

# Laboratory of Data Science

Monia Bennici, Gianmarco Pepi

January 18, 2021

# Overview

- 1 Build the data warehouse
- 2 Solve business question on data warehouse
- 3 Solve business question on datacube

# Build the datawarehouse

# Build the datawarehouse

## Step 1

We created the database schema shown in Figure 1 using SQL Server Management Studio, setting the right type of every attribute and the relations between tables through primary and foreign keys.

# Build the datawarehouse

## Step 1

We created the database schema shown in Figure 1 using SQL Server Management Studio, setting the right type of every attribute and the relations between tables through primary and foreign keys.

## Step 2

We created three new files starting from one, to distinguish ram sales, cpu sales and gpu sales, using only basic python functionalities.

# Build the datawarehouse

## Step 1

We created the database schema shown in Figure 1 using SQL Server Management Studio, setting the right type of every attribute and the relations between tables through primary and foreign keys.

## Step 2

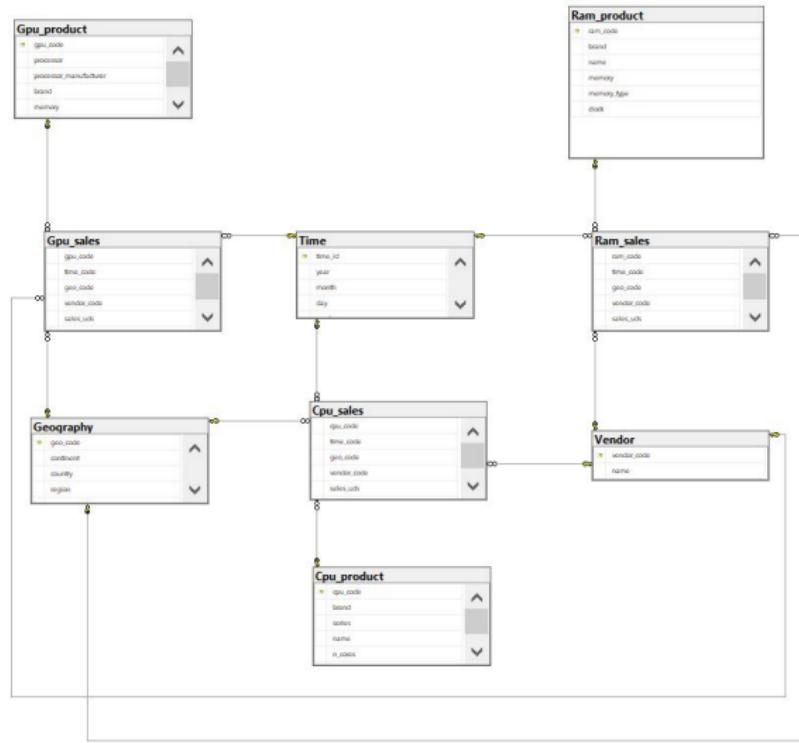
We created three new files starting from one, to distinguish ram sales, cpu sales and gpu sales, using only basic python functionalities.

## Step 3

We had to populate the data warehouse Group19HWMART with the three fact tables and the remaining dimension tables, after modifying *time.csv file*.



# Create the database schema



## Create 3 fact tables

```
file = open("fact.csv", "r")
csv1 = csv.reader(file, delimiter = ",")  
ram_file = open("ram_sales.csv", "w", newline = '\n')
writer = csv.writer(ram_file, delimiter = ",")  
header = ['ram_code', 'time_code', 'geo_code',
          'vendor_code', 'sales_uds', 'sales_currency']
writer.writerow(header)
next(csv1, None)
for line in csv1:
    if line[3] != '':
        line[3] = int(float(line[3]))
        del line[0:3]
        writer.writerow(line)
ram_file.close()
file.close()
```

# Populate the data warehouse

- Add *day\_of\_week* attribute to *time.csv* using the python library `datetime`, in particular we used the function `date().weekday()`.
- Add *quarter* attribute to *time.csv* using the following formula:

$$q = \text{int} \left( \frac{\text{int}(month) - 1}{3} \right) + 1$$

- Populate the data warehouse with modified tables and try to boost the performance of python inserting program *insert\_fast.py*.

# Solve business question on data warehouse

# Solve business question on data warehouse

## Question 1

For every region of Germany, the brand of ram ordered by sales

# Solve business question on data warehouse

## Question 1

For every region of Germany, the brand of ram ordered by sales

## Question 2

For every brand of gpu, calculate the ratio between sales during weekdays and sales during the weekend.

