POLITECNICO DI TORINO

Data Science and Database Technology

Exam 2023-02-21

Iniziato	martedì, 21 febbraio 2023, 14:12
Terminato	martedì, 21 febbraio 2023, 15:47
Tempo impiegato	1 ora 35 min.
Valutazione	**14,20** su un massimo di 32,00 (**44**%)

Domanda 1

Risposta corretta

Punteggio ottenuto 1,00 su 1,00

1 points (penalty 15% for a wrong answer)

The following sequence of operations is given within a log file:

B(T0) CK(T0) CK(T0) B(T1) I1(o4) Commit(T1) Commit(T0) B(T2) I2(o0) B(T3) B(T4) U2(o3) U4(o4) I2 (o3) I2(o2) FAILURE

Notation:

- Tn: Id of transaction n
- B(Tn): Begin of transaction Tn
- CK(Ta,Tb,...): checkpoint of transactions Ta, Tb, ...
- Commit(Tn): commit of transaction Tn
- Abort(Tn): abort (rollback) of transaction Tn
- Un(ox): update performed by transaction Tn on object ox
- In(ox): insert performed by transaction Tn on object ox
- Dn(ox): delete performed by transaction Tn on object ox

Which operations are performed for a warm restart?

- (a) None of the other answers is correct.

- (b) UNDO T0,T1 REDO T2,T3,T4
- (c) UNDO T1,T2,T3,T4 REDO T0
- (d) UNDO T3,T4 REDO T0,T1,T2
- (e) UNDO T0,T3,T4 REDO T1,T2
- (f) UNDO T2,T3,T4 REDO T0,T1 ✓
- (g) UNDO T0,T1,T2 REDO T3,T4

Risposta corretta.

La risposta corretta è: UNDO T2,T3,T4 REDO T0,T1

Domanda 2

Completo

Punteggio ottenuto 3,25 su 5,00

5 points (no penalty for a wrong answer)

The following data warehouse describes the trend over time of REIMBURSEMENT CLAIMS received from different stores downstream of insurance coverages for electronic device models. Specifically, the analysis should be done according to the electronic device model and its category, list of insurance coverages subscribed (Legal Warranty, 3-Year Extension, AccidentalDamage, Theft). It is possible to subscribe multiple insurance covers for the same device model (this list represents a configuration and each attribute assumes Y/N value). The data warehouse stores the gender, residence, and age range of the buyer. For each store handling claims, city, region, and corresponding geographic area are known. The metrics to analyze are the number of claims received, the number of claims completed, the total amount claimed, the total amount approved, and the duration of the claims processing expressed in days (DurationTotProcess). Metrics should be analyzed for each date, month, 2-month, 3-month, 6-month, year.

The data warehouse is characterized by the following conceptual scheme and the corresponding logical scheme.

STORE(**IDStore**, Store, City, Region, GeoArea)
DEVICE-MODEL (**IDDeviceModel**, DeviceModel, Category)
JUNK-INSURANCE_COVERAGES (**IDJIC**, LegalWarranty, 3-YearExtension, AccidentalDamage, Theft)
JUNK-BUYER-FEATURES (**IDJBF**, Gender, Residence, AgeRange)
TIME(**IDTime**, Date, Month, 2-Months, 3-Months, 6-Months, Year)
REIMBURSEMENT-CLAIMS (**IDStore**, **IDDeviceModel**, **IDJIC**, **IDJBF**, **IDTime**, #ClaimsReceived, #ClaimsCompleted, AmountTotClaimed, AmountTotApproved, DurationTotProcessInDays)

Given the above logic schema, consider the following queries of interest:

a. Considering the stores located in the northern geographic area, separately by store region and year, show the total number of completed claims, the total approved amount, and the average monthly duration of claims processing.

b. Considering insurance coverages that include accidental damage (attribute AccidentalDamage), but not theft (attribute Theft), separately by month and store geographic area, show the total number of completed claims and the difference between the total claimed and the total approved amount.

c. Considering the years 2021 and 2022, separately by semester (6-Months attribute) and store region, show the total number of completed claims and the corresponding average approved amount.

Given the above logical scheme, answer the following requests:

1. Define a materialized view with CREATE MATERIALIZED VIEW, so as to reduce the

response time of the queries of interest (a) to (c) above. Specifically, specify the SQL query associated with Block A in the following statement:

```
CREATE MATERIALIZED VIEW ViewRembursements
BUILD IMMEDIATE
REFRESH FAST ON COMMIT
AS
Block A
```

2. Define the **minimal set** of attributes that allows identification of the tuples belonging to the materialized ViewRembursements view.

3. Assume that the management of the materialized view (derived table) is carried out by means of triggers. Write the trigger to propagate to the ViewRembursements materialized view the changes due to the insertion of a new record into the REIMBURSEMENT-CLAIMS table.

1.
select year, month , 6-months , region , geoareA ,accidentaldamage , sum (claimscompleted) as totc , sum (claims
recieved) as totr , sum (amounttotapproved) as totapp , sum (durationtotprocessindays) as totdays
from reimbursment-cliam rc, time t, store s, junk-insurance-coverages j
where rc.idtime=t.idtime and rc.idstore=s.idstore and rc.idjic=j.idjic
group by year , month , 6-months , region , geoarea , accidentaldamage
---
2. minimal set: month , region , accidentaldamage

3.
create trigger mytrigger
after insert on reimbursment-cliams
for each row
declare
vary , var6 , varm , date ;
varregion , vararea , varacc , varchar(20);
n integer;
begin
select year , month, 6-month into vary , varm , var6
from time
where idtime=new.idtime

select region , geoarea into varregion , vararea
from store
where idstore=new.idstore

```
select accidentaldamage into varacc
from junk-insurance-cooverages
where idjic=new.isjic

select count(*) into n
from view rembursments
where month=varm and region=varregion and accidentaldamage= var acc

if (n>0) them
update view rembursment
set totc=totc+new.cliamscompleted ,
totr=totr+new.cliamsrecieved'
totapp=totapp++new.amounttotapproved '
totdays=totdays+new.durationtotprocessindays
where month=varm and region=varregion and accidentaldamage=varacc
else
insert on view rembursments () values(varm , varregion, varacc , new.cliamcompleted ,
new.cliamrecieved ,
new.amounttotapproved , new.durationtotprocessindays)
endif;
end;
```

Queries of interest:

(a) Select Region, Year, SUM(#ClaimsCompleted), SUM(AmountTotApproved), SUM(DurationTotProcessInDays)/COUNT(DISTINCT Month)
FROM STORE, TIME, REIMBURSEMENT-CLAIMS
WHERE join AND GeoArea = ‘North’
GROUP BY Region, Year

(b) Select Month, GeoArea, SUM(#ClaimsCompleted), SUM(AmountTotClaimed) - SUM(AmountTotApproved)
FROM STORE, TIME, JUNK-INSURANCE_COVERAGES, REIMBURSEMENT-CLAIMS
WHERE join AND AccidentalDamage = ‘Y’ AND Theft = ‘N’
GROUP BY Month, GeoArea

(c) Select 6-Months, Region, SUM(#ClaimsCompleted), SUM(AmountTotApproved)/SUM(#ClaimsCompleted)
FROM STORE, TIME, REIMBURSEMENT-CLAIMS
WHERE join AND (Year = 2021 OR Year = 2022)
GROUP BY 6-Months, Region

1. Block A – Query for materialized view

SELECT Month, 6-Months, Year, Region, GeoArea, AccidentalDamage, Theft, SUM(#ClaimsCompleted) AS ClaimsTot, SUM(AmountTotClaimed) AS AmountClaimed, SUM(AmountTotApproved) AS AmountApproved, SUM(DurationTotProcessInDays) AS DurationTot

FROM JUNK-INSURANCE_COVERAGES C, STORE S, TIME T, REIMBURSEMENT-CLAIMS R
WHERE C.IDJIC = R.IDJIC AND S.IDStore = R.IDStore AND T.IDTime = R.IDTime
GROUP BY Month, 6-Months, Year, Region, GeoArea, AccidentalDamage, Theft

2. Identifier
Month, Region, AccidentalDamage, Theft

3. Trigger

```
CREATE OR REPLACE TRIGGER MaintenanceViewRembursements
AFTER INSERT ON REIMBURSEMENT-CLAIMS
FOR EACH ROW

DECLARE
VarYear, VarSemester VarMonth DATE;
VarAccidentalDamage, VarTheft BOOLEAN;
VarRegion, VarGeoArea  VARCHAR(10);
N INTEGER;

BEGIN

   SELECT Month, 6-Months, Year INTO VarMonth, VarSemester, VarYear
   FROM TIME
   WHERE IDTime = :NEW.IDTime;

   SELECT AccidentalDamage, Theft INTO VarAccidentalDamage, VarTheft
   FROM JUNK-INSURANCE_COVERAGES
   WHERE IDJIC = :NEW.IDJIC;

   SELECT Region, GeoArea INTO VarRegion,  VarGeoArea
   FROM STORE
   WHERE IDStore = :NEW.IDStore;

   SELECT COUNT(*) INTO N
   FROM ViewRembursements
   WHERE Month = VarMonth AND AccidentalDamage = VarAccidentalDamage AND Theft =
VarTheft AND Region = VarRegion;

   IF N>0 THEN
      UPDATE ViewRembursements
      SET ClaimsTot = ClaimsTot + :NEW.#ClaimsCompleted, AmountClaimed = AmountClaimed
+ :NEW.AmountTotClaimed,
      AmountApproved = AmountApproved + :NEW.AmountTotApproved, DurationTot =
DurationTot + :NEW.DurationTotProcessInDays
      WHERE Month = VarMonth AND AccidentalDamage = VarAccidentalDamage AND Theft =
VarTheft AND Region = VarRegion;

   ELSE
      INSERT INTO ViewRembursements(...) VALUES (VarMonth, VarSemester, VarYear,
VarRegion, VarGeoArea, VarAccidentalDamage, VarTheft, :NEW.#ClaimsCompleted,
:NEW.AmountTotClaimed, :NEW.AmountTotApproved, :NEW.DurationTotProcessInDays);

   END IF;
```

END;

Commento:
1.

select year, month , 6-months , region , geoareA ,accidentaldamage , **MISSING Theft** sum (claimscompleted) as totc , ~~**sum (claims**~~

~~**recieved)**~~ **MISSING SUM(AmountTotClaimed)** as totr , sum (amounttotapproved) as totapp , sum (durationtotprocessindays) as totdays

from reimbursment-cliam rc, time t, store s, junk-insurance-coverages j

where rc.idtime=t.idtime and rc.idstore=s.idstore and rc.idjic=j.idjic

group by year , month , 6-months , region , geoarea , accidentaldamage **MISSING Theft**

---

2. minimal set: month , region , accidentaldamage **MISSING Theft**

3.

create trigger mytrigger

after insert on reimbursment-cliams

for each row

declare

vary , var6 , varm , date ;

varregion , vararea , varacc , varchar(20);

n integer;

begin

select year , month, 6-month into vary , varm , var6

from time

where idtime=new.idtime

select region , geoarea into varregion , vararea

from store

where idstore=new.idstore

select accidentaldamage **MISSING Theft** into varacc **MISSING varTheft**

from junk-insurance-cooverages

where idjic=new.isjic

select count(*) into n

from view rembursments

where month=varm and region=varregion and accidentaldamage= var acc **and condition on theft**

if (n>0) them

update view rembursment

set totc=totc+new.cliamscompleted ,

totr=totr+~~**new.cliamsrecieved'**~~ **:NEW.AmountTotClaimed**

totapp=totapp++new.amounttotapproved '

totdays=totdays+new.durationtotprocessindays

where month=varm and region=varregion and accidentaldamage=varacc **and condition on theft**

else

insert on view rembursments () values(varm , varregion, varacc , new.cliamcompleted , new.cliamrecieved ,

new.amounttotapproved , new.durationtotprocessindays)

endif;

end;

Domanda 3

Risposta non data

Punteggio max.: 1,50

1.5 points (penalty 15% for a wrong answer)

Note: In MongoDB it is possible to perform groupings ($group) on multiple attributes. To do this, it is sufficient to specify an object as the key (_id) in the form: { "key1": attribute1, "key2": attribute2, ... }.

In this case, a grouping will be performed on the attributes attribute1, attribute2, etc. Each document returned by the aggregation will have an _id value in the format:

```
"_id": {
    "key1": "attribute1_value1",
    "key2": "attribute2_value1",
...
}
```

A MongoDB collection is given that is used to record attendance for an online learning platform. Each attendance is recorded as a document within a "attendance" collection. The following is an example document extracted from the collection:

```
{
   "student": {
      "first": "MARGARET",
      "last": "MOORE"
   },
   "class_name": "Computer science",
   "lesson_number": 9,
   "duration": 120
}
```

It is desired to extract, for each course, the maximum number of participants in a single lesson. Which of the following queries satisfies the previous request?

Consider the pair (course name, lesson number) sufficient to uniquely identify any lesson held.

(a)

```
db.collection.aggregate([
  {
   "$group": {
    "_id": {
     "class": "$class_name",
    },
    "max_participants": {
     "$sum": {
      "$max": 1
     }
    }
   }
  }
])
```

(b)

```
db.collection.aggregate([
 {
  "$group": {
   "_id": "$_id",
   "max_participants": {
    "$max": "$participants"
   }
  },
 },
 {
  "$group": {
   "_id": {
    "class": "$class_name",
    "lesson": "$lesson_number"
   },
   "participants": {
    "$sum": 1
   },
  }
 }
])
```

(c)

```
db.collection.aggregate([
 {
  "$group": {
   "_id": {
    "class": "$class_name",
    "lesson": "$lesson_number"
   },
   "participants": {
    "$sum": 1
   },
  }
 },
 {
  "$group": {
   "_id": "$_id",
   "max_participants": {
    "$max": "$participants"
   }
  },
 }
])
```

(d)

```
db.collection.aggregate([
 {
  "$group": {
   "_id": {
    "class": "$class_name",
    "lesson": "$lesson_number"
   },
   "participants": {
    "$sum": 1
   },
  }
 },
 {
  "$group": {
   "_id": "$_id.class",
   "max_participants": {
    "$max": "$participants"
   }
  },
 }
])
```

(e)

```
db.collection.aggregate([
 {
  "$group": {
   "_id": {
    "class": "$class_name",
    "lesson": "$lesson_number"
   },
   "max_participants": {
    "$max": {
     "$sum": 1
    }
   }
  }
 }
])
```

(f)

```
db.collection.aggregate([
 {
  "$group": {
   "_id": {
    "class": "$class_name",
    "lesson": "$lesson_number"
   },
   "max_participants": {
    "$sum": {
     "$max": 1
    }
   }
  }
 }
])
```

- (g)

```
db.collection.aggregate([
 {
  "$group": {
   "_id": "$_id.class",
   "max_participants": {
    "$max": "$participants"
   }
  },
},
{
 "$group": {
  "_id": {
   "class": "$class_name",
   "lesson": "$lesson_number"
  },
  "participants": {
   "$sum": 1
  },
 }
])
```

- (h) None of the other answers allow to address the query of interest.
- (i)

```
db.collection.aggregate([
 {
  "$group": {
   "_id": {
    "class": "$class_name",
   },
   "max_participants": {
    "$max": {
     "$sum": 1
    }
   }
  }
 }
])
```

Risposta errata.

