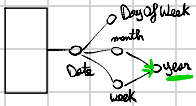
we need: date, day of the week, week, month, year

DIFFERENT ALTERNATIVES

! MONTH AND WEEK CANNOT BE CONNECTED.
EXAMPLE: WEEK THAT STARTS ON A MONTH AND ENDS ON ANOTHER

1)



in the Date table we will have:

D. Date = "9-12-2023" — 1 date

D. DayOfWeek = "Monday" ← (1 week in a year) 53 x # Years Date

D. Month = "December-2022" — 30 Date (30 days in a month is convention)

D. week = "48-2023" — 7 Date

D. Year = "2023" — 365 Date

1:N with date because a date corresponds to a single day of the week

1:N because a "specific month-year" belongs to a single year, but an year have different months

1:N "

2)

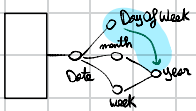
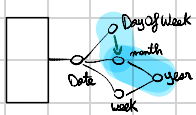


table is the same, except for

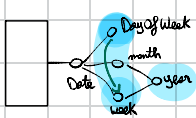
D. DayOfWeek = "Monday-2022" — 53 Date

3)



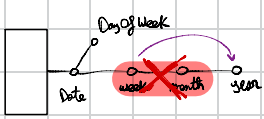
D. DayOfWeek = "Monday-November-22" — 4/5 Date

4)



D. DayOfWeek = "Monday-48-2023" is basically a date => redundant

5)



NO a one to many relationship because a week can be in the middle of two months

week – year is fine because for commercial purposes the first day of an year starts a new week

2022: 30 (Monday), 31 (Tuesday) ↓ week 53 - 2022

2023: 1 (Wednesday), 2 (Thursday) ↓ week 1 - 2023

© GLOSSARY: HOW TO COMPUTE THE MEASURES OF THE FACT

NUMBEROFACCIDENTS

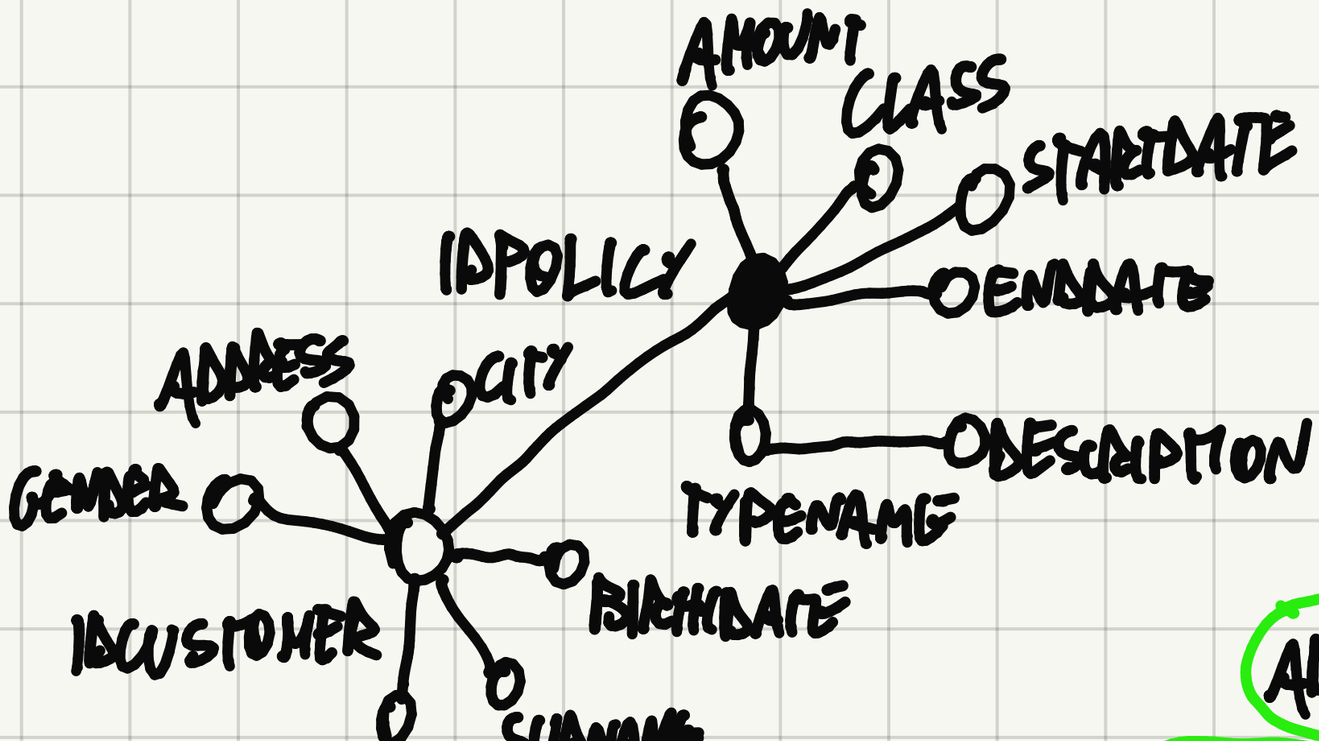
```
SELECT IDPOLICY, MOTIVATION, DATE, COUNT(*)
FROM ACCIDENT
GROUPBY IDPOLICY, MOTIVATION, DATE
```