# Solve business question on data warehouse

## Question 1

For every region of Germany, the brand of ram ordered by sales

## Question 2

For every brand of gpu, calculate the ratio between sales during weekdays and sales during the weekend.

## Question 3

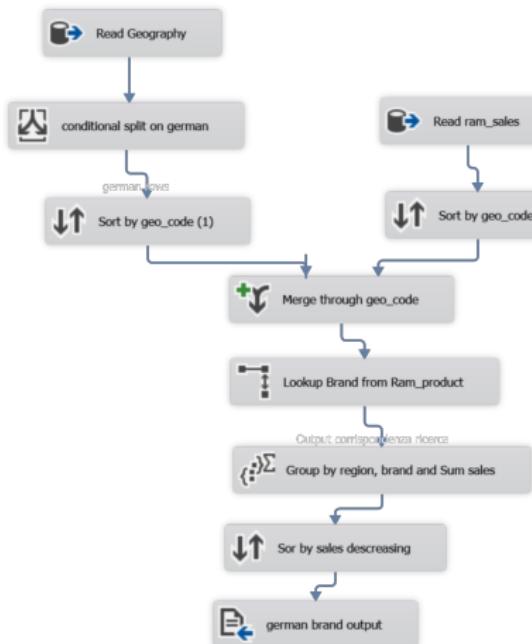
Calculate which type of product, cpu, gpu or ram, yields the most sales for each continent.

# SSIS 1

For every region of Germany, the brand of ram ordered by sales

# SSIS 1

For every region of Germany, the brand of ram ordered by sales

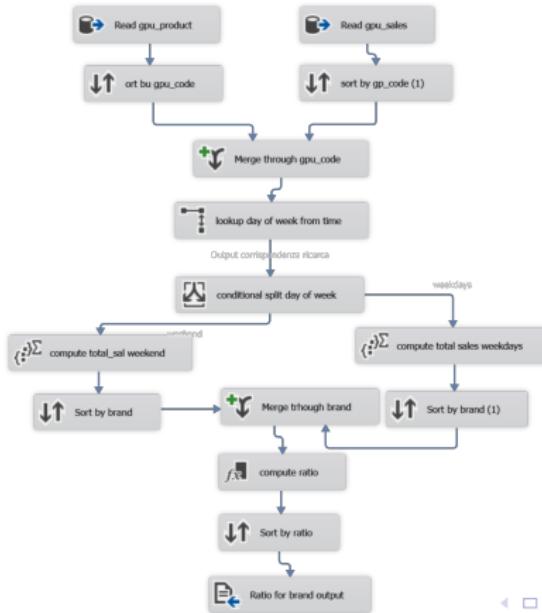


## SSIS 2

For every brand of gpu, calculate the ratio between sales during weekdays and sales during the weekend.

# SSIS 2

For every brand of gpu, calculate the ratio between sales during weekdays and sales during the weekend.

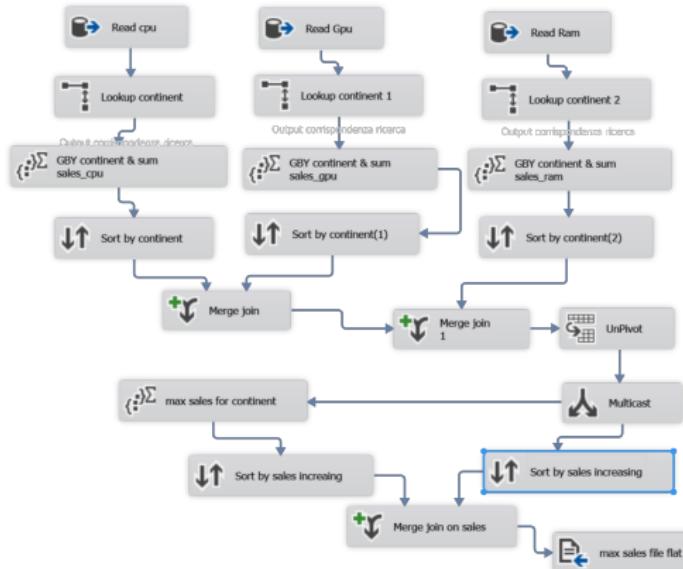


## SSIS 3

Calculate which type of product, cpu, gpu or ram, yields the most sales for each continent.

# SSIS 3

Calculate which type of product, cpu, gpu or ram, yields the most sales for each continent.



