as indicated in the specifications and create a file ready to be uploaded to staging
3. We upload the file into the bucket.



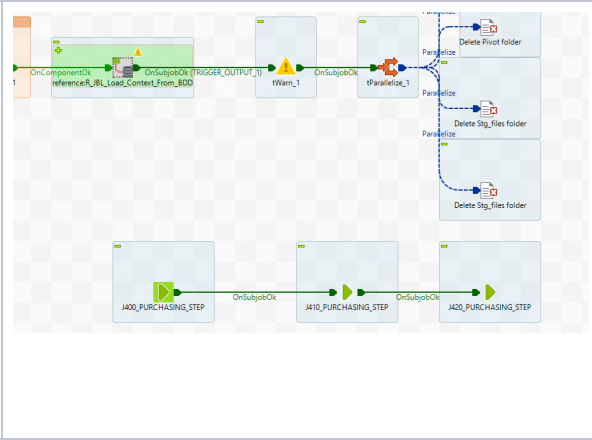
| | | |
|--------------------------------------|--|-------------------|
| Main jobs for loading Staging | <ul style="list-style-type: none"> • F200_PUR_GCS_TO_STG • J200_PUR_GCS_TO_STG | --to the top -- |
| | Job description by steps | Job design |
| | 1. The flow is used to call the job J200_PUR_GCS_TO_STG: | |

i. The Fixed Flow Input component contains the list of all files with the corresponding tag name.

| | | |
|--------------------------------------|---|--|
| <p>Main jobs for loading the ODS</p> | <ul style="list-style-type: none"> • F300_PUR_STG_TO_ODS • J300_PUR_STG_TO_ODS | <p>--to the top --</p> |
| <p>Job description by steps</p> | | <p>Job design</p> |
| | <ol style="list-style-type: none"> 1. The flow is used to call the job J300_PUR_STG_TO_ODS: 2. Looping over the list of staging tables: <ol style="list-style-type: none"> a. Truncate the corresponding ODS table b. Call the job J001_STAGING_TO_ODS to insert data in the table |  <p>The screenshot shows a flow diagram with components: 'flowinput_1', 'flowtolerate_1', 'Delete ODS table content', 'OnComponentOK', and 'SANBOK_REFERENCE001_STAGING_TO_ODS'. Below it is a table content view for 'lowinput_1' with a 'Content' field containing a list of staging table names and their corresponding ODS tables, such as 'STG_0000_0000_F001_F_W_PUR_DATA_HR_TARGET_ODS_0000_F001_F_W_PUR_DATA_HR_TARGET'.</p> |

| | | |
|----------------------------------|---|------------------------|
| <p>Main jobs for building DM</p> | <ul style="list-style-type: none"> • F400_PUR_PREP_DATA <ul style="list-style-type: none"> ◦ F400_PUR_CHASING_STEP ◦ F410_PUR_CHASING_STEP ◦ F420_PUR_CHASING_STEP | <p>--to the top --</p> |
| <p>Job description by steps</p> | | <p>Job design</p> |

1. This Talend flow first delete files that remain in the remote engine from previous executions
2. It call in sequece 3 subjobs which read the different ODS tables needed to build the dimension and pivot table.



| | | |
|------------------------|---------------------------------|------------------------|
| Prepare DM file | F400_PURCHASING_STEP | --to the top -- |
| | Job description by steps | Job design |

1. Reqd data from DATA_HR_TARGET GET:

- The job reads data from the ODS table ODS_0000_F001_F_W_PUR_D ATA_HR_TARGET and apply the necessary mapping

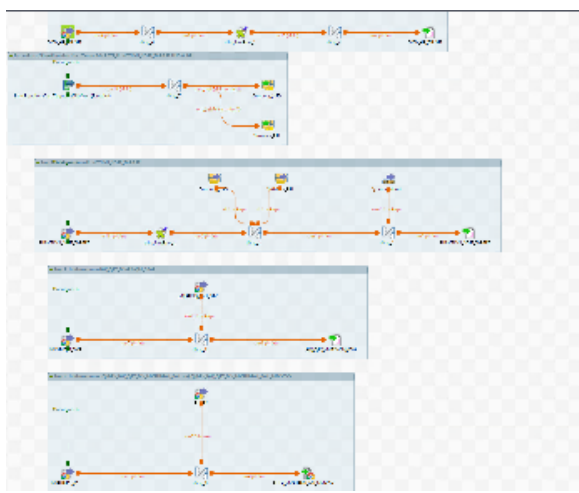
2. Calculation of rate cost:

- The job reads this [Google Sheet](#) in order to extract the various currency rate per year and saves them in memory.

