KPIs Interface DataBase

Michela Deodati

September 18, 2023

Contents

1 KPIs Databse

1.1 Disclaimer

The database structure shown here is the result of a taxonomic categorisation of KPIs (Key Performance Indicators) that I created for my three-year thesis project, the database is far from complete, the KPIs shown here are those that I was able to measure on the data collection database model provided by Zerynth (https://it.zerynth.com).

1.2 Database Structure

The database structure is table-based. The table in question was created using PostgreSQL. The columns of the table are:

- **id**: is the key value of the table and contains the full name of the KPIs present in the form "*SnakeCase*" (example: tot_wk_hours).
- **section**: describes the sub-tree to which the KPIs belong, in other words expresses the category in which the evaluation of the indicators is of interest.
- **children**:array of strings representing a list of KPIs that are exploited for the calculation of that of the corresponding row.
- **query**: is the column containing the parameterised query with the dollar sign followed by curly braces (${ ... }) used to evaluate and embed parameter dynamically, to make the database interface more scalable.
- **js_fun**: string containing the code of a pointer to a javascript function which is then used to calculate the value of the indicator corresponding to the function.

- **description**: column containing the descriptions of the corresponding KPIs.

1.2.1 Constraints

Only one constraint is applied to the Database: the 'mutually_exclusive_columns' constraint. The rule was applied as:

```
ALTER TABLE kpi_table
ADD CONSTRAINT mutually_exclusive_columns
CHECK ((query IS NULL AND children IS NULL)
      OR(query IS NULL AND children IS NOT NULL)
      OR(query IS NOT NULL AND children IS NULL));
```

this means that a row either contains the value for the column '*children*' or for the column '*query*' or both are null.

1.3 Data

In this section I will list the contents of the database, with each KPI id, section, description, query or children. Each row returning from the query has the value in the column set 'as v', so that the interface accessing it knows exactly which value to return.

1.3.1 Machine Usage:

id: average_idle_time

- **description**: This KPI measures the average time that machines spend in idle state
- **query**:

```
select round(avg(tot_idle_time), 1) as v
from (select device_id, sum((payload->'T'->0)::numeric) as tot_idle_time
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDate},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDate},'YYYY-MM-DD')
group by device_id )as sub_tb;
```

id: average_idle_time_past

- **description**: This KPI measures the average time that machines spend in idle state during past periods
- **query**:

```
select round(avg(tot_idle_time), 1) as v
from (select device_id, sum((payload->'T'->0)::numeric) as tot_idle_time
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDatePast},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDatePast},'YYYY-MM-DD')
group by device_id )as sub_tb;
```

id: average_wk_time

- **description**: This KPI measures the average time that machines spend in work state
- **query**:

```
select round(avg(tot_wk_time), 1) as v
from (select device_id, sum((payload->'T'->1)::numeric) as tot_wk_time
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDate},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDate},'YYYY-MM-DD')
group by device_id )as sub_tb;
```

id: average_wk_time_past

- **description**: This KPI measures the average time that machines spend in work state
- **query**:

```
select round(avg(tot_wk_time), 1) as v
from (select device_id, sum((payload->'T'->1)::numeric) as tot_wk_time
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDatePast},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDatePast},'YYYY-MM-DD')
group by device_id )as sub_tb;
```

id: highest_variability_wk_time

- **description**: This KPI is calculated as the coefficient of variability, i.e. standard deviation divided by the mean per cent
- **children**:

```
{average_wk_time,tot_connected_machine}
```

id: least_used_machine_id

- **description**: This KPI is the name of the least used machine
- **query**:

```
select device_id as v
from (select device_id, sum((payload->'T'->1)::numeric) as tot_wk_time
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDate},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDate},'YYYY-MM-DD')
group by device_id order by tot_wk_time asc LIMIT 1)as sub_tb;
```

id: least_used_machine_time

- **description**:This KPI measures the time spent to work of the least used machine
- **query**:

```
select min(tot_wk_time) as v
from (select device_id, sum((payload->'T'->1)::numeric) as tot_wk_time
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDate},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDate},'YYYY-MM-DD')
group by device_id)as sub_tb;
```