# Solve business question on datacube

# Solve business question on datacube

## Step 1

We set data source *Group19HWMART*, then we extracted a datamart shown in figure 2 as a data source view. At the end we set the dimensions choosing the measures, we selected the useful attributes and set the right hierarchies.

# Solve business question on datacube

## Step 1

We set data source *Group19HWMART*, then we extracted a datamart shown in figure 2 as a data source view. At the end we set the dimensions choosing the measures, we selected the useful attributes and set the right hierarchies.

## Step 2

We solved three business questions on multidimensional cube through MDX query.

# Solve business question on datacube

## Step 1

We set data source *Group19HWMART*, then we extracted a datamart shown in figure 2 as a data source view. At the end we set the dimensions choosing the measures, we selected the useful attributes and set the right hierarchies.

## Step 2

We solved three business questions on multidimensional cube through MDX query.

## Step 3

We responded to requests producing some dossier using two different software: PowerBI Desktop and MicroStrategy Desktop.

# Create a datacube

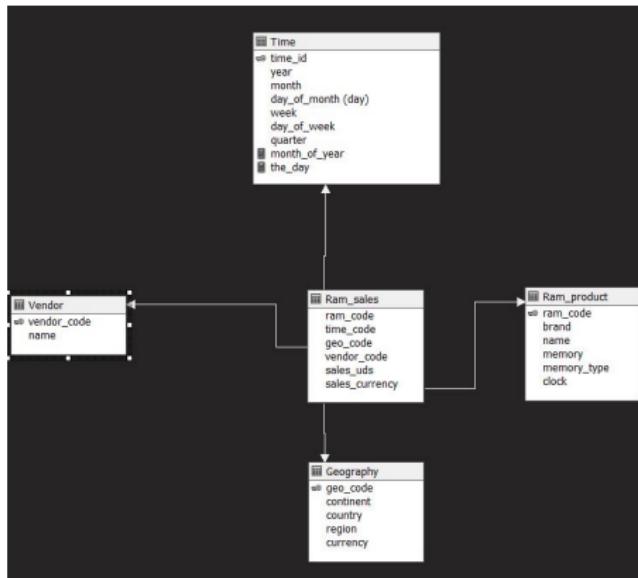
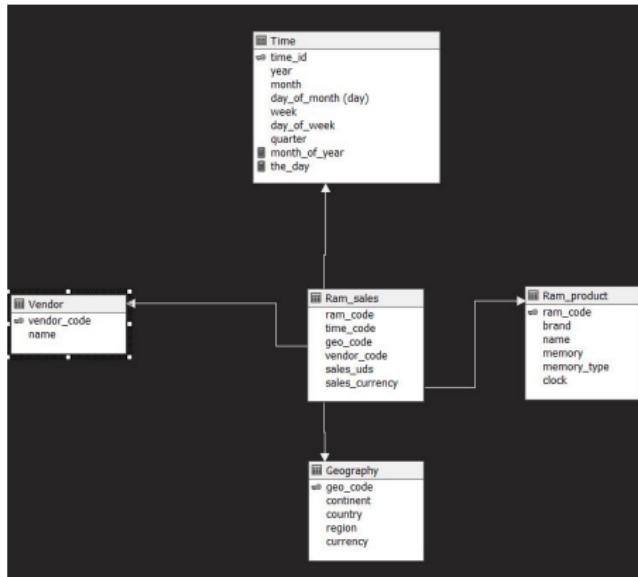


Figure: Extracted Data Mart

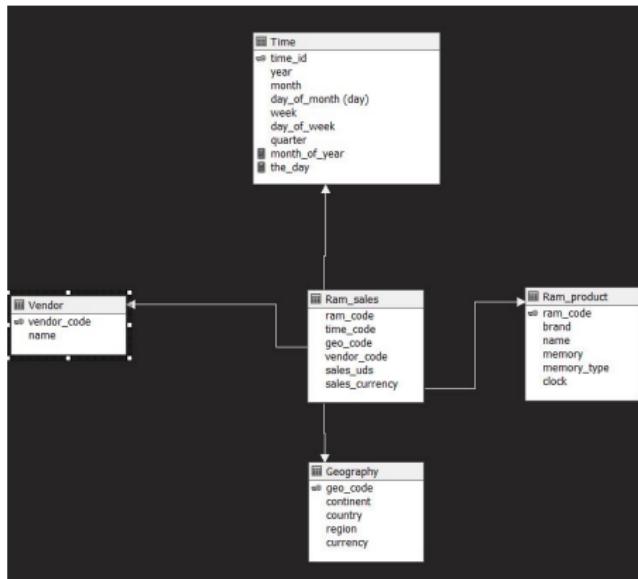
# Create a datacube



- **Data Source:**  
Group19HW MART

Figure: Extracted Data Mart

# Create a datacube



- **Data Source:**  
Group19HW MART
- **Data Source view:**  
DataMart shown in figure.