La risposta corretta è:

```
db.collection.aggregate([
 {
  "$group": {
   "_id": {
    "class": "$class_name",
    "lesson": "$lesson_number"
   },
   "participants": {
    "$sum": 1
   },
  }
},
{
  "$group": {
   "_id": "$_id.class",
   "max_participants": {
     "$max": "$participants"
    }
   },
  }
])
```

Domanda 4

Risposta errata

Punteggio ottenuto -0,15 su 1,00

1 points (penalty 15% for a wrong answer)

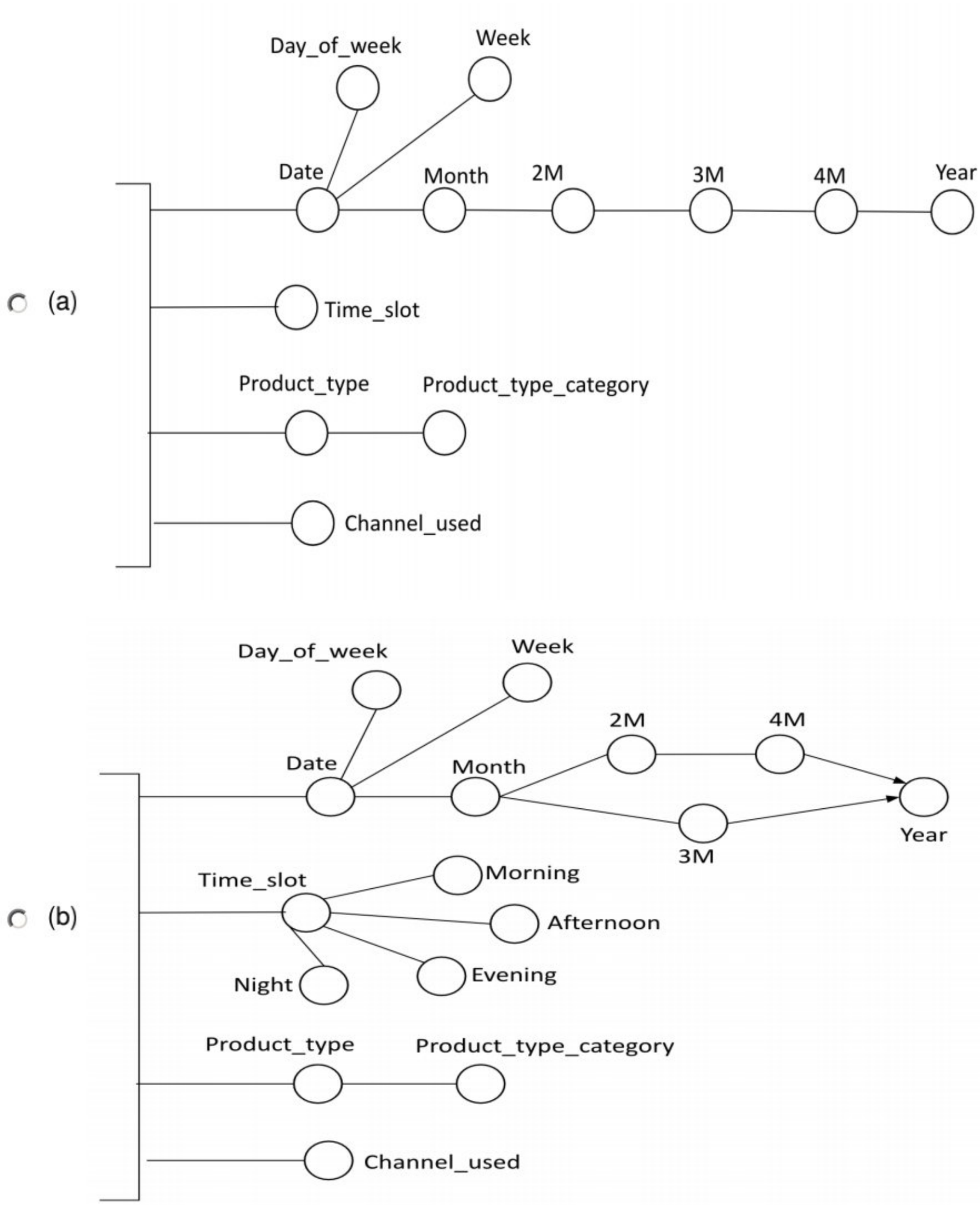
We want to analyze the information about to a chain of call centers that perform sales campaigns for different types of products supplied by a multinational company.

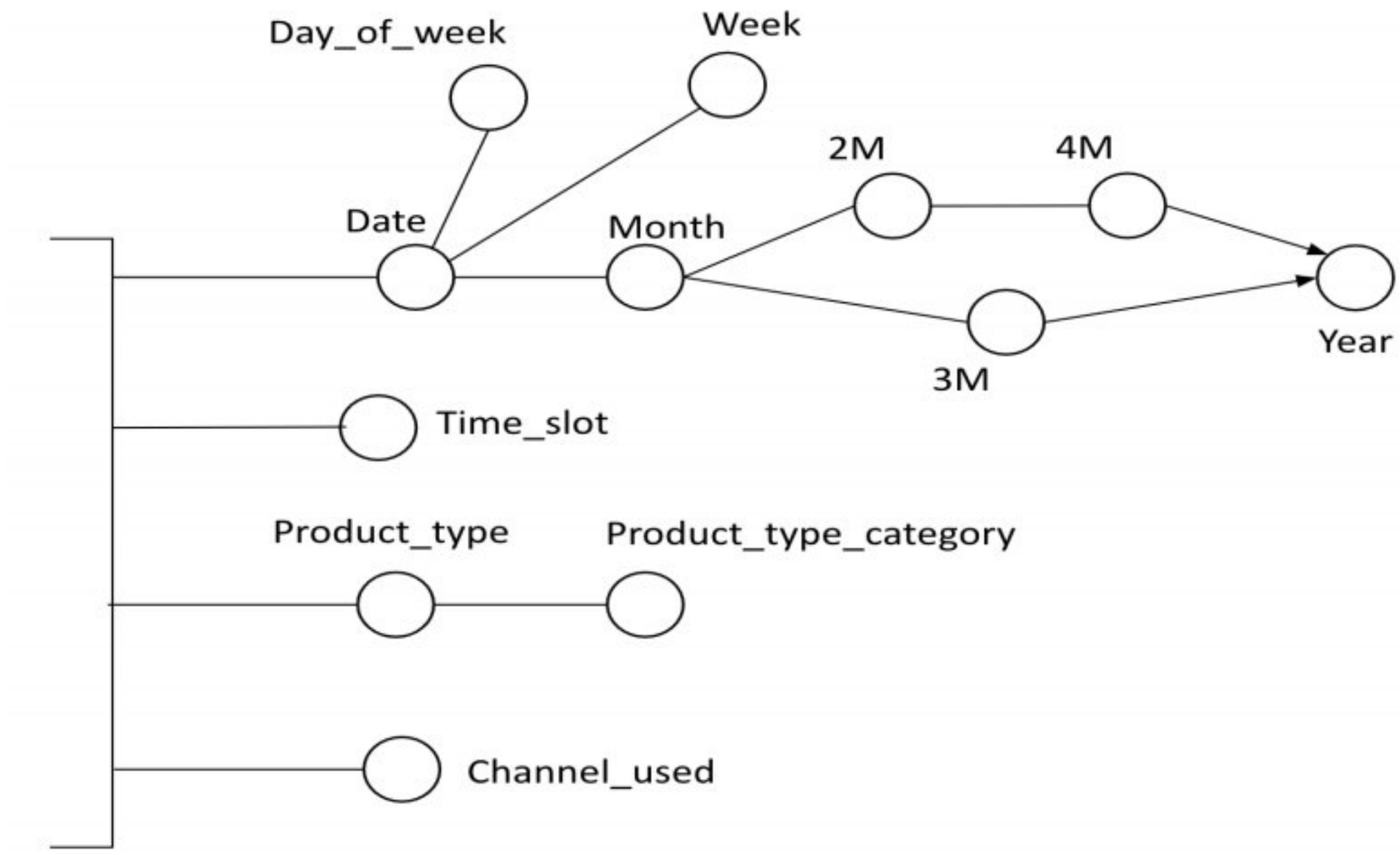
Each call center provides a multilingual and multichannel service for contacting potential customers and proposing products. The different communication channels used by each call center include, in addition to telephone calls, for example also emails and SMS messages. Each call center is also able to operate by supporting multiple languages (for example English, French and German). Therefore, each call center can contact customers in different countries around the world.

We want to analyze (1) the average number of contacted customers per product and (2) the average number of positive responses received per product, based on:

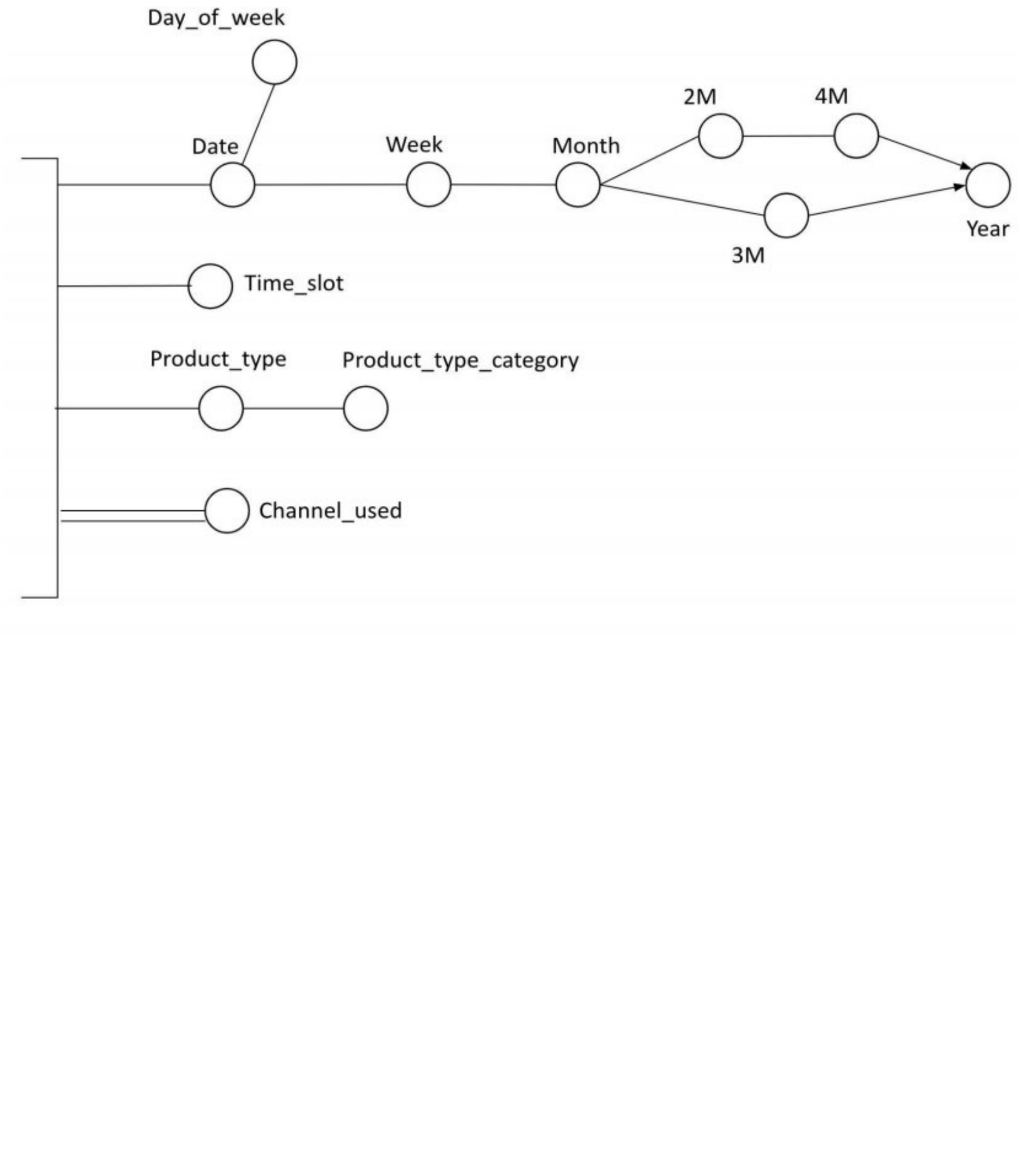
- call center, characterized by a name, its geographic location (expressed in terms of city, state, and continent where the call center is located), the size of the call center (expressed as small, medium, or large, based on the number of employees) and the indication of the languages supported by the call center (one or more languages among English, French, Italian, German and Spanish)
- temporal information on when the call center contacted the customer, expressed in terms of date, day of the week, week, month, 2-months, 3-months, 4-months, years and timeslot (a value among morning, afternoon, evening, and night)
- type of product proposed to the contacted customer
- categorization of the product type into basic, medium or luxury
- type of channel used to contact the customer (a value among telephone call, email, and SMS message)
- state and continent of the contacted customer
- characteristics of the contacted customer, in terms of gender, age range of the customer (a value among less than 20 years, between 20 and 40 years, between 40 and 60 years, and more than 60 years) and type of occupation of the customer (for example employee or freelancer or retired)

Select, from the dimensions proposed below, those that meet the requirements described in the problem specification.