3. Compute Function cost target:

- The job read data from the ODS table ODS_0000_F012_F_W_PUR_FUNCTION_COST_TARGET and apply mapping to build the FUNCTION_COST_TARGET

This job builds the first pieces of data necessary to build the FUNCTION cost target

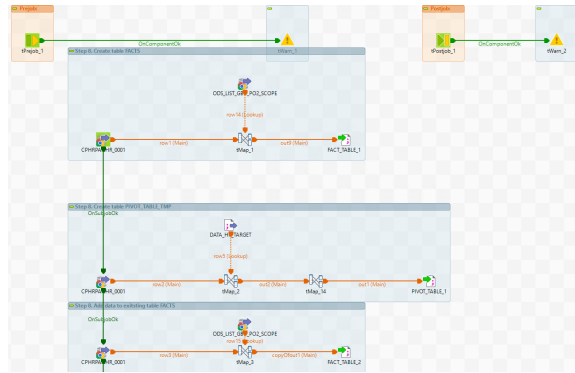


| | | |
|---|--|--------------------------|
| <p>Main jobs for source extraction</p> | <ul style="list-style-type: none"> • F410_PURCHASING_STEP | <p>--to the top --</p> |
| | <p>Job description by steps</p> | <p>Job design</p> |
| | <p>The job reads in sequence a list of ODS tables and apply the requested mappings</p> | |

1. ODS_0000_F010_F
W_PUR_C
PHRPNH
R_0001

a. The table is read twice

i. Reads ODS data and apply filter BUS - Unit - Ent - Grp - Key, FPUR and getting only the latest



two year of data.
ii. Reads ODS data applying filter SPLT (Job - Function, /) [OFFSET(0)] in PROC; and

g
e
t
t
i
n
g
o
n
l
y
t
h
e
l
a
s
t
t
w
o
y
e
a
r
o
f
d
a
t
a.

2. ODS_0000
_F007_F_
W_PUR_M
PR_FC001
_0001:

a. We
read
the
entire
table
and
apply
the
reque
sted
mappi
ngs

3. FUNCTIO
N_COST_
TARGET

a. The
file
create
d in
job
J400_
PURC
HASI
NG_S
TEP
is
parse
d to
adapt
it to
the
dimen
sion
and
fact
schema

4. ODS_0000
_F009_F_
W_PUR_W
O_PRODU
CT

a. We read the table applying the following filters on *Credit or_Class_P__Key* *not like ('%S%' OR '%X%')* and getting only the last two year of data.

5. ODS_0000
_F008_F_
W_PUR_M
VGSVQM_
0003:

a. We read the table applying the following filters on excluding when *Vendor_Domain_Key* contains 9 and getting only the last two year of data.

This Talend job is the second step of building Dimension and Fact table. All data in FACT tables are also assigned to ECO or SCO.

Main jobs for source extraction

- F420_PURCHASING_STEP

--to the top --

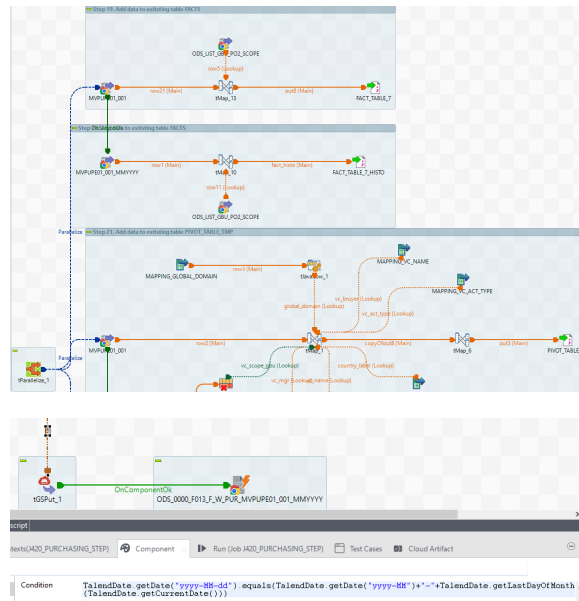
Job description by steps

Job design

The job reads in sequence a list of ODS tables and apply the requested mappings

1. ODS_0000_F011_F_W_PUR_MVPUPE01_001

- a. We read the table applying the following filters on VC_PROGRAM = 'Procurement' and getting only the last two year of data.