Figure: Extracted Data Mart

# Create a datacube

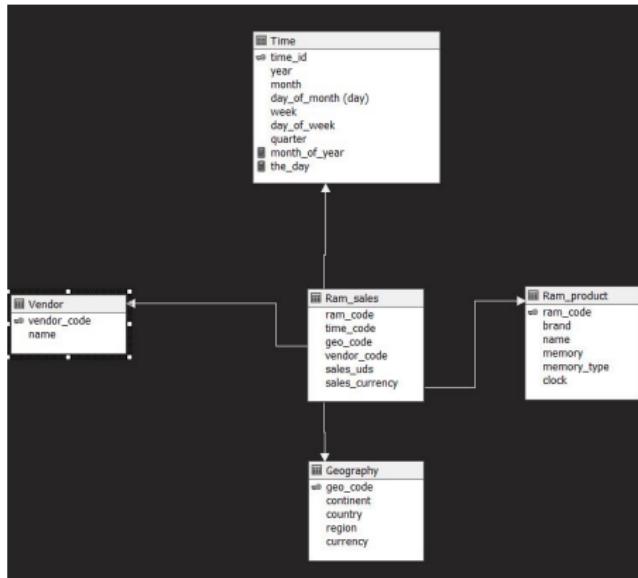


Figure: Extracted Data Mart

- **Data Source:**  
Group19HW MART
- **Data Source view:**  
DataMart shown in figure.
- **Measures:** Ram Sales and NVendor.

# Create a datacube

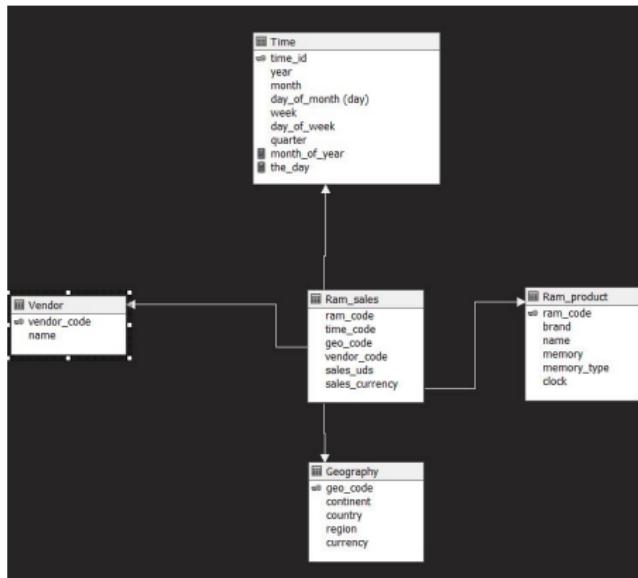


Figure: Extracted Data Mart

- **Data Source:** Group19HW MART
- **Data Source view:** DataMart shown in figure.
- **Measures:** Ram Sales and NVendor.
- **Hierarchies:** DayMonthYear, DayWeekYear and Gerarchia

# Business question using MDX

Show the percentage increase in total sales with respect to the previous month for each ram brand and each country.

## Business question using MDX

Show the percentage increase in total sales with respect to the previous month for each ram brand and each country.

```
with member incremento as ([Time].[DayMonthYear].currentmember,[Measures].[Sales Uds])
 -([Time].[DayMonthYear].currentmember.lag(1), [Measures].[Sales Uds])
 member perc as
 incremento/[Measures].[Sales Uds],
 format_string="percent"
 select {[Measures].[Sales Uds], incremento, perc} on columns,
 nonempty(([Ram Product].[Brand].[Brand],[Geography].[Gerarchia].[Country],
 [Time].[Year].[Year], [Time].[DayMonthYear].[Month Of Year])) on rows
 from [Group19HW Mart];
```

# Business question using MDX

For each region and ram brand show the total sales in percentage with respect to the total sales of the corresponding country.