Day_of_week
Week
Date
Month
2M
3M
4M
Year
Time_slot
Product_type
Product_type_category
Channel_used
(a)
Day_of_week
Week
Date
Month
2M
4M
3M
Year
Time_slot
Morning
Afternoon
Night
Evening
Product_type
Product_type_category
Channel_used
(b)

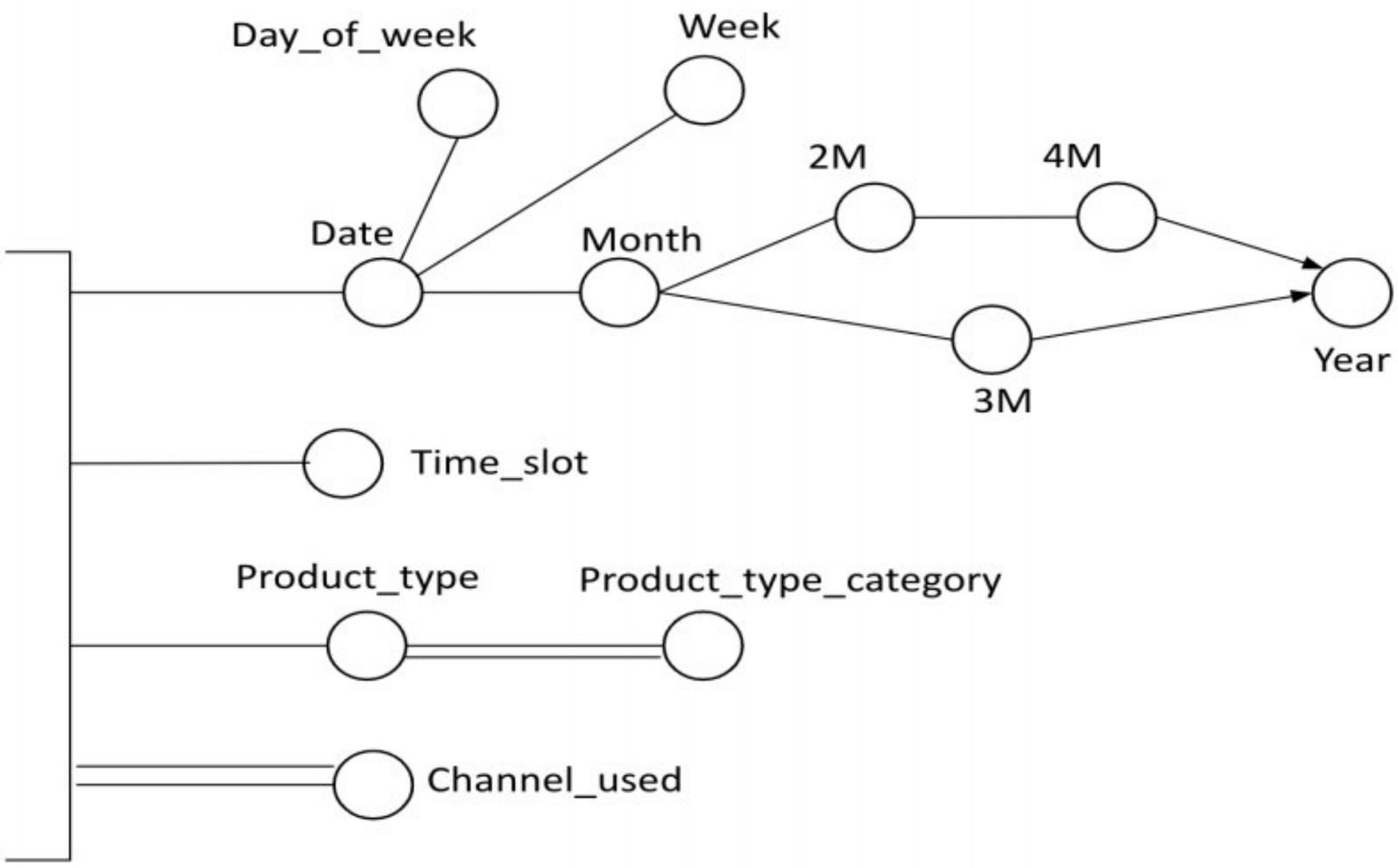

Day_of_week
Week
Date
Month
2M
4M
3M
Year
Time_slot
Product_type
Product_type_category
Channel_used

(c)

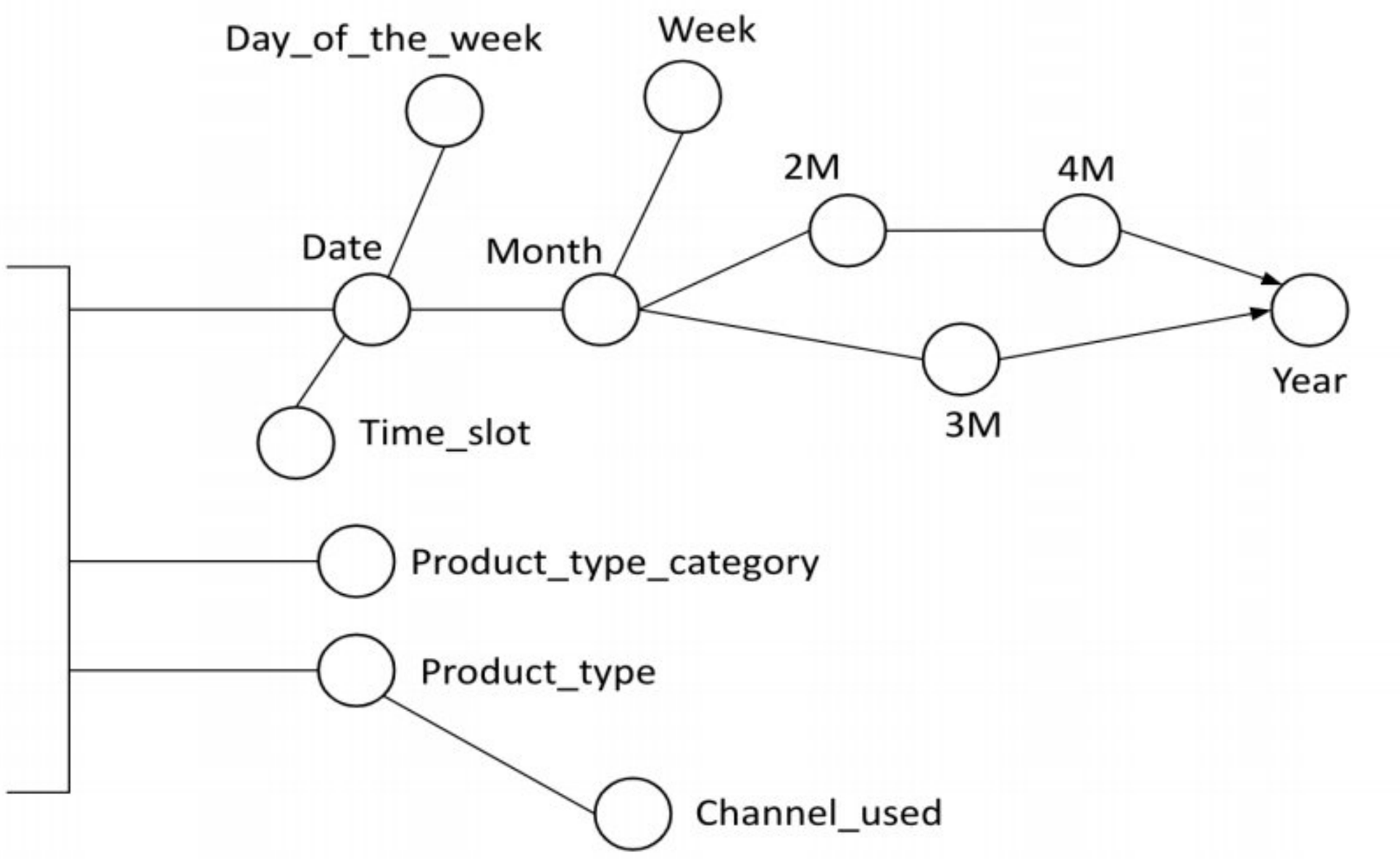

Day_of_week
Date
Week
Month
2M
4M
3M
Year
Time_slot
Product_type
Product_type_category
Channel_used

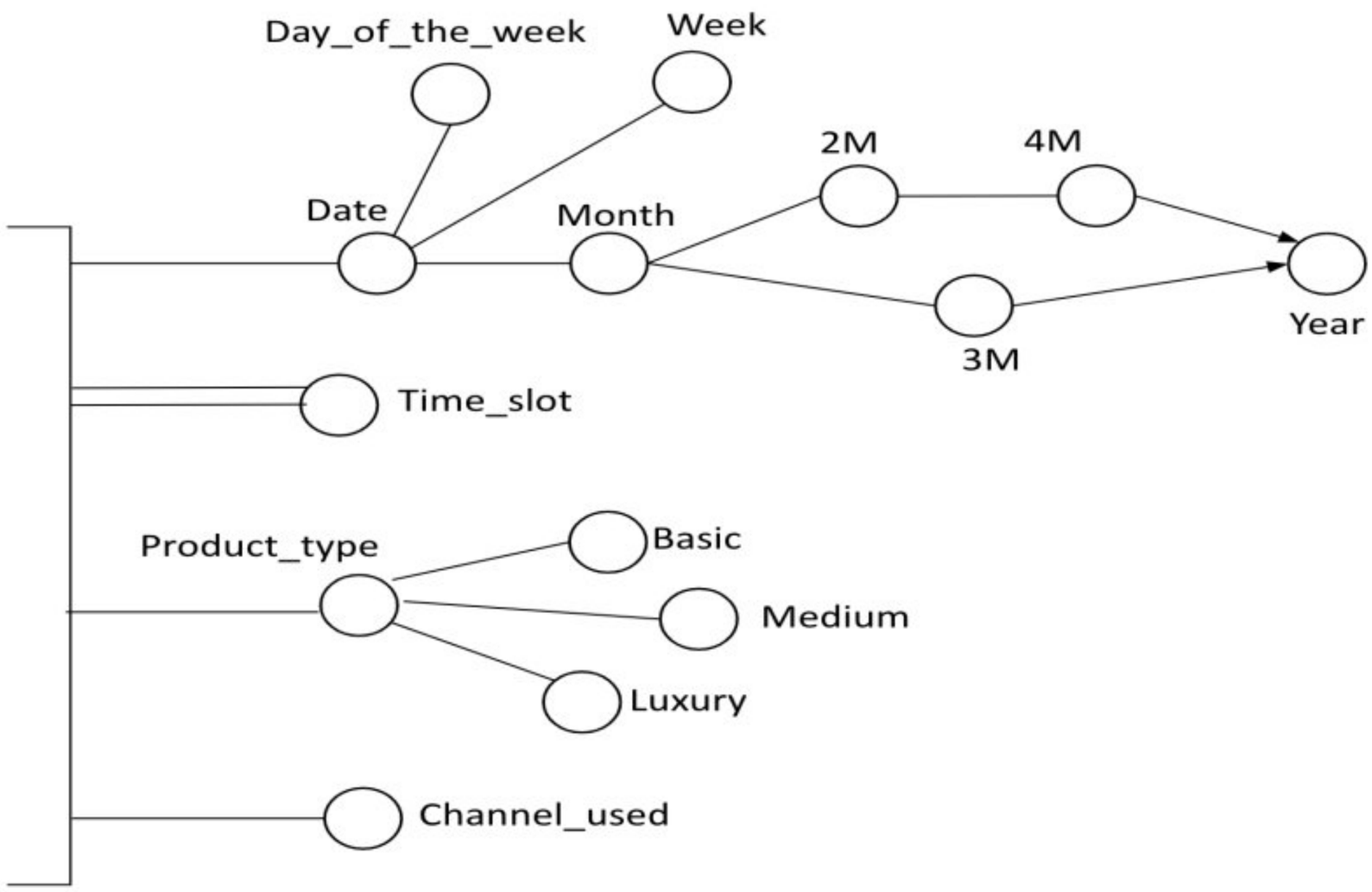
(d)

(e)


Day_of_week
Week
Date
Month
2M
4M
3M
Year
Time_slot
Product_type
Product_type_category
Channel_used

(f)

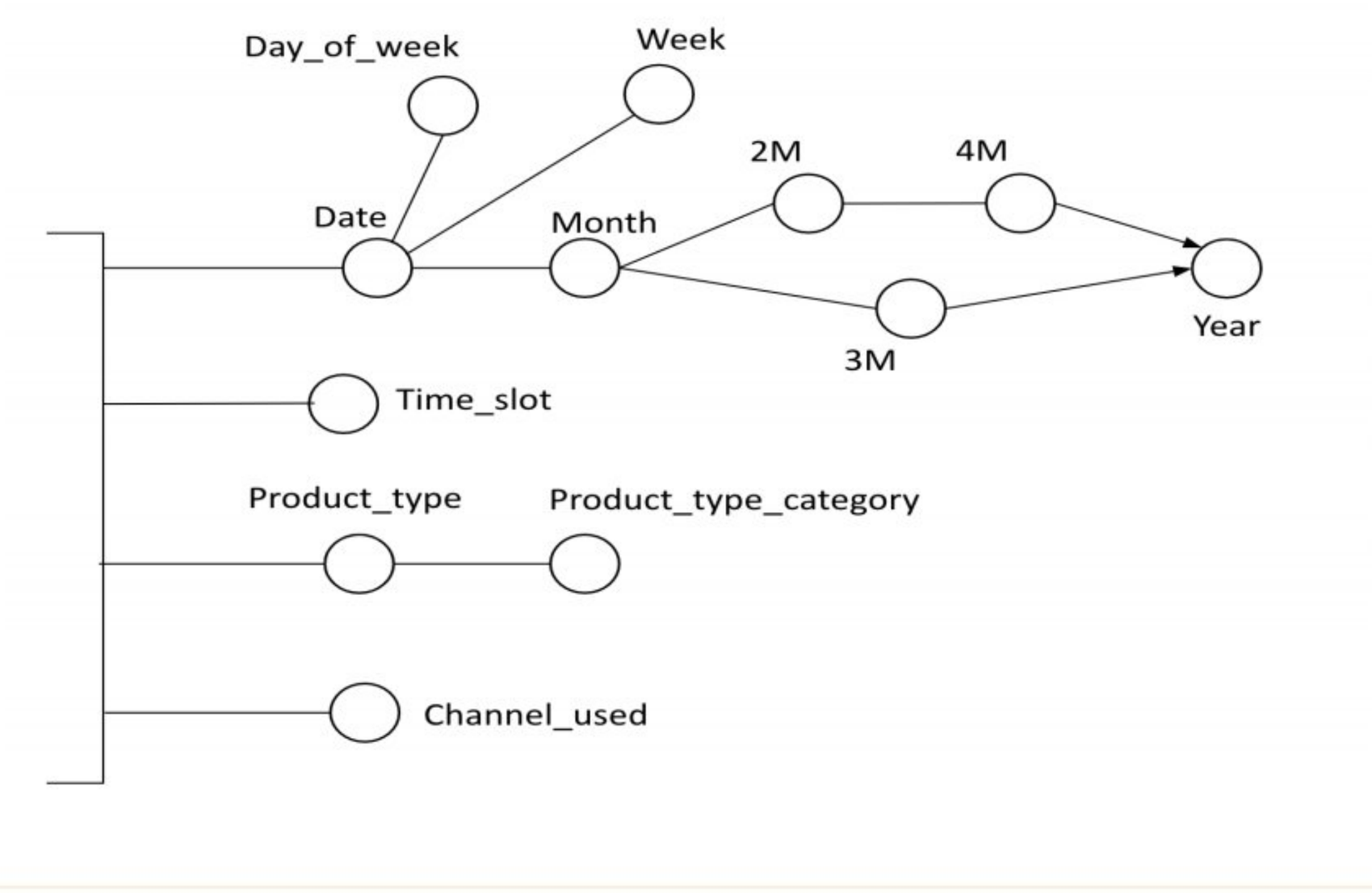

Day_of_the_week
Week
Date
Month
2M
4M
3M
Year
Time_slot
Product_type_category
Product_type
Channel_used



(g)

Risposta errata.