id: machine_usage_trend

- **description**: This Kpi calculates the trend between 2 time periods and returns the percentage of difference between them
- **children**:

```
{average_wk_time,average_wk_time_past}
```

id: machine_utilization_rate

- **description**:This KPI expresses the percentage of working time out of the total operating time
- **children**:

```
{tot_wk_time}
```

id: most_used_machine_id

- **description**:This KPI indicates the most used machine of the period
- **query**:

```
select device_id as v
from (select device_id, sum((payload->'T'->1)::numeric) as tot_wk_time
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDate},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDate},'YYYY-MM-DD')
group by device_id order by tot_wk_time desc LIMIT 1)as sub_tb;
```

id: most_used_machine_id_past

- **description**:This KPI indicates the most used machine of past periods
- **query**:

```
select device_id as v
from (select device_id, sum((payload->'T'->1)::numeric) as tot_wk_time
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDatePast},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDatePast},'YYYY-MM-DD')
group by device_id order by tot_wk_time desc LIMIT 1)as sub_tb;
```

id: most_used_machine_time

- **description**:This KPI measure the time spent working most used machine
- **query**:

```
select max(tot_wk_time) as v
from (select device_id, sum((payload->'T'->1)::numeric) as tot_wk_time
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDate},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDate},'YYYY-MM-DD')
group by device_id )as sub_tb;
```

id: most_used_machine_time_past

- **description**:This KPI measure the time spent working most used machine of past periods
- **query**:

```
select max(tot_wk_time) as v
from (select device_id, sum((payload->'T'->1)::numeric) as tot_wk_time
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDatePast},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDatePast},'YYYY-MM-DD')
group by device_id )as sub_tb;
```

id: tot_connected_machine

- **description**:This KPI counts how many machines are connected to the system during the period
- **query**:

```
select count(device_id) as v
from (select device_id from \{dataTableName}
where ts_device >= TO_TIMESTAMP(\${startDate},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(\${endDate},'YYYY-MM-DD')
group by device_id)as sub_t;
```

id: tot_idle_time

- **description**:This KPI measures the total time that all machines spent in idle state
- **query**:

```
select sum(tot_idle_time) as v
from (select device_id, sum((payload->'T'->0)::numeric) as tot_idle_time
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDate},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDate},'YYYY-MM-DD')
group by device_id)as sub_tb;
```

id: tot_idle_time_past

- **description**:This KPI measures the total time that all machines spent in idle state of past periods
- **query**:

```
select sum(tot_wk_time) as v
from (select device_id, sum((payload->'T'->0)::numeric) as tot_wk_time
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDatePast},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDatePast},'YYYY-MM-DD')
group by device_id )as sub_tb;
```

id: tot_machine_used_percentage_of_period

- **description**:This KPI counts how many machines were used for more than an arbitrary time expressed as a percentage of the operating time
- **query**:

```
select count(device_id) as v from(
select device_id, sum((payload->'TA'->1)::numeric) as s from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDate},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDate},'YYYY-MM-DD') group by device_id
)as sub_t
where s> \${timeSecond}
```

id: tot_wk_time

- **description**: This KPI measures the total time that all machines spent in work state
- **query**:

```
select sum(tot_wk_time) as v
from (select device_id, sum((payload->'T'->1)::numeric) as tot_wk_time
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDate},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDate},'YYYY-MM-DD')
group by device_id)as sub_tb;
```

id: tot_wk_time_past

- **description**: This KPI measures the total time that all machines spent in work state of past period
- **query**:

```
select device_id, sum((payload->'T'->1)::numeric) as tot_wk_time
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDate},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDate},'YYYY-MM-DD')
group by device_id
```

1.3.2 Power Consumption:

id: average_power_consumption

- **description:**This KPI measures the average consumption over the period, taking into account both states.
- **query:**

```
select avg(v) as v
from(select device_id ,
avg(((payload->'p_status'->0)::numeric)
+((payload->'p_status'->1)::numeric)) as v
from data
where ts_device >= TO_TIMESTAMP('2023-06-02','YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP('2023-07-15','YYYY-MM-DD')
group by device_id )as sub_t;
```

id: average_power_consumption_idle_state

- **description:**This KPI measures the average consumption over the period, taking into account idle states.

