Technologies for Information Systems

Part I (10 points)

prof. L. Tanca – July 9th, 2018

Available time: 25 minutes

Last Name _____

First Name _____

Student ID _____ **Signature**_____

1) Which type of data integration (materialized or virtual) is used in the case of Data Warehousing? Describe the main differences between the two approaches and justify the answer in your own words, possibly using a small example.
2) Consider the following data mining problems: frequent itemset mining and association rule discovery. Define them, discuss the differences between the two problems, and provide an application example for each of them.

- **During this part of the exam, students are not allowed to consult books or notes.**
- **Students should answer the theoretical questions using their own words, in order for the teachers to be able to assess their real level of understanding.**

Technologies for Information Systems

Part II (23 points)

prof. L. Tanca – July 9th, 2018

Available Time: 2h 00m

Last Name _____

First Name _____

Student ID _____ **Signature**_____

PoliPatents is an organization granting technological patents to companies in America. Each patent may be assigned to multiple companies (i.e., the patent assignees) and is associated with one or more inventors; inventors are the people having participated in the invention that has been patented. PoliPatents stores the patent-related data in a relational database.

UniPatents is a similar organization operating in Europe. UniPatents too allows multiple inventors per patent but, differently from PoliPatents, allows just one assignee per patent. UniPatents stores the patent-related data in a big XML document.

The two organizations have now merged into a unique organization named ***UniPoliPatents***, and the management of UniPoliPatents asks you to integrate the two data sources into a unique **relational** database. You must perform the integration ensuring to lose the least possible amount of information.

The relational schema employed by PoliPatents is as follows:

PATENT (PatentId, Title, GrantDate, Abstract, CPCCategory) *// The id of the patent is an alphanumeric string always starting with 'PP' (e.g., 'PP12345678'). CPCCategory is the category of the patent on the basis of the Cooperative Patent Classification (CPC).*

CITY (CityName, Country)

ASSIGNEE (AssigneeId, Name, CityName)

INVENTOR (InventorId, Firstname, Lastname, CityName)

PATENTASSIGNEE (PatentId, AssigneeId)

PATENTINVENTOR (PatentId, InventorId)

CITATION (CitingPatent, CitedPatent) *// A row in this table indicates that the document describing the citing patent cites the cited patent.*

The following is the DTD of the UniPatents XML document:

```
<!ELEMENT PatentsDB (Patent*)>
<!ELEMENT Patent (Title, Summary, GrantDate, Assignee, Inventors, IPCCategories)>
<!ELEMENT Title (#PCDATA)>
<!ELEMENT Summary (#PCDATA)>
<!ELEMENT GrantDate (#PCDATA)>
<!ELEMENT Assignee (Name, CityName, Country)>
<!ELEMENT Inventors (Inventor+)>
<!ELEMENT IPCCategories (IPCCategory+)>
<!ELEMENT Name (#PCDATA)>
<!ELEMENT CityName (#PCDATA)>
<!ELEMENT Country (#PCDATA)>
<!ELEMENT Inventor (Name, CityName, Country)>
<!ELEMENT IPCCategory EMPTY>
<!ATTLIST Patent PatentId CDATA #REQUIRED>
<!ATTLIST Assignee AssigneeId CDATA #REQUIRED>
<!ATTLIST Inventor InventorId CDATA #REQUIRED>
<!ATTLIST IPCCategory CatName CDATA #REQUIRED>
```

The following is a portion of a valid XML document according to the previous DTD:

```
<PatentsDB>
        <Patent PatentId="UP12345678">
                <Title>New algorithm to automatically solve university exams</Title>
                <Summary> … </Summary>
                <GrantDate>2017-12-15</GrantDate>
                <Assignee AssigneeId="10000001">
                        <Name>Fake Corporation</Name>
                        <CityName>Milan</CityName>
                        <Country>Italy</Country>
                </Assignee>
                <Inventors>
                        <Inventor InventorId="82000000">
                                <Name>Mario#Rossi</Name>
                                <CityName>Monza</CityName>
                                <Country>Italy</Country>
                        </Inventor>
                        <Inventor InventorId="82000001"/>
                                <Name>John#Smith</Name>
                                <CityName>London</CityName>
                                <Country>United Kingdom</Country>
                        </Inventor>
                </Inventors>
                <IPCCategories>
                        <IPCCategory CatName="Life-Saving"/>
                        <IPCCategory CatName="Computer-Aided Design"/>
                </IPCCategories>
        </Patent>
        <Patent>
                …
        </Patent>
</PatentsDB>
```

Remarks:

- The id of the patent is an alphanumeric string always starting with 'UP' (e.g., 'UP12345678').
- IPCCategory is the category of the patent on the basis of International Patent Classification (IPC).
- Inventor names in UniPatents are represented in the form 'Firstname#Lastname'.
- UniPatents allows assignees and inventors to change city, so in different patents the same assignee/inventors may be associated with different cities.

NOTES:

- PoliPatents and UniPatents refer to different geographic areas, and you can assume that patents, assignees and inventors in the two data sources are disjoint.
- CPC categories and IPC categories are different.