La risposta corretta è:



Domanda 5

Risposta errata

Punteggio ottenuto -0,15 su 1,00

1 points (penalty 15% for a wrong answer)

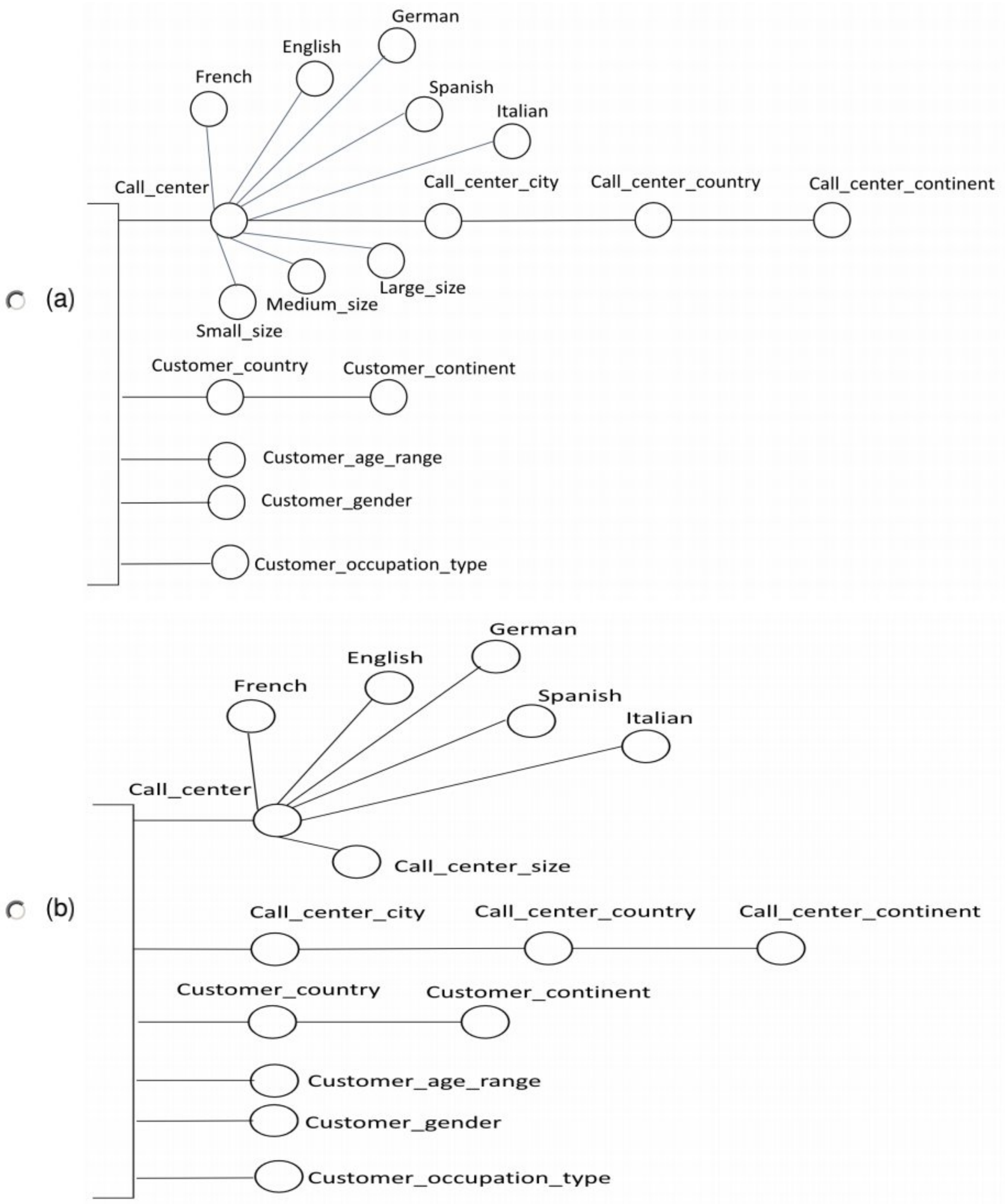
We want to analyze the information about to a chain of call centers that perform sales campaigns for different types of products supplied by a multinational company.

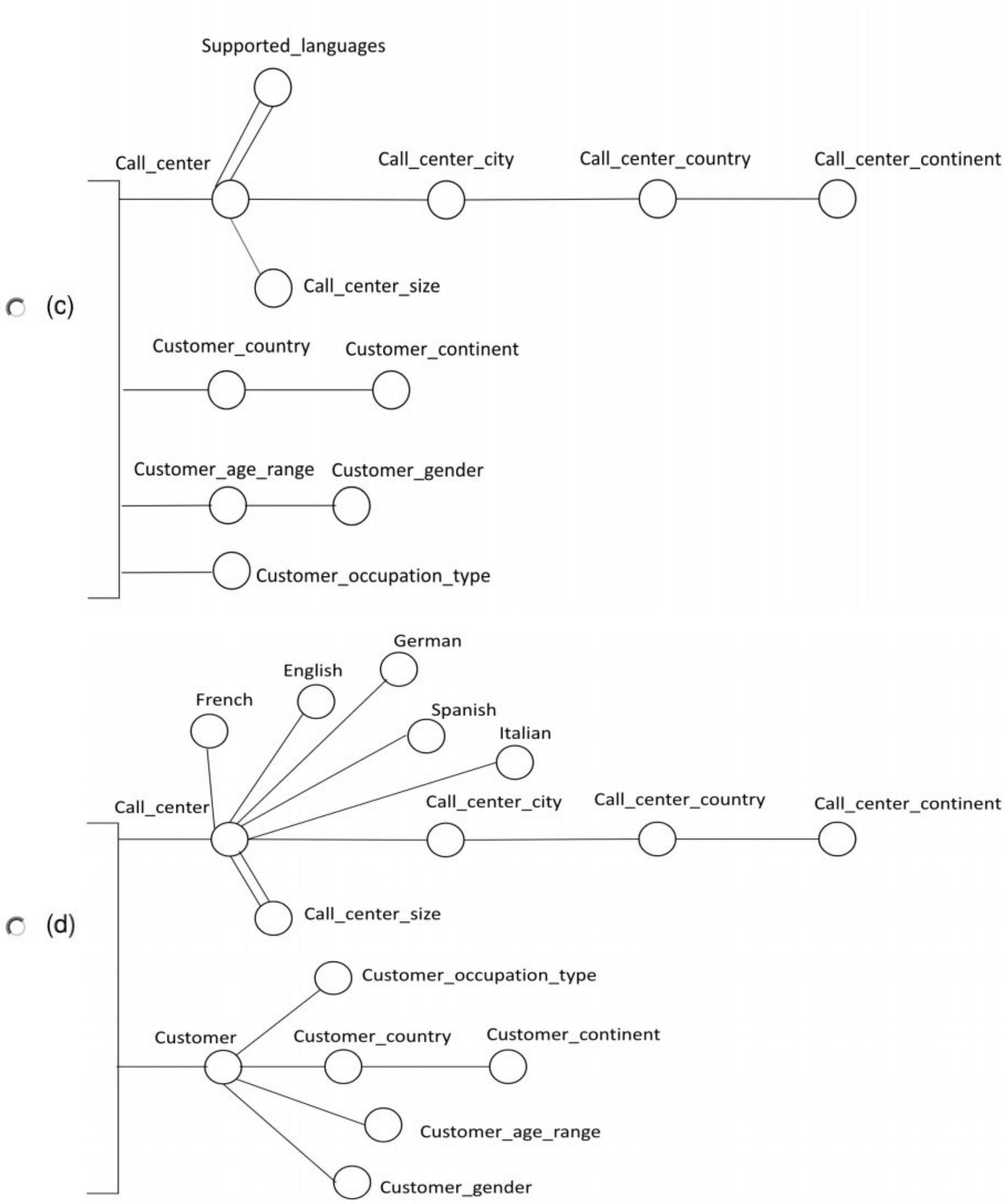
Each call center provides a multilingual and multichannel service for contacting potential customers and proposing products. The different communication channels used by each call center include, in addition to telephone calls, for example also emails and SMS messages. Each call center is also able to operate by supporting multiple languages (for example English, French and German). Therefore, each call center can contact customers in different countries around the world.

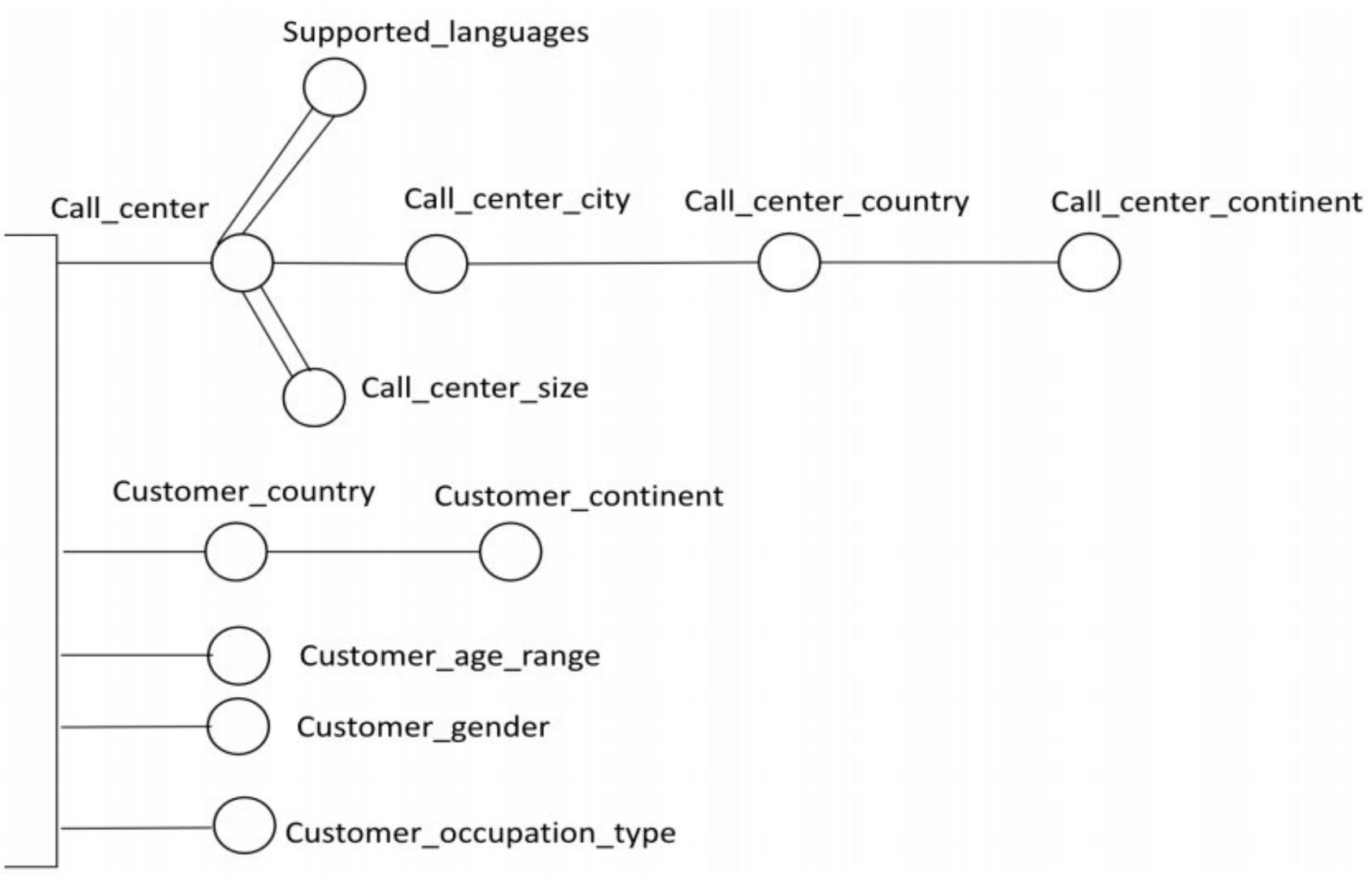
We want to analyze (1) the average number of contacted customers per product and (2) the average number of positive responses received per product, based on:

- call center, characterized by a name, its geographic location (expressed in terms of city, state, and continent where the call center is located), the size of the call center (expressed as small, medium, or large, based on the number of employees) and the indication of the languages supported by the call center (one or more languages among English, French, Italian, German and Spanish)
- temporal information on when the call center contacted the customer, expressed in terms of date, day of the week, week, month, 2-months, 3-months, 4-months, years and timeslot (a value among morning, afternoon, evening, and night)
- type of product proposed to the contacted customer
- categorization of the product type into basic, medium or luxury
- type of channel used to contact the customer (a value among telephone call, email, and SMS message)
- state and continent of the contacted customer
- characteristics of the contacted customer, in terms of gender, age range of the customer (a value among less than 20 years, between 20 and 40 years, between 40 and 60 years, and more than 60 years) and type of occupation of the customer (for example employee or freelancer or retired)