- **query:**

```
select avg(v) as v
from(select device_id ,
avg(((payload->'p_status'->0)::numeric)
) as v
from data
where ts_device >= TO_TIMESTAMP('2023-06-02','YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP('2023-07-15','YYYY-MM-DD')
group by device_id )as sub_t;
```

id: average_power_consumption_wk_state

- **description:**This KPI measures the average consumption over the period, taking into account work states.

- **query:**

```
select avg(v) as v
from(select device_id ,
avg(((payload->'p_status'->1)::numeric)
) as v
from data
where ts_device >= TO_TIMESTAMP('2023-06-02','YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP('2023-07-15','YYYY-MM-DD')
group by device_id )as sub_t;
```

id: highest_power_consumption

- **description:**This KPI measures the highest value of machinery power consumption

- **query:**

```
Select device_id, Round((sum((payload ->'e_status'->1)::numeric)
+sum((payload ->'e_status'->0)::numeric)),2) as v
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDate},'YYYY-MM-DD')
```

```
AND ts_device <= TO_TIMESTAMP(${endDate},'YYYY-MM-DD')
group by device_id order by v desc LIMIT 1
```

id: highest_powe_consumption_idle_state_machine_id

- **description:**This KPI indicates the machine with the highest value of power consumption in idle state
- **query:**

```
select device_id as v, sum((payload->'e'->0)::numeric) as s
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDate},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDate},'YYYY-MM-DD')
group by device_id order by s desc LIMIT 1
```

id: highest_power_consumption_idle_state_machine_tot_value

- **description:**This KPI measures the highest total value of machine power consumption in idle state
- **query:**

```
select max(tot_pw_cons_idle) as v
from (select device_id,
sum((payload->'e_status'->0)::numeric) as tot_pw_cons_idle
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDate},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDate},'YYYY-MM-DD')
group by device_id )as sub_tb;
```

id: highest_powe

- **description:**This KPI measures the highest total value of machine power consumption in idle state broken down by hour
- **children:**

```
{highest\_power\_consumption\_idle\_state\_machine\_tot\_value}
```

id: highest_power_consumption_machine_id

- **description:**This KPI indicates the machine with the highest power consumption value
- **query:**

```
Select device_id as v, Round((sum((payload ->'e_status'->1)::numeric)
+sum((payload ->'e_status'->0)::numeric)),2) as v2
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDate},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDate},'YYYY-MM-DD')
group by device_id order by v2 desc LIMIT 1
```

id: highest_power_consumption_wk_state_machine_id

- **description:**This KPI indicates the machine with the highest value of power consumption in work state
- **query:**

```
select device_id as v
from (select device_id,
sum((payload->'e_status'->1)::numeric) as tot_pw_cons_wk
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDate},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDate},'YYYY-MM-DD')
group by device_id order by tot_pw_cons_wk desc LIMIT 1)as sub_tb;
```

id: highest_power_consumption_wk_state_machine_tot_value

- **description:**This KPI measures the highest total value of machine power consumption in work state
- **query:**

```
select max(tot_pw_cons_wk) as v
from (select device_id, sum((payload->'e_status'->1)::numeric) as tot_pw_cons_wk
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDate},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDate},'YYYY-MM-DD')
group by device_id )as sub_tb;
```

id: highest_power_consumption_wk_state_machine_value_per_hour

- **description:**This KPI measures the highest total value of machine power consumption in work state broken down by hour
- **children:**

{highest_power_consumption_wk_state_machine_tot_value}

id: lowest_power_consumption

- **description:**This KPI measures the minimum value of machinery power consumption
- **query:**

```
Select device_id, Round((sum((payload ->'e_status'->1)::numeric)
+sum((payload ->'e_status'->0)::numeric)),2) as v
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDate},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDate},'YYYY-MM-DD')
group by device_id order by v asc LIMIT 1
```

id: lowest_power_consumption_idle_state_machine_id

- **description:**This KPI indicates the machine with the lowest value of power consumption in idle state
- **query:**

```
select device_id as v from
( select device_id, sum((payload->'e_status'->0)::numeric) as pw_cons_idle_state
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDate},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDate},'YYYY-MM-DD')
group by device_id order by pw_cons_idle_state
asc LIMIT 1)as sub_t;
```

id: lowest_power_consumption_idle_state_machine_tot_value

- **description:**This KPI measures the lowest total value of machine power consumption in idle state