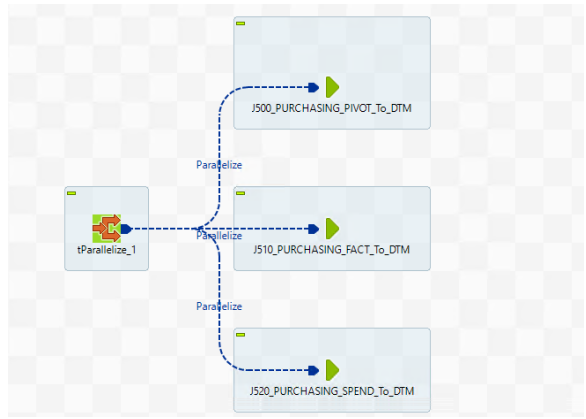
```

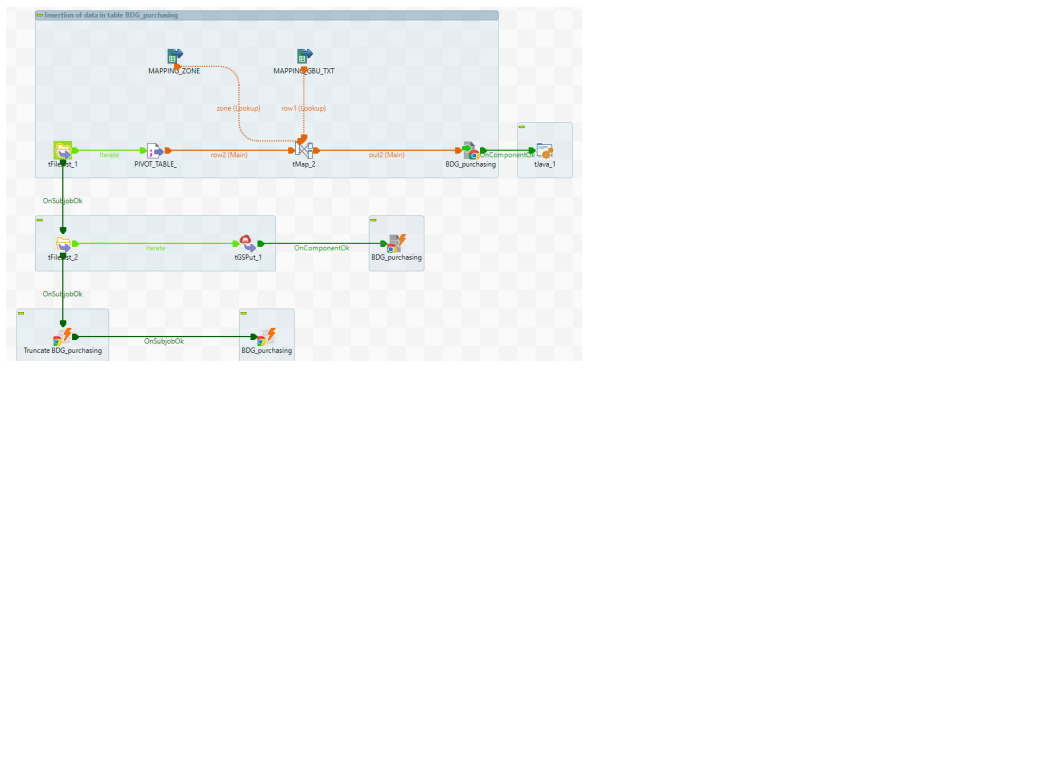
Condition
TalendDate.getDate("yyyy-MM-dd") equals(TalendDate.getDate("yyyy-MM")+ "-" + TalendDate.getLastDayOfMonth(
TalendDate.getCurrentDate()))

```

2. ODS_000_0_F005_F_W_PU_R_Strategy_Target
 - a. We read the table applying the following filters for getting only the last two year of data. and where TARGET_EUR is not empty.
3. At the end of step there is a condition **that verifies if the current date is the last day of the month** , if true it appends the content of the table ODS_000_0_F013_F_W_PU_R_MVPU_PE01_001 to the table ODS_000_0_F013_F_W_PU_R_MVPU_PE01_001_MMYYY