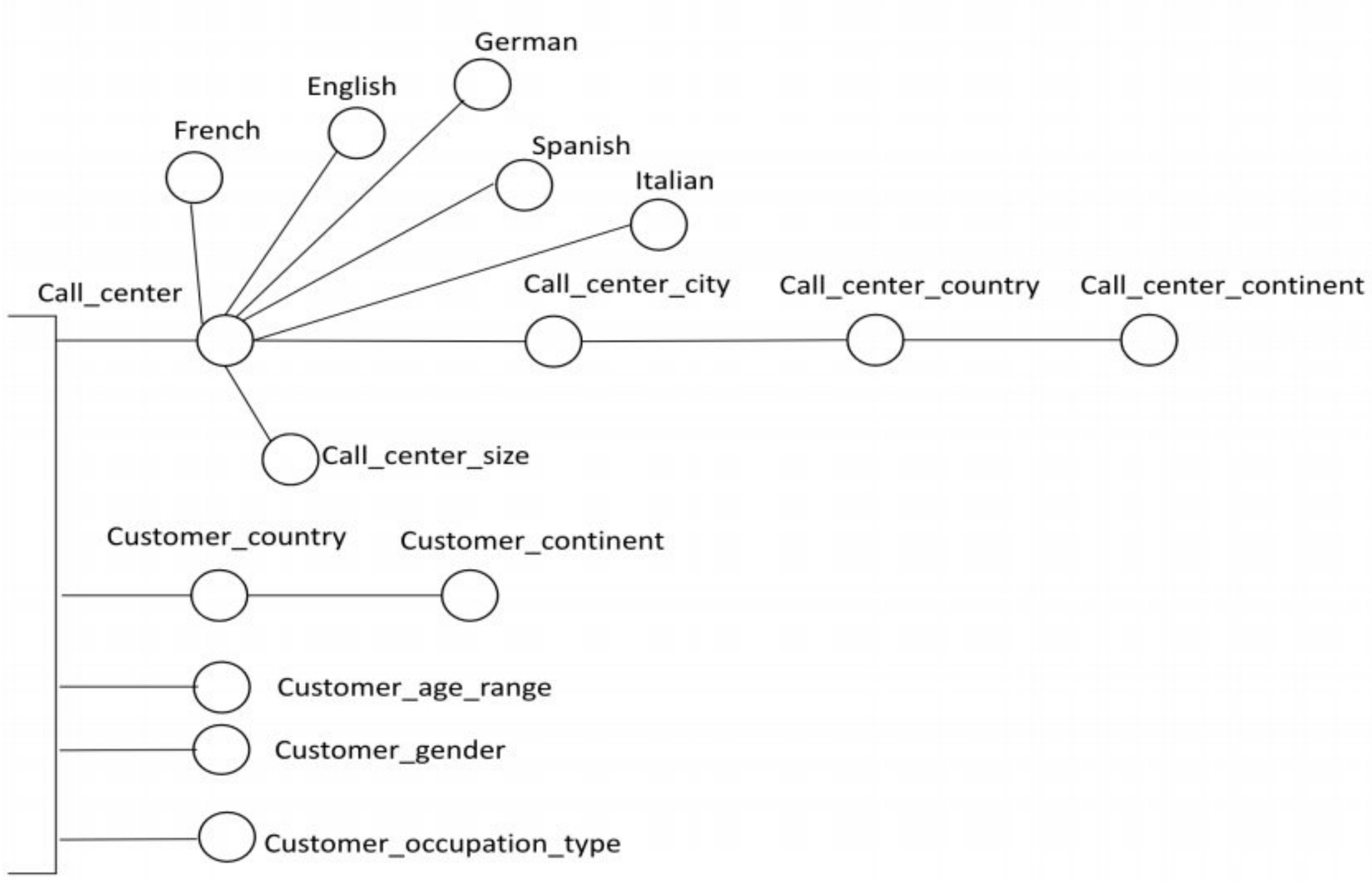
Select, from the dimensions proposed below, those that meet the requirements described in the problem specification.


German
English
French
Spanish
Italian
Call_center
Call_center_city
Call_center_country
Call_center_continent
Large_size
Medium_size
Small_size
Customer_country
Customer_continent
Customer_age_range
Customer_gender
Customer_occupation_type
(a)
German
English
French
Spanish
Italian
Call_center
Call_center_size
(b)
Call_center_city
Call_center_country
Call_center_continent
Customer_country
Customer_continent
Customer_age_range
Customer_gender
Customer_occupation_type


Supported_languages
Call_center
Call_center_city
Call_center_country
Call_center_continent
Call_center_size
(c)
Customer_country
Customer_continent
Customer_age_range
Customer_gender
Customer_occupation_type
German
English
French
Spanish
Italian
Call_center
Call_center_city
Call_center_country
Call_center_continent
Call_center_size
(d)
Customer_occupation_type
Customer
Customer_country
Customer_continent
Customer_age_range
Customer_gender

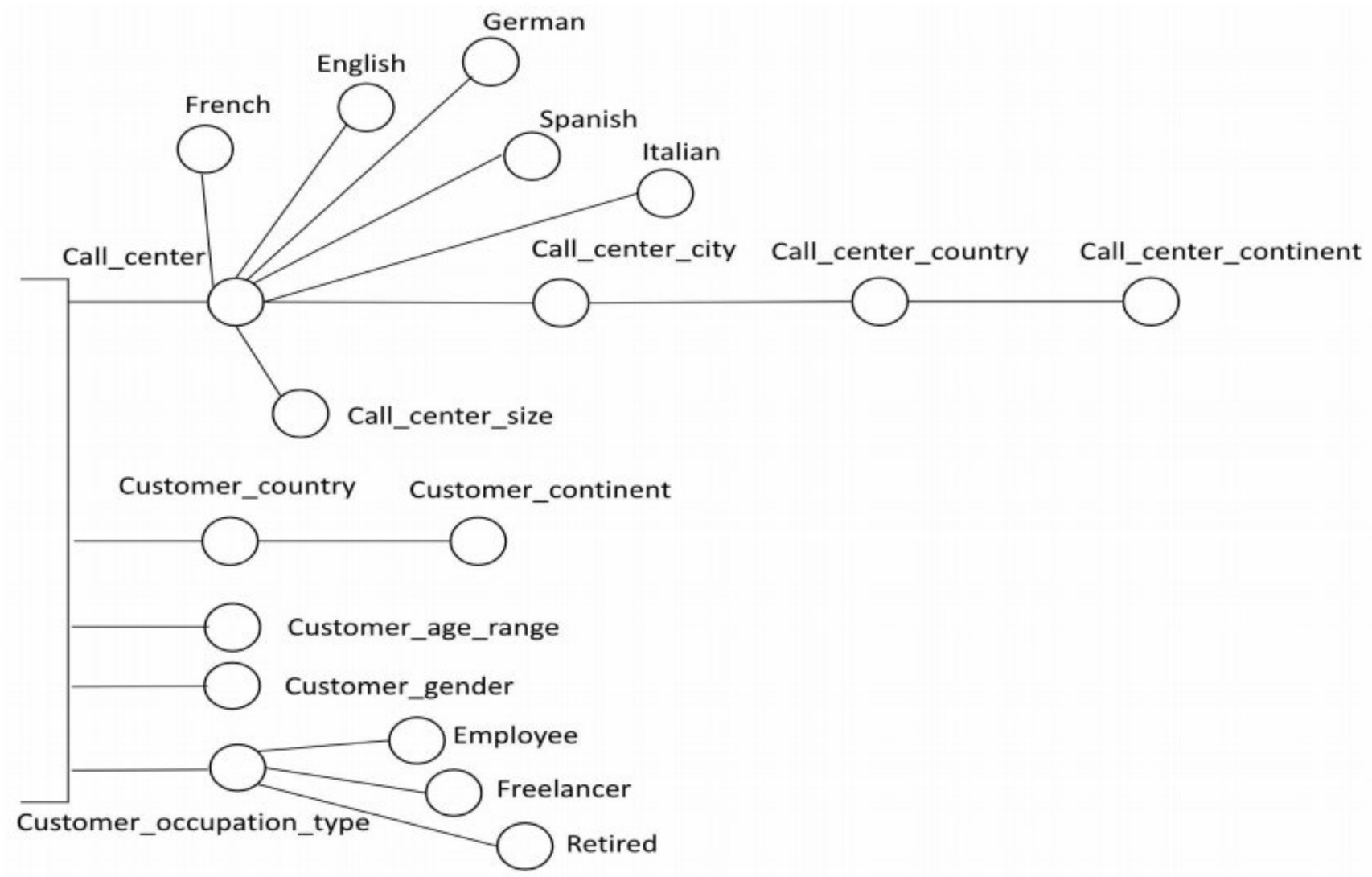

Supported_languages
Call_center
Call_center_city
Call_center_country
Call_center_continent
Call_center_size
Customer_country
Customer_continent
Customer_age_range
Customer_gender
Customer_occupation_type

(e)


German
English
French
Spanish
Italian
Call_center
Call_center_city
Call_center_country
Call_center_continent
Call_center_size
Customer_country
Customer_continent
Customer_age_range
Customer_gender
Customer_occupation_type

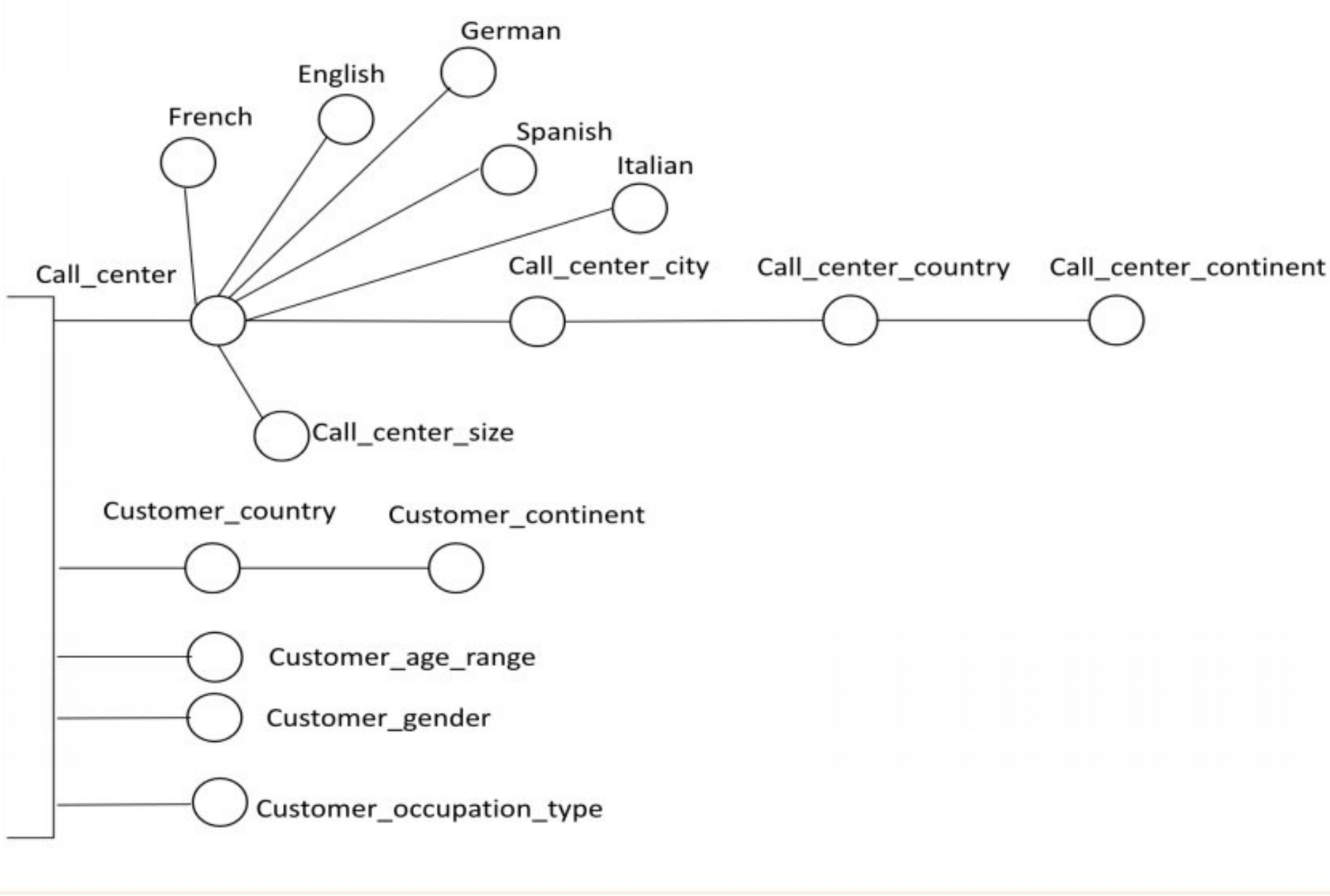
(f)

(g)



Risposta errata.

La risposta corretta è:



Domanda 6

Risposta corretta

Punteggio ottenuto 1,00 su 1,00

1 points (penalty 15% for a wrong answer)

We want to analyze the information about to a chain of call centers that perform sales campaigns for different types of products supplied by a multinational company.

Each call center provides a multilingual and multichannel service for contacting potential customers and proposing products. The different communication channels used by each call center include, in addition to telephone calls, for example also emails and SMS messages. Each call center is also able to operate by supporting multiple languages (for example English, French and German). Therefore, each call center can contact customers in different countries around the world.

We want to analyze (1) the average number of contacted customers per product and (2) the average number of positive responses received per product, based on:

- call center, characterized by a name, its geographic location (expressed in terms of city, state, and continent where the call center is located), the size of the call center (expressed as small, medium, or large, based on the number of employees) and the indication of the languages supported by the call center (one or more languages among English, French, Italian, German and Spanish)
- temporal information on when the call center contacted the customer, expressed in terms of date, day of the week, week, month, 2-months, 3-months, 4-months, years and timeslot (a value among morning, afternoon, evening, and night)
- type of product proposed to the contacted customer
- categorization of the product type into basic, medium or luxury
- type of channel used to contact the customer (a value among telephone call, email, and SMS message)
- state and continent of the contacted customer
- characteristics of the contacted customer, in terms of gender, age range of the customer (a value among less than 20 years, between 20 and 40 years, between 40 and 60 years, and more than 60 years) and type of occupation of the customer (for example employee or freelancer or retired)

Select from the list all and only those attributes needed to correctly model the measures in the fact table required by the specification (multiple correct answers possible).

Scegli una o più alternative:

- [x] (a) Total number of contacted customers ✓
- [ ] (b) Total number of operators
- [ ] (c) Total number of call centers
- [ ] (d) Average number of calls per call center
- [x] (e) Total number of positive responses ✓
- [x] (f) Total number of proposed products ✓

- [ ] (g) Total number of calls
- [ ] (h) Average daily number of positive responses

Risposta corretta.

La risposta corretta è: Total number of contacted customers, Total number of positive responses, Total number of proposed products

Domanda 7

Risposta non data

Non valutata

This is not an exam question.

You can use the text box below for notes or drafts (for example, to write the intermediate steps of an exercise).

Any comments/feedback for the teacher can be written here.

The text entered in this exercise will not be considered in the exam correction phase.