- **query:**

```
select min(pw_cons_idle_state) as v from
( select device_id, sum((payload->'e_status'->0)::numeric) as pw_cons_idle_state
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDate},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDate},'YYYY-MM-DD') group by device_id
)as sub_t;
```

id: lowest_power_consumption_idle_state_value_per_hours

- **description:**This KPI measures the lowest total value of machine power consumption in idle state broken down by hour

- **children:**

```
{lowest_power_consumption_idle_state_machine_tot_value}
```

id: lowest_power_consumption_machine_id

- **description:**This KPI indicates the machine with the minmum power consumption value

- **query:**

```
Select device_id as v, Round((sum((payload ->'e_status'->1)::numeric)
+sum((payload ->'e_status'->0)::numeric)),2) as v2
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDate},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDate},'YYYY-MM-DD')
group by device_id order by v2 asc LIMIT 1
```

id: lowest_power_consumption_wk_state_machine_id

- **description:**This KPI indicates the machine with the lowest value of power consumption in work state

- **query:**

```
select device_id as v from
( select device_id, sum((payload->'e_status'->1)::numeric) as pw_cons_wk_state
from ${dataTableName}
```

```
where ts_device >= TO_TIMESTAMP(${startDate},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDate},'YYYY-MM-DD')
group by device_id order by pw_cons_wk_state
asc LIMIT 1 )as sub_t;
```

id: lowest_power_consumption_wk_state_machine_tot_value

- **description:**This KPI measures the lowest total value of machine power consumption in work state
- **query:**

```
select min(pw_cons_wk_state) as v from
( select device_id,
sum((payload->'e_status'->1)::numeric) as pw_cons_wk_state
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDate},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDate},'YYYY-MM-DD')
group by device_id )as sub_t;
```

id: lowest_power_consumption_wk_state_machine_value_per_hour

- **description:**This KPI measures the hlowest total value of machine power consumption in work state broken down by hour
- **children:**

```
{lowest_power_consumption_wk_state_machine_tot_value}
```

id: tot_power_consumption

- **description:**This KPI measures the total power consumption value over the period.
- **query:**

```
select sum(v2)as v from
(select device_id, Round((sum((payload ->'e_status'->1)::numeric)
+sum((payload ->'e_status'->0)::numeric)),2) as v2
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDate},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDate},'YYYY-MM-DD')
group by device_id )as sub_T;
```

1.3.3 Power Consumption Cost:

id: estimate_tot_cost

- **description:**This KPI measures the estimated total cost of energy consumption over the period.
- **children:**

```
{tot_power_consumption}
```

id: highest_power_consumption_idle_state_machine_cost

- **description:**This KPI measures the total cost of the machine that consumed the most in the idle state
- **children:**

```
{highest_power_consumption_idle_state_machine_tot_value}
```

id: highest_power_consumption_wk_state_machine_cost

- **description:**This KPI measures the total cost of the machine that consumed the most in the work state
- **children:**

```
{highest_power_consumption_wk_state_machine_tot_value}
```

id: lowest_power_consumption_idle_state_machine_cost

- **description:**This KPI measures the total cost of the machine that consumed the least in the idle state
- **children:**

```
{lowest_power_consumption_idle_state_machine_tot_value}
```

id: lowest_power_consumption_wk_state_machine_cost

- **description:**This KPI measures the total cost of the machine that consumed the least in the work state
- **children:**

```
{lowest_power_consumption_wk_state_machine_tot_value}
```

id: power_cost_per_hour

- **description:**This KPI measure the average cost per hour
- **children:**

```
{tot_power_consumption}
```

id: tot_power_cost_per_idle_hours

- **description:**This KPI measure the total cost for idle hours.
- **children:**

```
{tot_wk_time}
```

id: tot_power_cost_per_wk_hours

- **description:**This KPI measure the total cost for working hours.
- **children:**

```
{tot_idle_time}
```