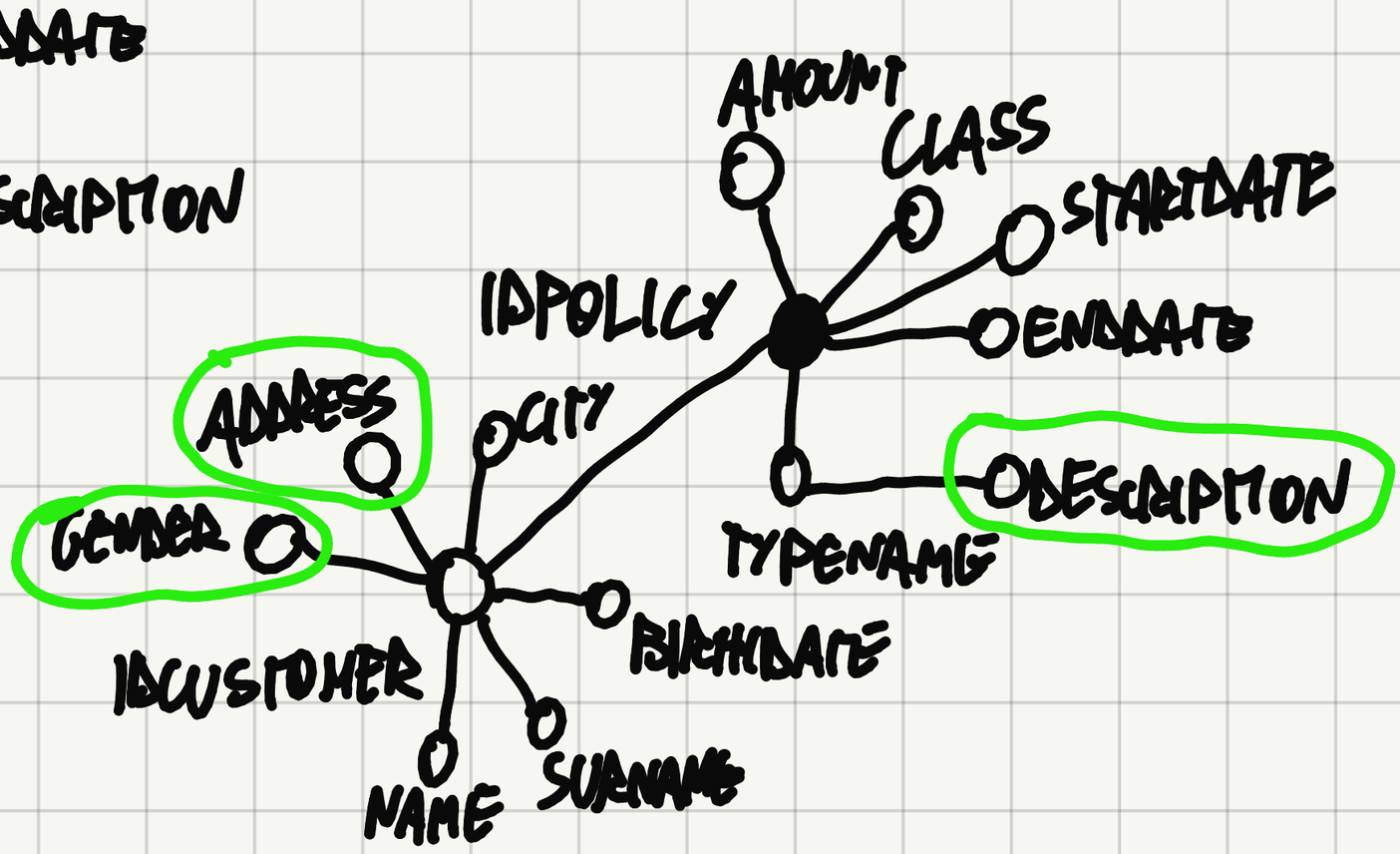
COST

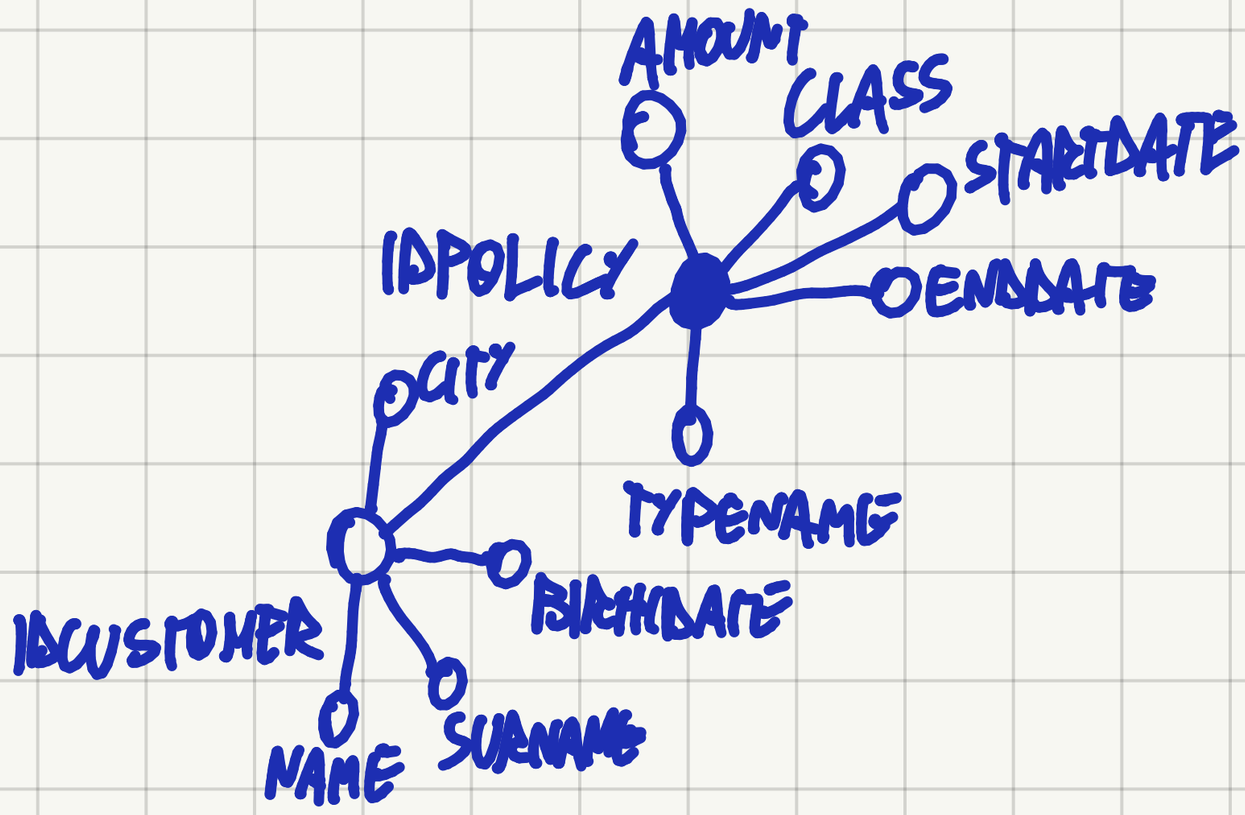
```
SELECT IDPOLICY, MOTIVATION, DATE, SUM(COST)
FROM ACCIDENT
GROUPBY IDPOLICY, MOTIVATION, DATE
```