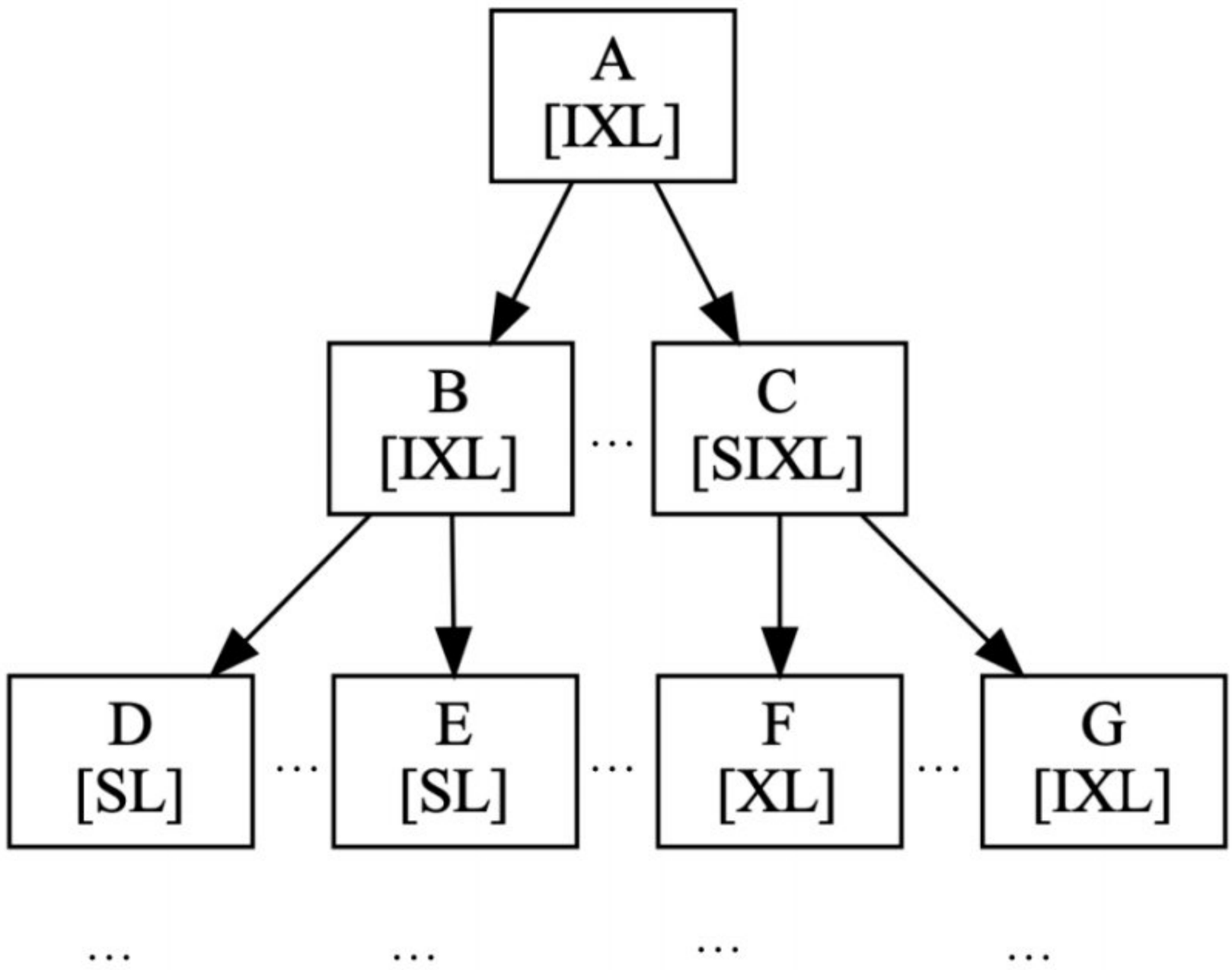
Domanda 8

Risposta non data

Punteggio max.: 1,00

1 point (penalty 15% for a wrong answer)

The following tree shows a subset of a database with different levels of granularity (e.g. tables/fragments/records). For each node in the tree, the locks already acquired by other transactions are shown in [square brackets].



Note that in the tree, each node always inherits a lock of at least the same level as its parent, even if not explicitly indicated in the image.

Which of the following lock sequences can be acquired by a new transaction?

- (a) IXL on A, ISL on B, XL on E
- (b) IXL on A, ISL on B, SL on E
- (c) ISL on A, ISL on B, IXL on E
- (d) IXL on A, IXL on B, IXL on E
- (e) SIXL on A, ISL on B, ISL on E
- (f) ISL on A, SL on B, SL on E
- (g) ISL on A, ISL on B, XL on E

Risposta errata.

La risposta corretta è: IXL on A, ISL on B, SL on E

Domanda 9

Risposta non data

Punteggio max.: 1,50

1.5 points (penalty 15% for a wrong answer)

The MAX (complete) linkage policy states that the distance between two clusters X and Y can be com puted as:

$$dist(X, Y) = max_{x \in X, y \in Y} dist(x, y)$$

where $dist(x, y)$ is a distance that can be computed for any pair of points.

For a dataset of 5 points, the following distance matrix is calculated:

	a	b	c	d	e
a	0	10	6	5	13
b	10	0	21	12	25
c	6	21	0	4	11
d	5	12	4	0	2
e	13	25	11	2	0

Agglomerative hierarchical clustering is applied to extract 3 clusters. The "MAX linkage" (complete linkage) policy is used.

What are the 3 clusters obtained?

- (a) {d, e}, {a}, {b, c}
- (b) {d, e}, {a, b}, {c}
- (c) {a, d}, {b, e}, {c}
- (d) It is not possible to answer the question with the available information.
- (e) {b}, {d, e}, {a, c}
- (f) None of the other answers is correct.
- (g) {a}, {b}, {c, d, e}
- (h) {c, e}, {a, d}, {b}

Risposta errata.

La risposta corretta è: {b}, {d, e}, {a, c}

Domanda 10

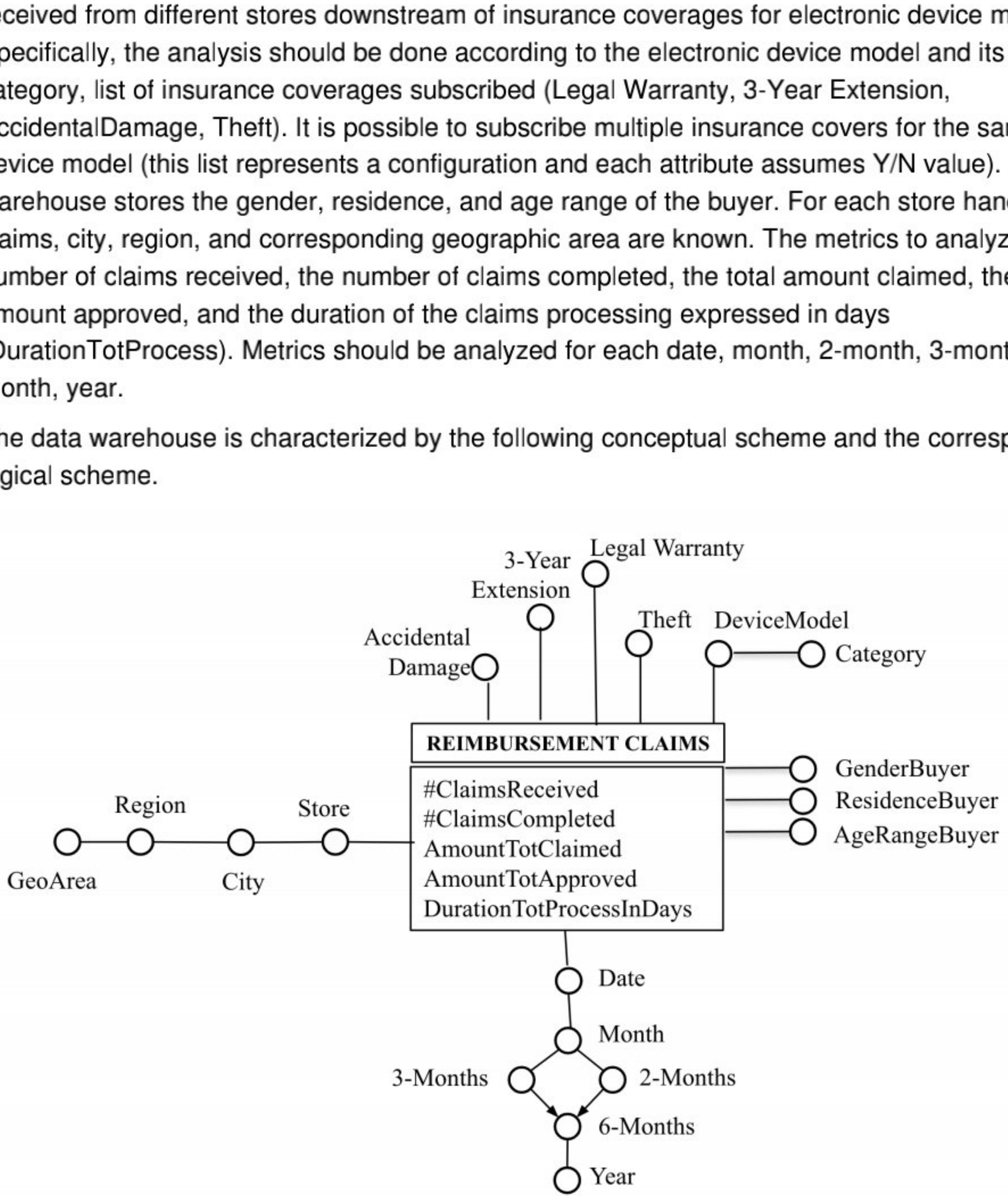
Completo

Punteggio ottenuto 2,50 su 3,00

3 points (no penalty for a wrong answer)

The following data warehouse describes the trend over time of REIMBURSEMENT CLAIMS received from different stores downstream of insurance coverages for electronic device models. Specifically, the analysis should be done according to the electronic device model and its category, list of insurance coverages subscribed (Legal Warranty, 3-Year Extension, AccidentalDamage, Theft). It is possible to subscribe multiple insurance covers for the same device model (this list represents a configuration and each attribute assumes Y/N value). The data warehouse stores the gender, residence, and age range of the buyer. For each store handling claims, city, region, and corresponding geographic area are known. The metrics to analyze are the number of claims received, the number of claims completed, the total amount claimed, the total amount approved, and the duration of the claims processing expressed in days (DurationTotProcess). Metrics should be analyzed for each date, month, 2-month, 3-month, 6-month, year.

The data warehouse is characterized by the following conceptual scheme and the corresponding logical scheme.



STORE(**IDStore**, Store, City, Region, GeoArea)
DEVICE-MODEL (**IDDeviceModel**, DeviceModel, Category)
JUNK-INSURANCE_COVERAGES (**IDJIC**, LegalWarranty, 3-YearExtension, AccidentalDamage, The ft)
JUNK-BUYER-FEATURES (**IDJBF**, Gender, Residence, AgeRange)
TIME(**IDTime**, Date, Month, 2-Months, 3-Months, 6-Months, Year)
REIMBURSEMENT-CLAIMS (**IDStore**, **IDDeviceModel**, **IDJIC**, **IDJBF**, **IDTime**, #ClaimsReceived, # ClaimsCompleted, AmountTotClaimed, AmountTotApproved, DurationTotProcessInDays)

Considering insurance coverages that include only the legal warranty (attribute LegalWarranty), separately by 3-month period and store city, show:

- the percentage of completed claims compared to those received
- the average duration of processing per claim received

Associate each displayed record with a ranking position:

- according to the total amount approved separately by store geographic area (1 for the record with the lowest total amount approved)
- according to the difference between the total claimed amount and the total approved amount (1 for the record with the highest value of the difference between the total claimed amount and the total approved amount) separately by year

---

select city , 3-months
-100* sum(cliamscompleted)/sum(claimsrecived)
-sum(durationtotprocessindays)/sum(claimsreceived)
-rank()over (partiton by geoarea order by sum(amounttotapproved)desc)
-rank() over (partition by year order by sum(amounttotclaimed)/sum(amounttotapproved))
from reimbursment-claims rc, junk-insurance-coverage j , time t, store s
where rc.idtime=t.idtime and rc.idjic=j.idjic and rc.idstore=s.idstore
and junk-insurance-coerage(idjic)='legalwarranty'
group by 3-months , city,geoarea , year

Select 3-Months, City, 100*SUM(#ClaimsCompleted)/SUM(#ClaimsReceived),
SUM(DurationTotProcessInDays)/SUM(#ClaimsReceived)
RANK() OVER (PARTITION BY GeoArea ORDER BY SUM(AmountTotApproved))
RANK() OVER (PARTITION BY Year ORDER BY SUM(AmountTotClaimed) -
SUM(AmountTotApproved) DESC)
FROM STORE, TIME, JUNK-INSURANCE_COVERAGES, REIMBURSEMENT-CLAIMS
WHERE join AND LegalWarranty = ‘Y’ AND 3-YearExtension = ‘N’ AND AccidentalDamage = ‘N’
AND Theft= ‘N’
GROUP BY 3-Months, City, GeoArea, Year

Commento:

select city , 3-months

-100* sum(cliamscompleted)/sum(claimsrecived)

-sum(durationtotprocessindays)/sum(claimsreceived)

-rank()over (partiton by geoarea order by sum(amounttotapproved)desc)

-rank() over (partition by year order by sum(amounttotclaimed)/- sum(amounttotapproved))

from reimbursment-claims rc, junk-insurance-coverage j , time t, store s

where rc.idtime=t.idtime and rc.idjic=j.idjic and rc.idstore=s.idstore

~~and junk-insurance-coerage(idjic)='legalwarranty'~~

AND LegalWarranty = 'Y' AND 3-YearExtension = 'N' AND AccidentalDamage = 'N' AND Theft= 'N'

group by 3-months , city,geoarea , year

Domanda 11

Completo

Punteggio ottenuto 3,00 su 4,00

4 points (no penalty for a wrong answer)

The following data warehouse describes the trend over time of REIMBURSEMENT CLAIMS received from different stores downstream of insurance coverages for electronic device models. Specifically, the analysis should be done according to the electronic device model and its category, list of insurance coverages subscribed (Legal Warranty, 3-Year Extension, AccidentalDamage, Theft). It is possible to subscribe multiple insurance covers for the same device model (this list represents a configuration and each attribute assumes Y/N value). The data warehouse stores the gender, residence, and age range of the buyer. For each store handling claims, city, region, and corresponding geographic area are known. The metrics to analyze are the number of claims received, the number of claims completed, the total amount claimed, the total amount approved, and the duration of the claims processing expressed in days (DurationTotProcess). Metrics should be analyzed for each date, month, 2-month, 3-month, 6-month, year.

The data warehouse is characterized by the following conceptual scheme and the corresponding logical scheme.