4.4 - Load to DM (calculations and transformations)

| | | |
|--|---|--|
| Main jobs for source extraction | <ul style="list-style-type: none"> • F500_PUR_DATA_TO_DTM | --to the top -- |
| | Job description by steps | Job design |
| | This jobs reads the files stored in remote engine and computed in the previous and use them to populate dimensions and fact tables. |  |

| | | |
|--|---|--|
| Main jobs for source extraction | <ul style="list-style-type: none"> • J500_PURCHASING_PIVOT_To_DTM | --to the top -- |
| | Job description by steps | Job design |
| | 1. The job starts by truncating the table <i>BDG_purchasing_tmp</i> |  |

2. Read Pivot files:

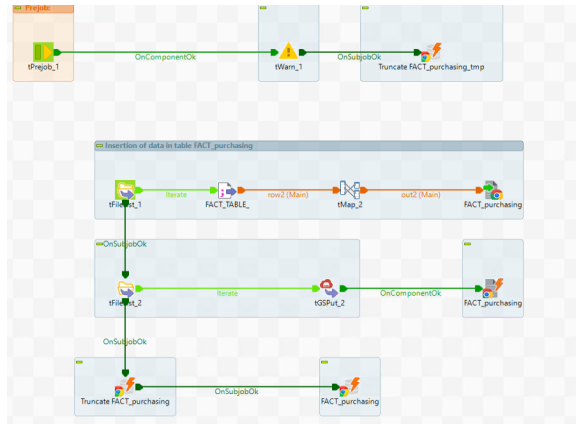
- The job iterates over all files of the *Pivot* folder (whose name starts with *PIVOT_TABLE_*).
- Each file is processed and a lookup is done with the following Google Sheets to get a uniform GBU text.
- An BigQuery Bulk Output file is created

| | | |
|--|--|--|
| | <p>3. Upload data in the <i>BDG_purchasing_tmp</i> table:</p> <p>a. The job upload all bulk files into Cloud Storage and each file is appended into the <i>BDG_purchasing_tmp</i> table</p> <p>4. The job truncates the table <i>BDG_purchasing</i>.</p> <p>5. . The job inserts copies data into from the <i>BDG_purchasing_tmp</i> table. During the copy the columns <i>mont_year</i> and <i>vc_year</i> are converted into string.</p> | |
|--|--|--|

| | | |
|--|---|---------------------------------|
| Main jobs for source extraction | <ul style="list-style-type: none"> • J510_PURCHASING_FACT_TO_DTM | --to the top -- |
| | Job description by steps | Job design |
| | <p>1. The job starts by truncating the table <i>FACT_purchasing_tmp</i></p> | |

2. Read Pivot files:

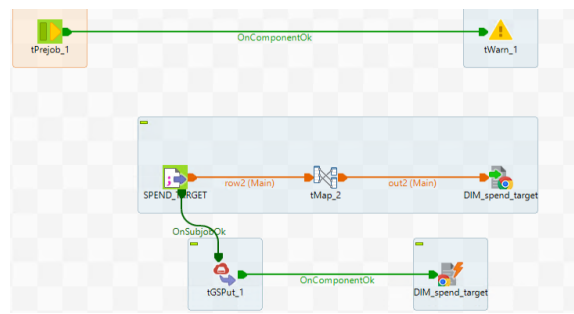
- The job iterates over all files of the Pivot folder (whose name starts with *FACT_TABLE_*). Each file is processed to have the desired schema of the fact table.
- An BigQuery Bulk Output file is created



| | | |
|--|---|--|
| | <p>3. Upload data in the <i>FACT_purchasing_tmp</i> table:</p> <p>a. The job uploads all bulk files into Cloud Storage and each file is appended into the <i>FACT_Purchasing_tmp</i> table</p> <p>4. The job truncates the table <i>FACT_purchasing</i>.</p> <p>5. . The job inserts copies data into from the <i>FACT_purchasing_tmp</i> table</p> | |
|--|---|--|

| | | |
|---------------------------------|---|---------------------------------|
| Main jobs for source extraction | <ul style="list-style-type: none"> • J520_PURCHASING_SPEND_TO_DTM | --to the top -- |
| Job description by steps | Job design | |

1. Read SPEND_TARGET_1 file:
 - The job reads the file SPEND_TARGET_1.csv in the folder Spend_Target
 - A BigQuery Bulk Output file is created
2. Upload the file bulk file in Google Cloud Storage
3. The job uses the bulk file to overwrite the values in the table DIM_spend_target.



4.5 - Scheduling and Automation

TMC - PL_QV_TO_TABLEAU_PURCHASING - Daily at 5 AM.

4.6 - Data Validation