POLICY

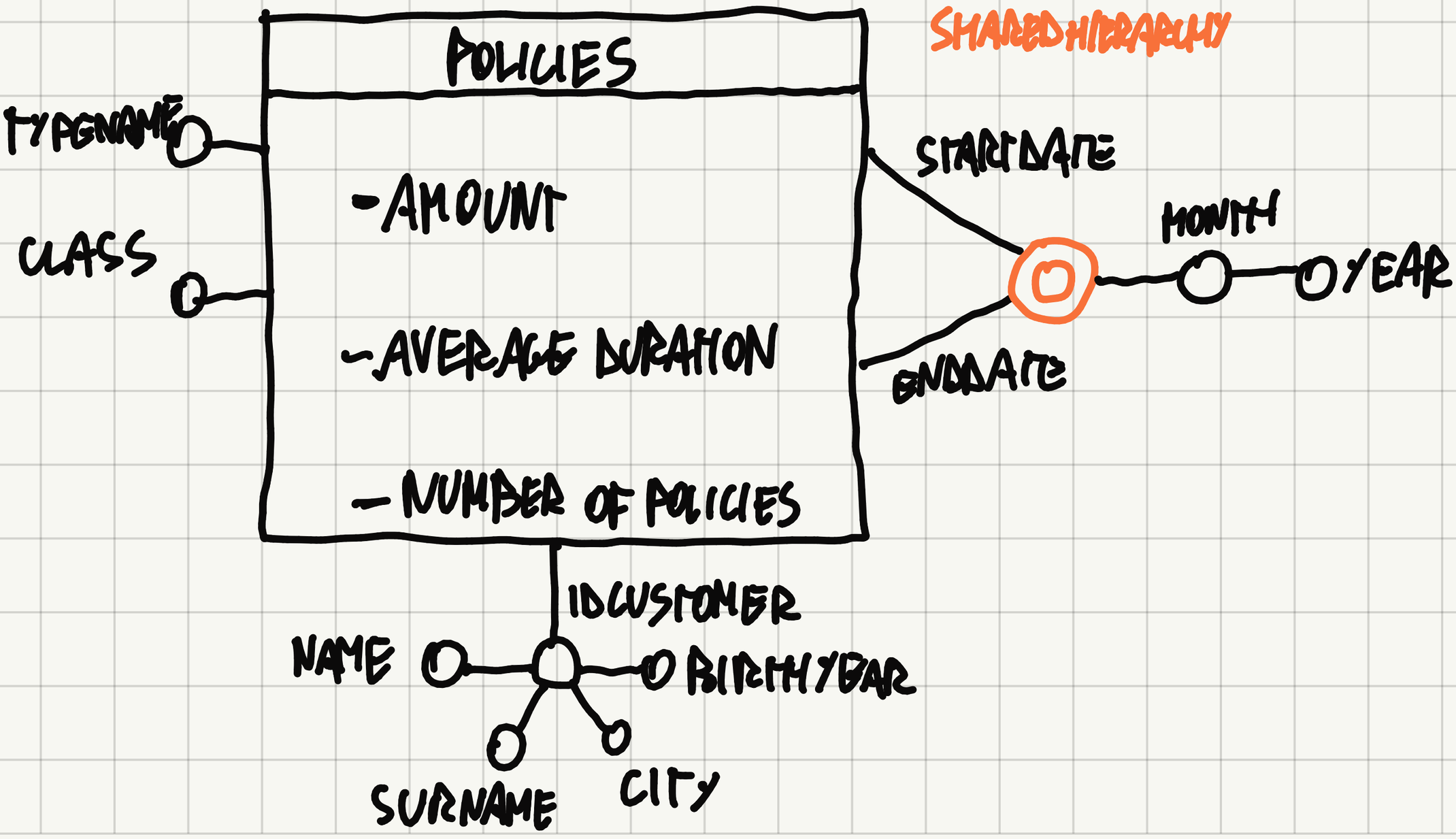
@ ATTRIBUTE TREE







ⓑ FACT SCHEMA



SHARED HIERARCHY

©GLOSSARY: *FOR THIS COURSE'S PURPOSE, WE CAN SIMPLIFY ASSUMING THAT THERE EXIST A FUNCTION THAT COMPUTE THE REQUESTED WEIGHTED AVERAGE