STORE(**IDStore**, Store, City, Region, GeoArea)
DEVICE-MODEL (**IDDeviceModel**, DeviceModel, Category)
JUNK-INSURANCE_COVERAGES (**IDJIC**, LegalWarranty, 3-YearExtension, AccidentalDamage, The ft)
JUNK-BUYER-FEATURES (**IDJBF**, Gender, Residence, AgeRange)
TIME(**IDTime**, Date, Month, 2-Months, 3-Months, 6-Months, Year)
REIMBURSEMENT-CLAIMS (**IDStore**, **IDDeviceModel**, **IDJIC**, **IDJBF**, **IDTime**, #ClaimsReceived, # ClaimsCompleted, AmountTotClaimed, AmountTotApproved, DurationTotProcessInDays)

Considering the years before 2020, separately by electronic device model, semester (6-Months attribute), and buyer gender, show:

- the average approved amount per completed claim
- the cumulative approved amount with passing semesters, separately by electronic device model
- the cumulative claimed amount independently by device model, semester, and Gender

---

select devicemodel , 6-months, gender
-sum(amounttotapproved)/ sum (claimscompleted)
-sum (sum (amounttotapproved) over (partition by devicemodel order by 6-months rows unbounded preceding)
-sum(sum amounttotcliamed) over (partition by devicemodel,gender , 6-moths)
from reimbursment-claims rc, time t, device-model dm , junk-buyer-feature j
where rc.idtime=t.idtime and rc.iddevicemodel=dm.iddevicemodel and rc.idjbf=j.idjbf
and year <2020

group by devicemodel , 6-months , gender

Select DeviceModel, 6-Months, Gender, SUM(AmountTotApproved)/SUM(#ClaimsCompleted),
SUM(SUM(AmountTotApproved)) OVER (PARTITION BY DeviceModel ORDER BY 6-Months
ROWS UNBOUNDED PRECEDING)
SUM(SUM(AmountTotClaimed)) OVER ()
FROM JUNK-BUYER-FEATURES, TIME, REIMBURSEMENT-CLAIMS, DEVICE-MODEL
WHERE join AND Year < 2020
GROUP BY DeviceModel, 6-Months, Gender

Commento:
select devicemodel , 6-months, gender

-sum(amounttotapproved)/ sum (claimscompleted)

-sum (sum (amounttotapproved) over (partition by devicemodel order by 6-months rows unbounded preceding)

-sum(sum amounttotcliamed) over (~~**partition by devicemodel,gender , 6-moths**~~)

from reimbursment-claims rc, time t, device-model dm , junk-buyer-feature j

where rc.idtime=t.idtime and rc.iddevicemodel=dm.iddevicemodel and rc.idjbf=j.idjbf

and year <2020

group by devicemodel , 6-months , gender

Domanda 12

Completo

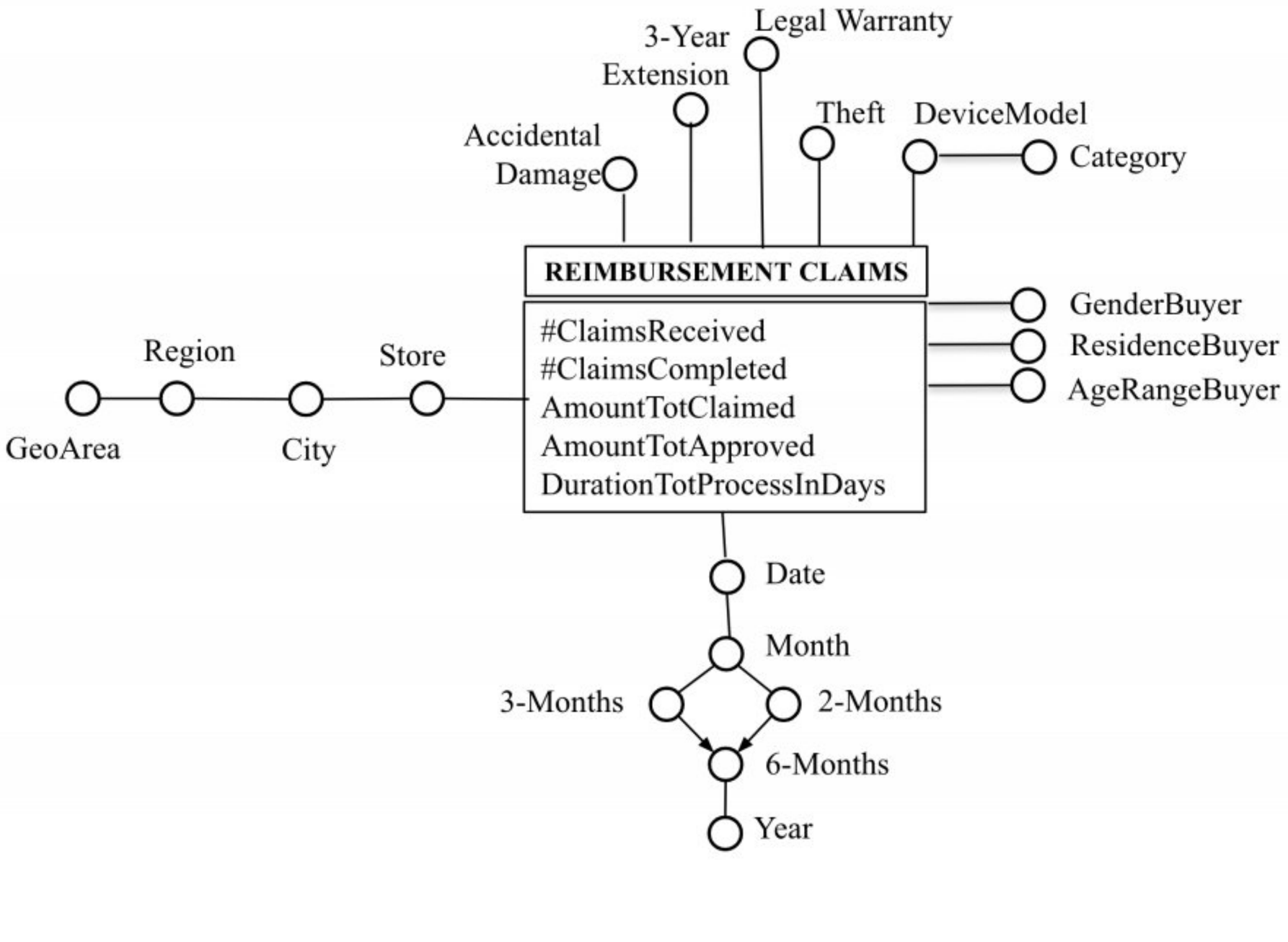
Punteggio ottenuto 2,50 su 4,00

4 points (no penalty for a wrong answer)

The following data warehouse describes the trend over time of REIMBURSEMENT CLAIMS received from different stores downstream of insurance coverages for electronic device models. Specifically, the analysis should be done according to the electronic device model and its category, list of insurance coverages subscribed (Legal Warranty, 3-Year Extension, AccidentalDamage, Theft). It is possible to subscribe multiple insurance covers for the same device model (this list represents a configuration and each attribute assumes Y/N value). The data warehouse stores the gender, residence, and age range of the buyer. For each store handling claims, city, region, and corresponding geographic area are known. The metrics to analyze are the number of claims received, the number of claims completed, the total amount claimed, the total amount approved, and the duration of the claims processing expressed in days

(DurationTotProcess). Metrics should be analyzed for each date, month, 2-month, 3-month, 6-month, year.

The data warehouse is characterized by the following conceptual scheme and the corresponding logical scheme.



STORE(**IDStore**, Store, City, Region, GeoArea)
DEVICE-MODEL (**IDDeviceModel**, DeviceModel, Category)
JUNK-INSURANCE_COVERAGES (**IDJIC**, LegalWarranty, 3-YearExtension, AccidentalDamage, Theft)
JUNK-BUYER-FEATURES (**IDJBF**, Gender, Residence, AgeRange)
TIME(**IDTime**, Date, Month, 2-Months, 3-Months, 6-Months, Year)
REIMBURSEMENT-CLAIMS (**IDStore**, **IDDeviceModel**, **IDJIC**, **IDJBF**, **IDTime**, #ClaimsReceived, #ClaimsCompleted, AmountTotClaimed, AmountTotApproved, DurationTotProcessInDays)

Separately by store, electronic device category, and 2-month period, show:

- the percentage of the requested amount that was approved
- the difference between the number of claims received and those completed
- the total approved amount independently of the store
- the ratio between the approved amount and the total approved amount of all stores located in the same city, separately by electronic device category and 2-month period

---

select store , 2-months,category
-100*(sum amounttotapproved)
-sum(claimsrecieved)/sum(cliamescompleted)

-sum (amounttotapproved)over (partition by store)
-sum(amounttotapproved)/sum (sum amounttotapproved) over (partition by city ,cetegory , 2-months)
from reimbursment-cliam rc, time t, store s, device-model dm
where rc.idtime=t.idtime and rc.idstore=s.idstore and rc.idevicemodel=dm.iddevicemodel
group by store , 2-months , category, city

SELECT Store, Category, 2-Months,
100*SUM(AmountTotApproved)/SUM(AmountTotClaimed),
SUM(#ClaimsReceived) - SUM(#ClaimsCompleted),
SUM(SUM(AmountTotApproved)) OVER (PARTITION BY Category, 2-Months)
SUM(AmountTotApproved)/SUM(SUM(AmountTotApproved)) OVER (PARTITION BY City, Category, 2-Months)
FROM REIMBURSEMENT-CLAIMS, TIME, DEVICE-MODEL, STORE
WHERE join
GROUP BY Store, Category, 2-Months, City

Commento:
select store , 2-months,category

-100*(sum amounttotapproved)/**SUM(AmountTotClaimed),**

-sum(claimsrecieved)/ **-** sum(cliamescompleted)

-sum (amounttotapproved)over (partition by ~~**store**~~ **Category, 2M**)

-sum(amounttotapproved)/sum (sum amounttotapproved) over (partition by city ,cetegory , 2-months)

from reimbursment-cliam rc, time t, store s, device-model dm

where rc.idtime=t.idtime and rc.idstore=s.idstore and rc.idevicemodel=dm.iddevicemodel

group by store , 2-months , category, city

Domanda 13

Risposta non data

Punteggio max.: 1,00

1 points (penalty 15% for a wrong answer)

A decision tree is trained on a dataset containing N points, M attributes, and P classes.
What is the maximum number of leaf nodes that the trained tree can have?

- (a) None of the other answers is correct.
- (b) N + P nodes
- (c) P nodes
- (d) N + M nodes
- (e) It is not possible to answer with the information provided.
- (f) P + M nodes
- (g) M nodes
- (h) N nodes

Risposta errata.

La risposta corretta è: N nodes

Domanda 14

Parzialmente corretta

Punteggio ottenuto 1,55 su 5,00

5 points overall (penalty 15% for a wrong answer)

The following tables are provided:

Song(CodS, Title, Genre, CodA)
Artist(CodA, Name, Surname, Nationality, BirthDate)
User(CodU, Name, Surname, Nationality, BirthDate , Email)
UserLike(CodU, CodS, Date, Platform)

Assume the following cardinalities:

- card(Song) = 10^8 tuples
  - Distinct values of Genre = 10
- card(ARTIST) = $5 \cdot 10^6$ tuples
  - Distinct values of Nationality = 100
  - MIN(BirthDate) = 1/1/1900, MAX(BirthDate) = 31/12/1999
- card(USER) = $2 \cdot 10^7$ tuples
  - MIN(BirthDate) = 1/1/1930, MAX(BirthDate) = 31/12/2004
  - Distinct values of Nationality = 100
- card(USERLIKE) = 10^{10} tuples
  - Distinct values of Platform = 5
  - MIN(Date) = 1/1/2003, MAX(Date) = 31/12/2022

Furthermore, assume the following reduction factor for the having clauses:

- Having COUNT(*)≥150 = 1/5

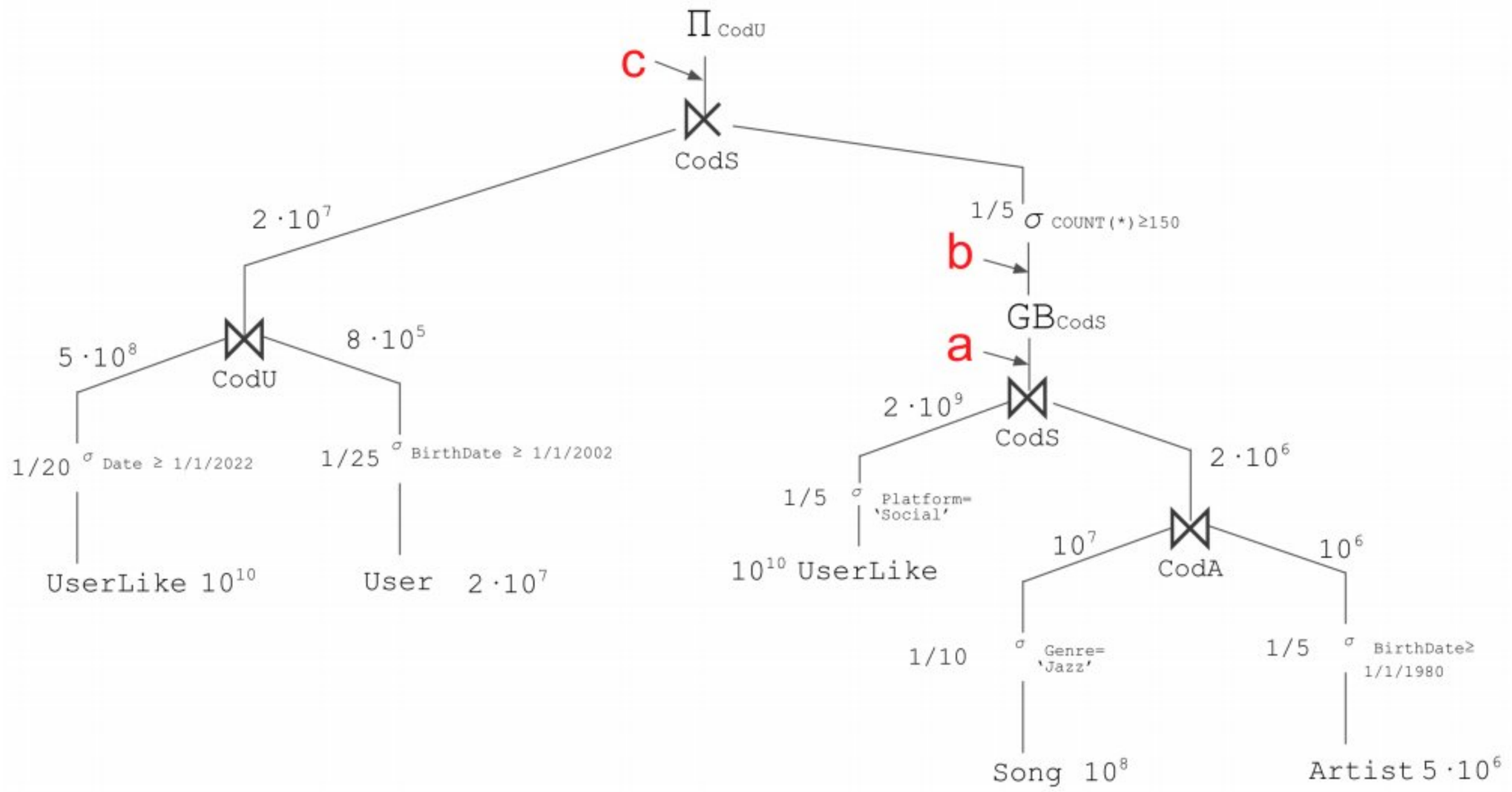
Consider the following query:

```
select UL1.CodU
from  User U, UserLike UL1
where U.CodU=UL1.CodU
and UL1.Date ≥ 1/1/2022
and U.BirthDate ≥ 1/1/2002
and UL1.CodS IN (SELECT UL2.CodS
  FROM Song S, Artist A, UserLike UL2
  WHERE S.CodA=A.CodA and UL2.CodS=S.CodS
  and UL2.Platform='Social'
  and A.BirthDate≥ 1/1/1980
  and S.Genre='Jazz'
  GROUP BY UL2.CodS
  HAVING COUNT(*)≥150)
```

Cardinalities

(1.5 points, penalty -15% for a wrong answer)

The figure below represents the query tree for the query above.