1.3.4 Alarm:

id: average_alarm_time

- **description**:This KPI measures the average time that machine alarms last
- **query**:

```
select round(avg(wk_a + id_a), 1) as v
from(select device_id,sum((payload->'TA'->1)::numeric)
as wk_a, sum((payload->'TA'->0)::numeric) as id_a
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDate},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDate},'YYYY-MM-DD')
group by device_id ) as sub_t;
```

id: average_alarm_time_past

- **description**:This KPI measures the average time that machine alarms last of past periods
- **query**:

```
select round(avg(wk_a + id_a), 1) as v
from(select device_id,sum((payload->'TA'->0)::numeric) as wk_a,
sum((payload->'TA'->1)::numeric) as id_a
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDatePast},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDatePast},'YYYY-MM-DD')
group by device_id ) as sub_t;
```

id: highest_alarm_time_trend

- **description**:This KPI measures for all machines the alarm time trend and takes the maximum
- **query**:

```
select t1.device_id as v1, ROUND(max(((t1.var1 - t2.var2)/t2.var2)*100), 2) as v
from( select device_id ,
(sum((payload->'TA'->0)::numeric)+sum((payload->'TA'->1)::numeric))/2 as var1
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDate},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDate},'YYYY-MM-DD')
group by device_id ) as t1
JOIN (select device_id,
(sum((payload->'TA'->0)::numeric)+sum((payload->'TA'->1)::numeric))/2 as var2
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDatePast},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDatePast},'YYYY-MM-DD')
```

```
13 group by device_id)as t2
   ON t1.device_id = t2.device_id group by t1.device_id order by v desc LIMIT 1;
```

id: highest_alarm_time_trend_machine_id

- **description**:This KPI measures for all machines the alarm time trend and takes the machine id with the maximum one
- **query**:

```
   select t1.device_id as v, max(((t1.var1 - t2.var2)/t2.var2)*100) as v1
2  from( select device_id ,
   (sum((payload->'TA'->0)::numeric)+sum((payload->'TA'->1)::numeric))/2 as var1
4  from ${dataTableName}
   where ts_device >= TO_TIMESTAMP(${startDate},'YYYY-MM-DD')
6  AND ts_device <= TO_TIMESTAMP(${endDate},'YYYY-MM-DD')
   group by device_id ) as t1
8  JOIN (select device_id,
   (sum((payload->'TA'->0)::numeric)+sum((payload->'TA'->1)::numeric))/2 as var2
10 from ${dataTableName}
   where ts_device >= TO_TIMESTAMP(${startDatePast},'YYYY-MM-DD')
12 AND ts_device <= TO_TIMESTAMP(${endDatePast},'YYYY-MM-DD')
   group by device_id
14 )as t2
   ON t1.device_id = t2.device_id group by t1.device_id
16 order by v1 desc LIMIT 1;
```

id:longest_alarm_time

- **description**:This KPI measures the total alarm time for each machine and takes the maximum
- **query**:

```
   select case
2  when max(alarm_time_1) >= max(alarm_time_0) then max(alarm_time_1)
   else max (alarm_time_0)
4  end as v
   from(select device_id, ((payload->'TA'->1)::numeric) as alarm_time_1,
6  ((payload ->'TA'->0)::numeric) as alarm_time_0
   from ${dataTableName}
8  where ts_device >= TO_TIMESTAMP(${startDate},'YYYY-MM-DD')
   AND ts_device <= TO_TIMESTAMP(${endDate},'YYYY-MM-DD'))as sub_t;
```

id:longest_alarm_time_machine_id

- **description**:This KPI measures the total alarm time for each machine and the machine id with the highest value
- **query**:

```
select device_id as v
from(SELECT device_id,
Greatest(max(((payload->'TA'->0)::numeric)),max(((payload->'TA'->1)::numeric)))
AS max_v FROM ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDate},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDate},'YYYY-MM-DD')
group by device_id order by max_v desc LIMIT 1)as sub_t;
```

id:lowest_alarm_time_trend

- **description**:This KPI measures for all machines the alarm time trend and takes the minimum
- **query**:

```
 select t1.device_id, ROUND(min(((t1.var1 - t2.var2)/t2.var2)*100), 2) as v
from( select device_id , (sum((payload->'TA'->0)::numeric)
+sum((payload->'TA'->1)::numeric))/2 as var1
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDate},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDate},'YYYY-MM-DD')
group by device_id ) as t1
JOIN (select device_id, (sum((payload->'TA'->0)::numeric)
+sum((payload->'TA'->1)::numeric))/2 as var2
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDatePast},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDatePast},'YYYY-MM-DD')
group by device_id
)as t2
ON t1.device_id = t2.device_id group by t1.device_id order by v asc LIMIT 1;
```

id:lowest_alarm_time_trend_machine_id

- **description**:This KPI measures for all machines the alarm time trend and takes the machine id with the minimum one