AMOUNT

```
SELECT  OWNER,TYPENAME,STARTDATE,ENDDATE,CLASS,SUM(AMOUNT)
FROM    POLICY
GROUPBY OWNER,TYPENAME,STARTDATE,ENDDATE,CLASS
```

AVERAGEDURATION

```
SELECT  OWNER,TYPENAME,STARTDATE,ENDDATE,CLASS,AVG(ENDDATE-
                                               -STARTDATE) *
FROM    POLICY

GROUPBY OWNER,TYPENAME,STARTDATE,ENDDATE,CLASS
```

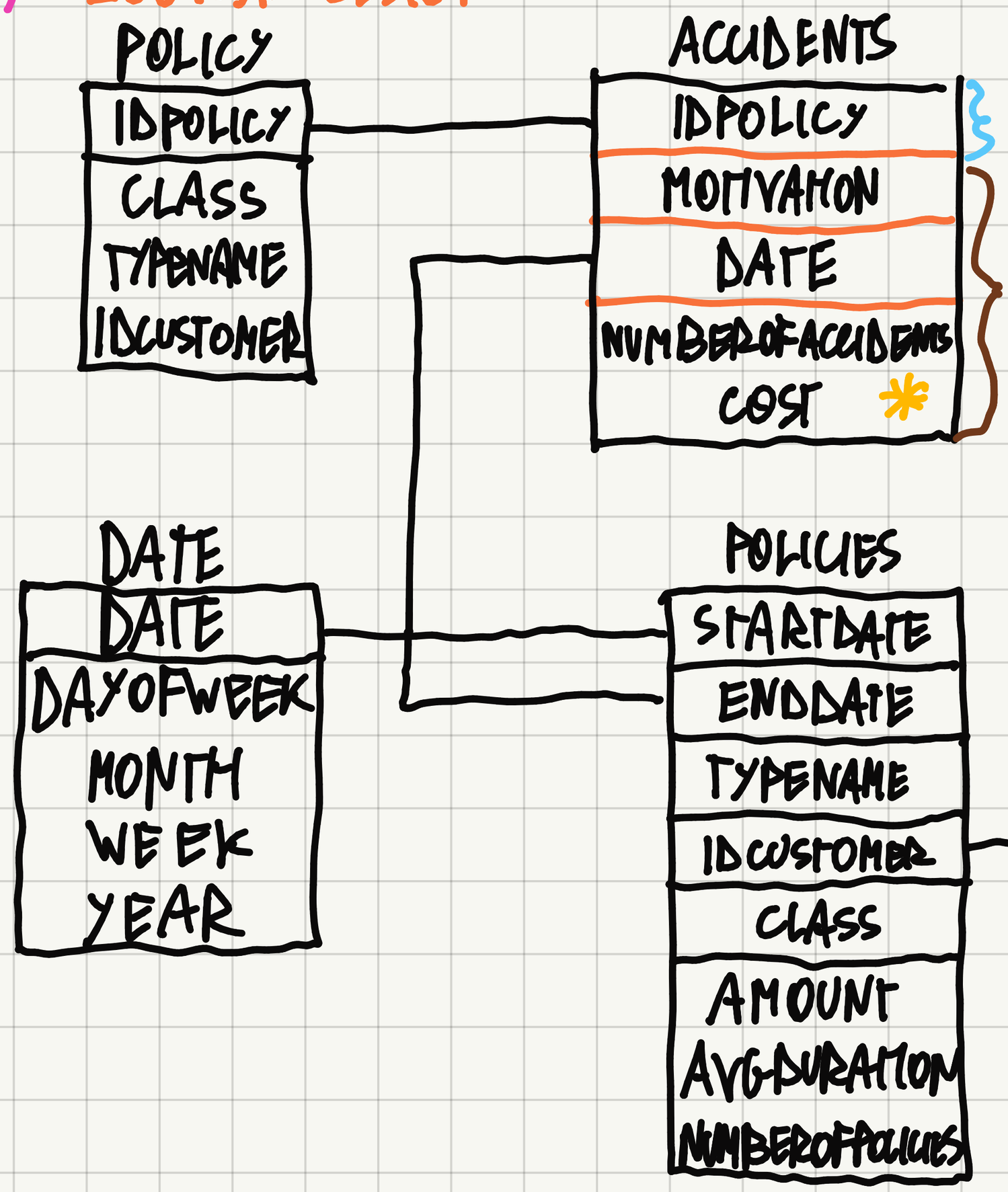
NUMBEROF POLICIES // SUPPORT MEASURE FOR THE WHEIGHTED AVERAGE

```
SELECT  OWNER,TYPENAME,STARTDATE,ENDDATE,CLASS,COUNT(*)
FROM    POLICY

GROUPBY OWNER,TYPENAME,STARTDATE,ENDDATE,CLASS
```