Select the correct answer for the cardinality of **(a)**:

- $4 \cdot 10^6$
- $4 \cdot 10^7$ ✓
- $5 \cdot 10^6$
- $4 \cdot 10^5$

Punteggio ottenuto 5,00 su 5,00

La risposta corretta è: $4 \cdot 10^7$

Select the correct answer for the cardinality of **(b)**:

- 10^4
- 10^7 ✗
- $2 \cdot 10^6$
- 10^8

Punteggio ottenuto -0,75 su 5,00

La risposta corretta è: $2 \cdot 10^6$

Select the correct answer for the cardinality of **(c)**:

- $8 \cdot 10^4$
- $2 \cdot 10^5$ ✗
- $2 \cdot 10^7$
- $8 \cdot 10^6$

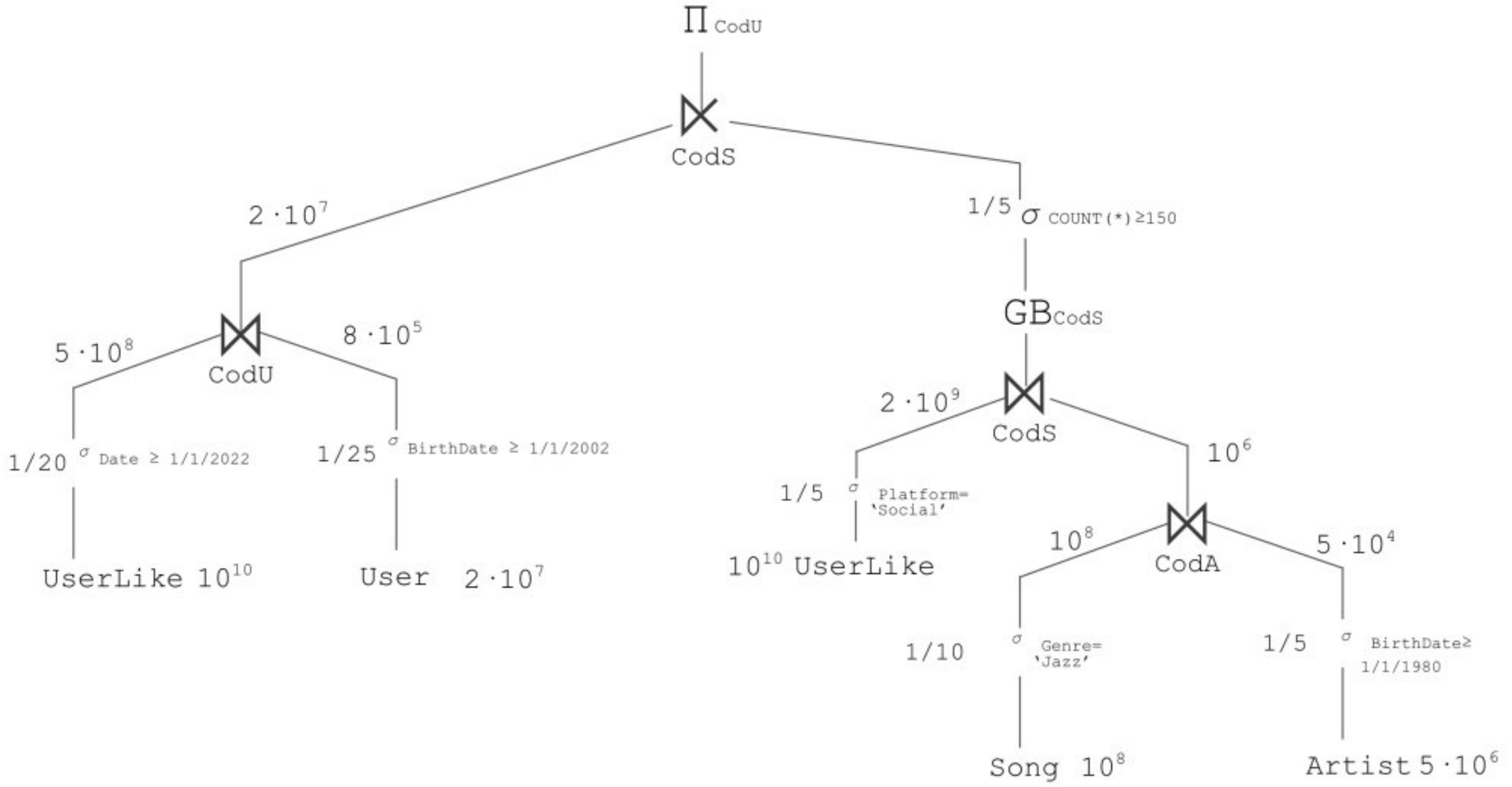
Punteggio ottenuto -0,75 su 5,00

La risposta corretta è: $8 \cdot 10^{4}$

Indexes

(1.5 points, penalty -15% for a wrong answer)

The figure below represents the query tree for the query above.



Select, for each table, one or more secondary physical structures to increase query performance (if possible) among the options below.

Table USERLIKE

- CREATE INDEX IndexA ON USERLIKE(Date) - HASH
- None of the proposed secondary physical structures on this table would increase query performance
- CREATE INDEX IndexB ON USERLIKE(Date) - B+- Tree ✓

Punteggio ottenuto 3,00 su 3,00

La risposta corretta è: CREATE INDEX IndexB ON USERLIKE(Date) - B+- Tree

Table USERLIKE

- CREATE INDEX IndexD ON USERLIKE(Platform) - B+- Tree
- None of the proposed secondary physical structures on this table would increase query

performance ✓

- CREATE INDEX IndexC ON USERLIKE(Platform) - HASH

Punteggio ottenuto 3,00 su 3,00

La risposta corretta è: None of the proposed secondary physical structures on this table would increase query performance

Table USER

- None of the proposed secondary physical structures on this table would increase query performance
- CREATE INDEX IndexE ON USER(BirthDate) - HASH
- CREATE INDEX IndexF ON USER(BirthDate) - B+- Tree ✓

Punteggio ottenuto 3,00 su 3,00

La risposta corretta è: CREATE INDEX IndexF ON USER(BirthDate) - B+- Tree

Table SONG

- CREATE INDEX IndexH ON SONG(Genre) - B+- Tree
- CREATE INDEX IndexG ON SONG(Genre) - HASH ✓
- None of the proposed secondary physical structures on this table would increase query performance

Punteggio ottenuto 3,00 su 3,00

La risposta corretta è: CREATE INDEX IndexG ON SONG(Genre) - HASH

Table ARTIST

- None of the proposed secondary physical structures on this table would increase query performance ✓
- CREATE INDEX IndexJ ON ARTIST(Nationality) - B+- Tree
- CREATE INDEX IndexI ON ARTIST(Nationality) - HASH

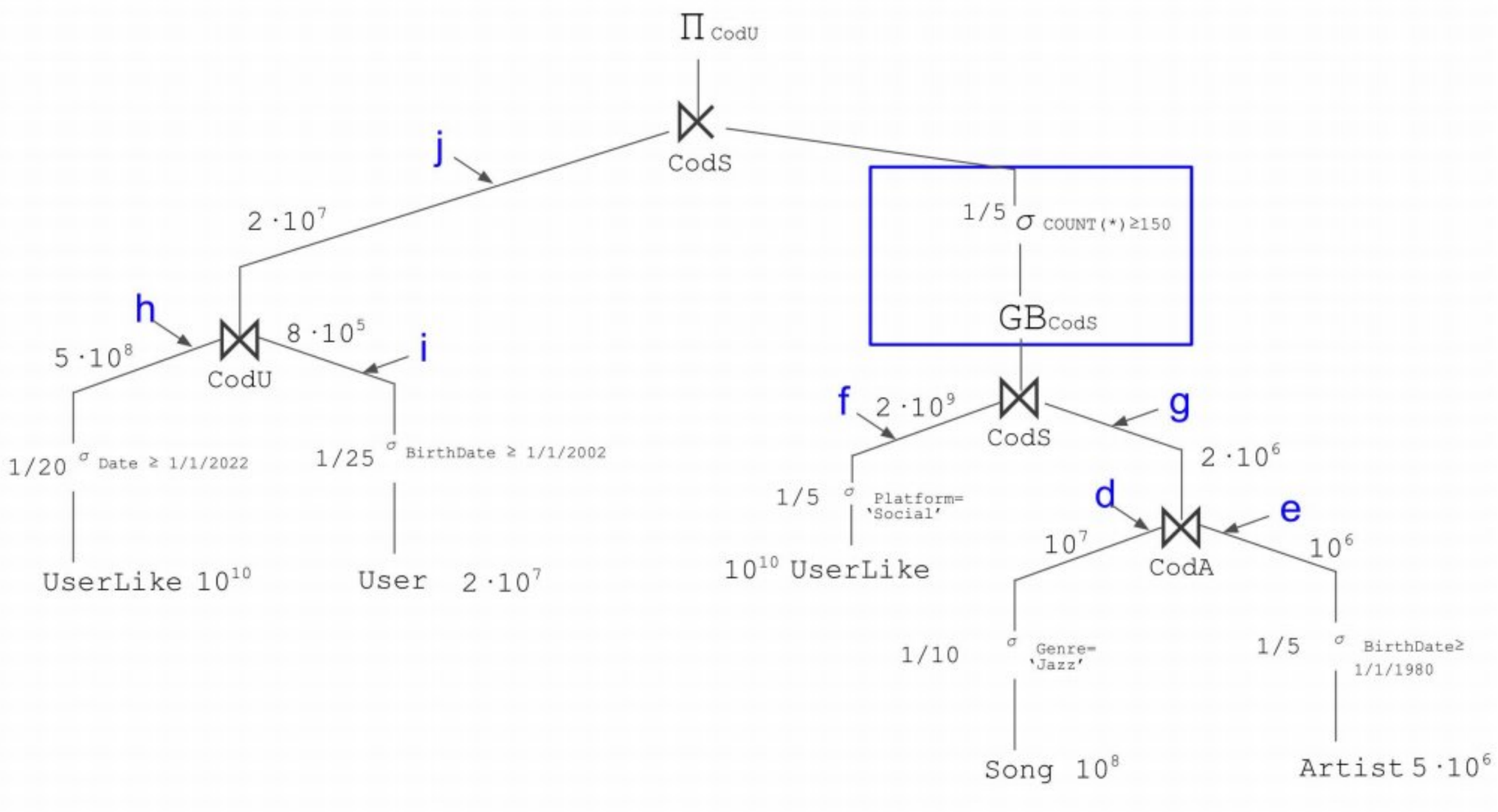
Punteggio ottenuto 3,00 su 3,00

La risposta corretta è: None of the proposed secondary physical structures on this table would increase query performance

Group By Anticipation

(2 points, penalty -15% for a wrong answer)

The figure below represents the query tree for the query above.



Analyze the group by anticipation of GROUP BY **GROUP BY UL2.CodS HAVING COUNT(*)≥150** represented in the box. Select the solution that **allows maximum efficiency** in executing the query (if any).

- It is possible to anticipate it in branch g
- It is possible to anticipate it in branch j ✗
- It is possible to anticipate it in branch h
- It is possible to anticipate it in branch f
- It is possible to anticipate it in branch e
- It is not possible to anticipate the Group By GROUP BY LU2.CodB HAVING COUNT(*)≥150
- It is possible to anticipate it in branch i
- It is possible to anticipate it in branch d

Punteggio ottenuto -3,00 su 20,00

La risposta corretta è: It is possible to anticipate it in branch f

1) La risposta corretta è : $4 \cdot 10^7$
2) La risposta corretta è : $2 \cdot 10^6$
3) La risposta corretta è : $8 \cdot 10^4$
4) La risposta corretta è : CREATE INDEX IndexB ON USERLIKE(Date) - B+- Tree
5) La risposta corretta è : None of the proposed secondary physical structures on this table would increase query performance
6) La risposta corretta è : CREATE INDEX IndexF ON USER(BirthDate) - B+- Tree
7) La risposta corretta è : CREATE INDEX IndexG ON SONG(Genre) - HASH
8) La risposta corretta è : None of the proposed secondary physical structures on this table would increase query performance
9) La risposta corretta è : It is possible to anticipate it in branch f

Domanda 15

Risposta errata

Punteggio ottenuto -0,30 su 2,00

2 points (penalty 15% for a wrong answer)

The following transactional database is given:

	Transactions
0	A D E
1	C E
2	A B C E
3	B D
4	B D
5	C E
6	A B C
7	A E
8	A D E
9	D E

Apply the Apriori algorithm to extract frequent itemsets. Use minsup = 2 (an itemset is frequent if it appears in at least 2 transactions).

What are the length-3 itemsets generated by Apriori **after the join step and prune step** (applying the Apriori principle), **before counting the support** in the database?

- (a) ABC, ACE, ADE
- (b) ABC, ABD, ABE, ACD, ACE, ADE, BCD, BCE
- (c) ABC, ABE
- (d) ABC, ADE
- (e) None of the other answers is correct. ✗
- (f) ABC, ABD, ACE, ADE
- (g) ABE, ACD, ECD
- (h) It is not possible to answer the question with the available information.
- (i) ABC, ABD, ABE, ACD, ACE, ADE, ECD

Risposta errata.

La risposta corretta è: ABC, ABD, ACE, ADE