- **query**:

```
select t1.device_id as v, min(((t1.var1 - t2.var2)/t2.var2)*100) as v1
from( select device_id , (sum((payload->'TA'->0)::numeric)
+sum((payload->'TA'->1)::numeric))/2 as var1
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDate},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDate},'YYYY-MM-DD')
group by device_id ) as t1
JOIN (select device_id, (sum((payload->'TA'->0)::numeric)
+sum((payload->'TA'->1)::numeric))/2 as var2
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDatePast},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDatePast},'YYYY-MM-DD')
group by device_id)as t2
ON t1.device_id = t2.device_id group by t1.device_id order by v1 asc LIMIT 1;
```

id:shortest_alarm_time

- **description**:This KPI measures the total alarm time for each machine and takes the minimum

- **query**:

```
select case
when min(alarm_time_1) >= min(alarm_time_0) then max(alarm_time_0)
else max (alarm_time_1)
end as v
from
(select device_id, ((payload->'TA'->1)::numeric) as alarm_time_1,
((payload ->'TA'->0)::numeric) as alarm_time_0
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDate},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDate},'YYYY-MM-DD'))as sub_t;
```

id:shortest_alarm_time_machine_id

- **description**:This KPI measures the total alarm time for each machine and the machine id with the lowest value

- **query**:

```
select device_id as v
from
```

```
(SELECT device_id,
LEAST(MIN(((payload->'TA'->0)::numeric)), MIN(((payload->'TA'->1)::numeric)))
AS min_v FROM ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDate},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDate},'YYYY-MM-DD')
group by device_id order by min_v LIMIT 1)as sub_t;
```

id:tot_alarm_time

- **descriptiion**:This KPI measures the total alarm time of all machines throughout the period
- **query**:

```
select sum(wk_a + id_a)as v
from(select device_id,sum((payload->'TA'->1)::numeric) as wk_a,
sum((payload->'TA'->0)::numeric) as id_a
from ${dataTableName}
where ts_device >= TO_TIMESTAMP(${startDate},'YYYY-MM-DD')
AND ts_device <= TO_TIMESTAMP(${endDate},'YYYY-MM-DD')
group by device_id ) as sub_t;
```

2 Javascript Interface Module

The javascript interface created to be able to request values from the database is an importable module, requiring the dowload of pg, pg-promise via npm command:

```
$ npm install pg
$ npm install pg-promise
```

which are a collection of node.js modules for interfacing with your PostgreSQL database in a non-blocking way. The technical specification of the code is ECMAScrip 7 (ES7), because in the code there are the keywords ”async” and ”await” introduced in this version. the module is called ”interface.mjs”, for testing purposes of new KPIs added it is possible to call the interface also via terminal, only on the value concerned using the file ”test.js”, with the command

```
$ node test.js
```

to perform customised tests, the parameters must be changed in the test.js code.

The interface requires a number of parameters to function, three mandatory and three optional. The three mandatory are:

- **kpi** : identifier of the required kpi (taken from the list in the previous section)
- **startDate and endDate**: which are respectively the start and end date of the period whose performance are to be observed.

The three optional, which may be set to null if not needed, are:

- **cost**: indicates the cost in kWh of the energy, even if in the database the unit of measurement is Wh the interface will provide the conversion.
- **howManyPeriods**: this is a variable that is used for indicators that refer to the past period such as Trends. If passed as a null value even if required, it will be set to 1.

- **percentage**: this parameter has only one use for now and that is to be passed to the KPI *tot_machine_used_percentage_of_period*. If not passed, it will automatically be set to 50%.

In order to make calls to the PostgreSQL database, the interface utilizes the Promise construct, concealed by the ”syntactic sugar” of async functions and await keyword. Consequently, calls to the interface module in other programs must be enclosed within async functions and invoked using await.