## Business question using MDX

For each region and ram brand show the total sales in percentage with respect to the total sales of the corresponding country.

```
WITH MEMBER tot_country AS  
([Geography].[Gerarchia].currentmember.parent,[Measures].[Sales Uds])  
MEMBER perc_sales AS  
[Measures].[Sales Uds]/tot_country,  
FORMAT_STRING = "Percent"  
SELECT perc_sales ON COLUMNS,  
NONEMPTY(([Geography].[Gerarchia].[Region],[Ram Product].[Brand].[Brand]))  
FROM [Group19HW Mart];
```

# Business question using MDX

Show the ram memory types having a total sales greater than 10% of the totals sales in each continent by continent and year.

## Business question using MDX

Show the ram memory types having a total sales greater than 10% of the totals sales in each continent by continent and year.

```
WITH MEMBER cont_sales AS  
([Ram Product].[Memory Type].currentmember.parent,[Measures].[Sales Uds])  
MEMBER perc_var AS  
[Measures].[Sales Uds]/cont_sales,  
FORMAT_STRING = "Percent"  
SELECT {cont_sales,[Measures].[Sales Uds],perc_var} ON COLUMNS,  
nonempty(FILTER(([Geography].[Gerarchia].[Continent],[Time].[Year].[Year],  
[Ram Product].[Memory Type].[Memory Type]),  
perc_var > 0.1))ON ROWS  
FROM [Group19HW Mart];
```

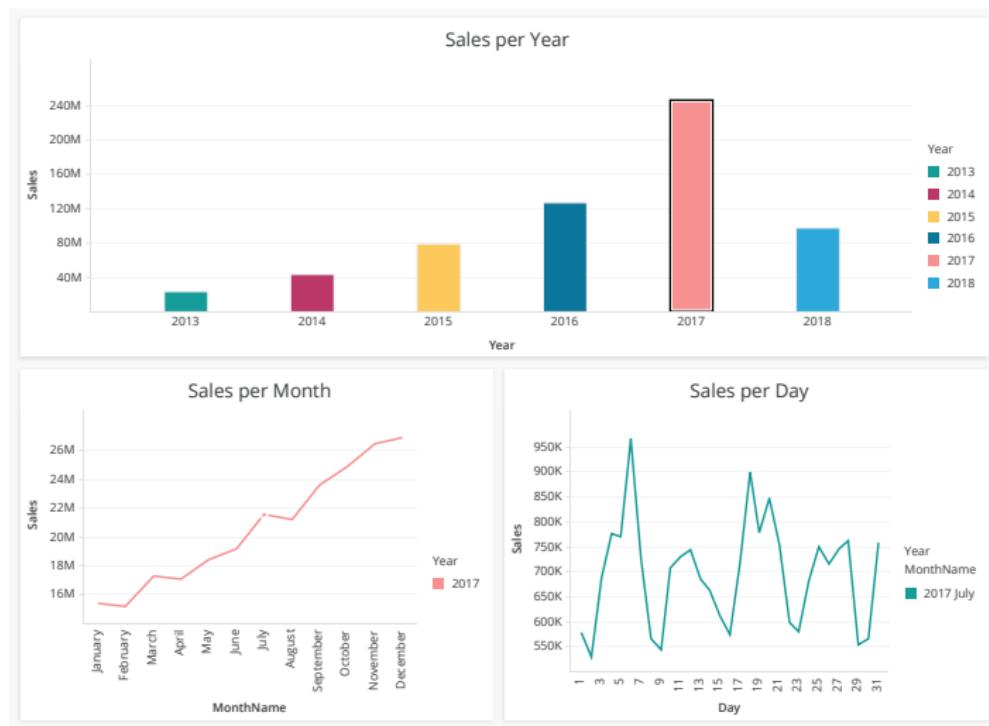
# Different reports

- Create a dashboard that shows how sales change over time, giving the user the opportunity to see the sales behavior for different time granularity
- Show the geographical distribution of sales and of the number of products purchased.
- Create a plot/dashboard of your choosing, that you deem interesting w.r.t. the data available in your cube