3) LOGICAL DESIGN

3) LOGICAL DESIGN



PRIMARY KEY ATTRIBUTES

ATTRIBUTES ARE SEPARATED

MEASURES ARE NOT SEPARATED

"UNKNOWN" IS VALID FOR OPTIONALS ATTRIBUTES

4) QUERIES

- USUALLY, WE MIGHT HAVE AT LEAST A FACT IN THE FROM
- WE BASE OUR QUERIES ON THE DATA WAREHOUSE'S TABLE (LOGICAL DESIGN)

a) SELECT P.IDPOLICY, D.WEEK, A.MOTIVATION, SUM(A.NUMBEROFACCIDENTS)

FROM ACCIDENTS AS A, POLICY AS P, DATE AS D

WHERE A.DATE=D.DATE AND A.IDPOLICY=P.IDPOLICY AND

D.DAYOFWEEK='SUNDAY-2015' AND P.CLASS='14'

AND P.TYPENAME='COMPREHENSIVE'

GROUPBY P.POLICY, D.WEEK, A.MOTIVATION

b) SELECT C.IDCUSTOMER, C.NAME, C.SURNAME, D.DATE, SUM(A.COST)
FROM ACCIDENTS AS A, POLICY AS P, CUSTOMER AS C, DATE AS D
WHERE A.IDPOLICY=P.IDPOLICY AND P.IDCUSTOMER=C.IDCUSTOMER
AND A.DATE=D.DATE AND D.MONTH='MARCH-2017' AND
C.BIRTHYEAR='1975' AND C.CITY='MILAN'
GROUPBY C.IDCUSTOMER, C.NAME, C.SURNAME, D.DATE

c) SELECT P.CLASS, P.TYPENAME, D.DATE, C.CITY, C.BIRTHYEAR, SUM(P.AMOUNT)
DATE AS D_1, DATE AS D_2
FROM POLICIES AS P, DATE AS D, CUSTOMER AS C
P.STARTDATE=D_1.DATE AND P.ENDDATE=D_2.DATE
WHERE P.STARTDATE=D.DATE AND P.IDCUSTOMER=C.IDCUSTOMER
AND D.MONTH='JULY-2015'
GROUPBY P.CLASS, P.TYPENAME, D.DATE, C.CITY, C.BIRTHYEAR

! REMEMBER: IN CASE OF MULTIPLE CONDITIONS ON DAY, WHE SHOULD HAVE DONE AS IN BROWN

d) SELECT C.IDCUSTOMER, C.NAME, C.SURNAME, D.DATE,
SUM(P.AVGDURATION * P.NUMBEROFPOLICIES)/SUM(P.NUMBEROFPOLICIES)
FROM POLICIES AS P, DATE AS D, CUSTOMER AS C
WHERE P.ENDDATE=D.DATE AND P.IDCUSTOMER=C.IDCUSTOMER AND
D.MONTH='FEBRUARY-2017'
GROUPBY C.IDCUSTOMER, C.NAME, C.SURNAME, D.DATE