# Dashboard 1

dossier\_g19

Capitolo 1 - Sales per YearMonthDay - Riepilogo

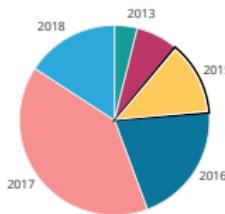


# Dashboard 1

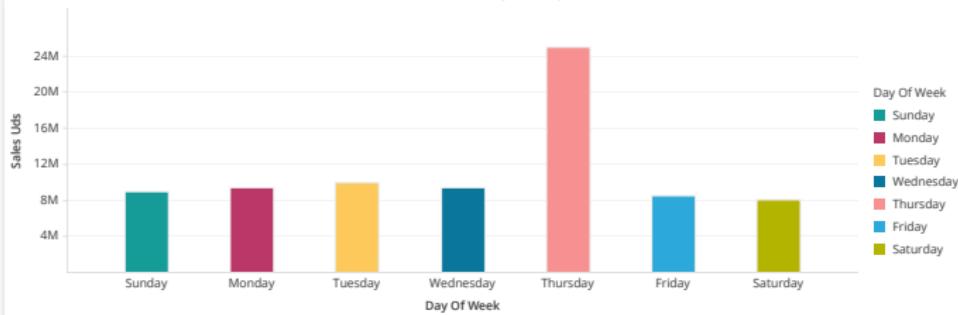
dossier\_g19

Capitolo 1 - Sales per day of week - Riepilogo

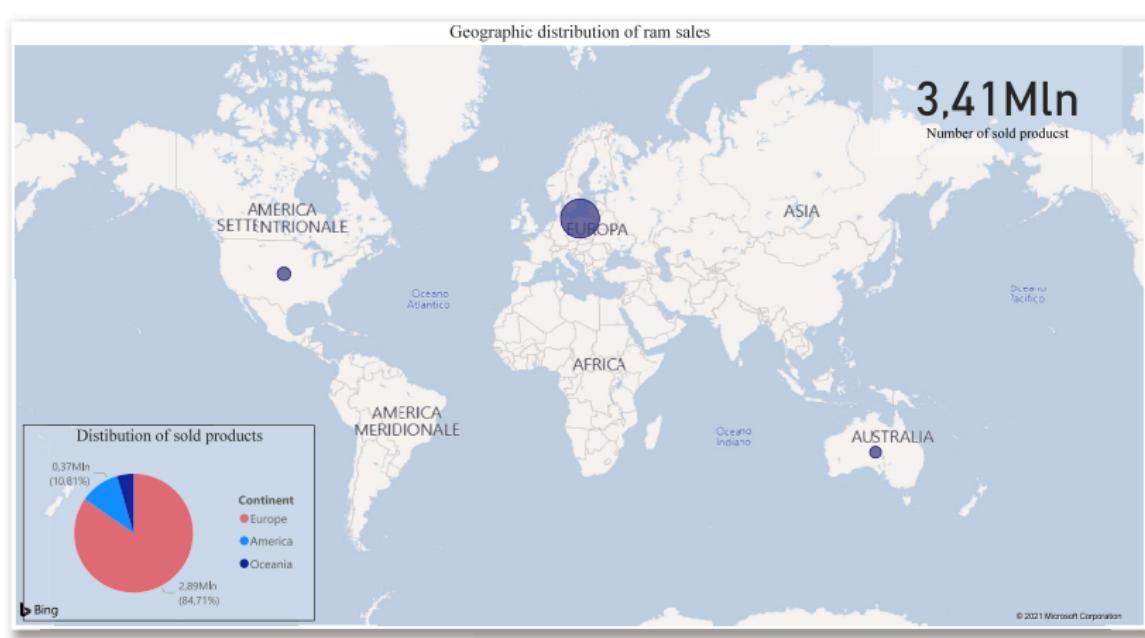
Sales per Year



Bar Chart of Sales per day of Week



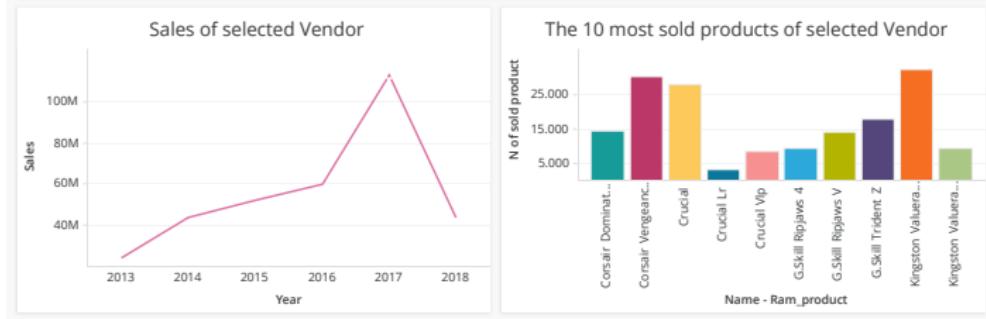
# Dashboard 2



# Dashboard 3

dossier\_g19

Capitolo 3 - Vendor Sales - Riepilogo



# Dashboard 3