1. **Source schema reverse engineering**. Provide, for each input data source, the reverse engineering from the logical schema to the conceptual model (ER graph). For the XML source *UniPatents* provide also the relational translation of the ER schema. (5 points)

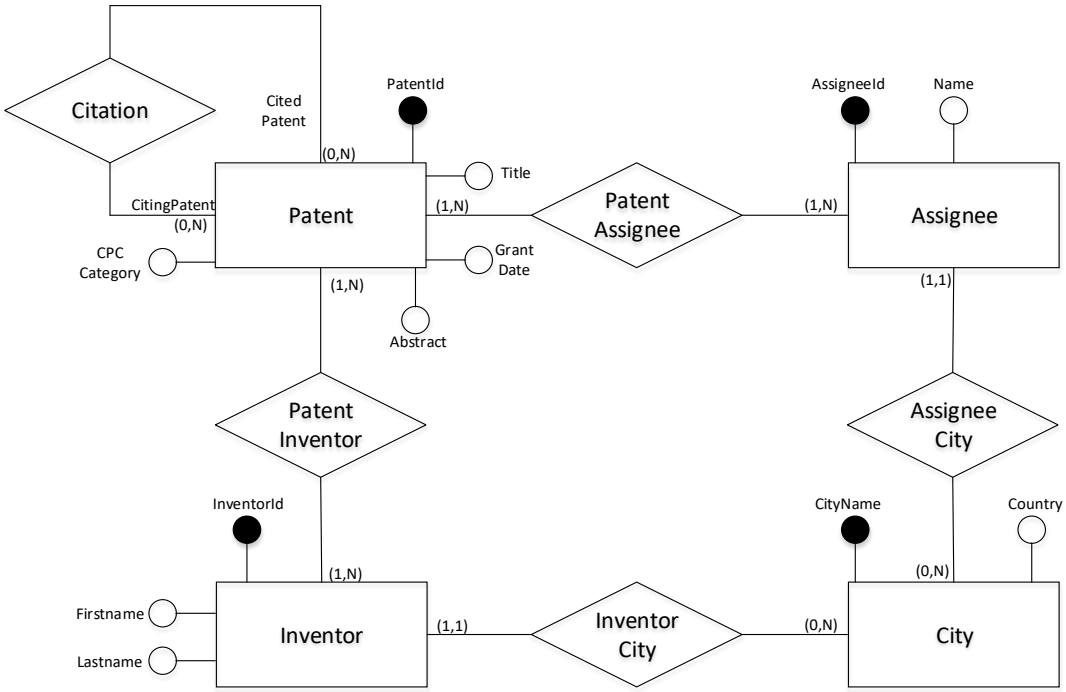
2. **Schema integration**. Design an integrated global conceptual schema (ER graph) for *UniPoliPatents* capturing all the data coming from both *PoliPatents* and *UniPatents*, and provide the corresponding global logical (relational) schema. In more detail, follow these steps:
    a. *Related concept identification and conflict analysis and resolution*. Write a table as shown in the exercise sessions, using the following columns: "PoliPatents concept", "UniPatents concept", "Conflict", "Solution". (3.5 points)
    b. *Integrated conceptual schema* (ER graph). (4 points)
    c. *Conceptual to logical translation of the integrated schema*. (2.5 points)

3. **Query answering and mapping definition**. Consider the query Q: "Find the pairs (id of the patent, name of the assignee) associated with assignees from Milan and patents granted in 2017".
    a. *Query formulation*. Consider query Q posed on the logical schema of *UniPoliPatents* and write it in SQL. (1.5 points)
    b. *Mapping definition*. Write the GAV mappings between the schema of *UniPoliPatents* and the two sources using SQL. For *UniPatents*, write the mappings between the *UniPoliPatents* integrated relational schema and the *UniPatents* relational schema defined at point 1. Write the mappings only for the tables used to answer query Q**.** (4 points)
    c. *Query rewriting*. Show the rewriting of Q on the two data sources using SQL. Again, for *UniPatents* consider the relational schema defined at point 1. (2.5 points)

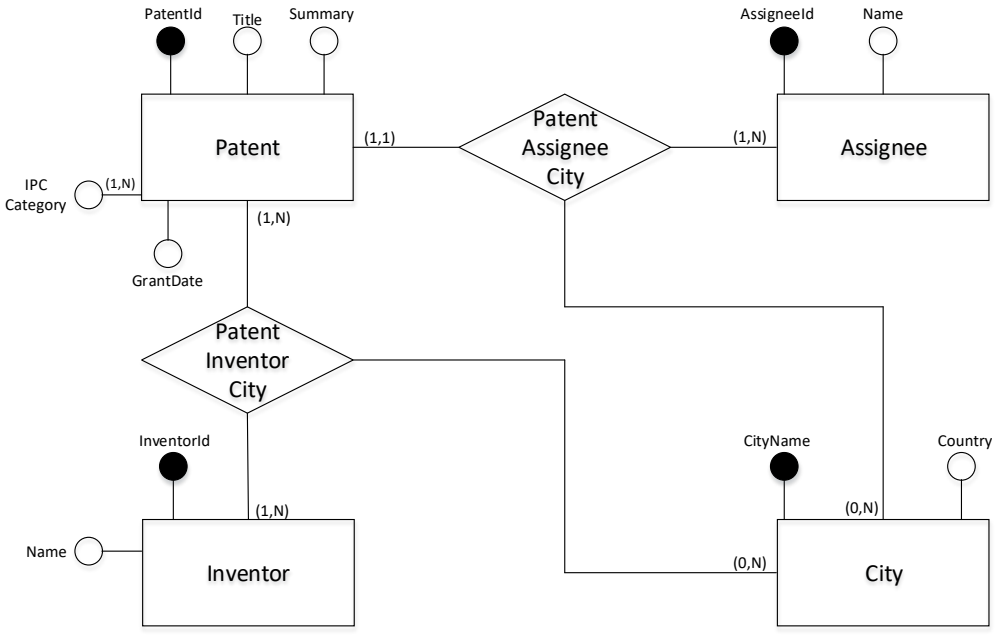
SOLUTION

1. Source schema reverse engineering

PoliPatents



UniPatents



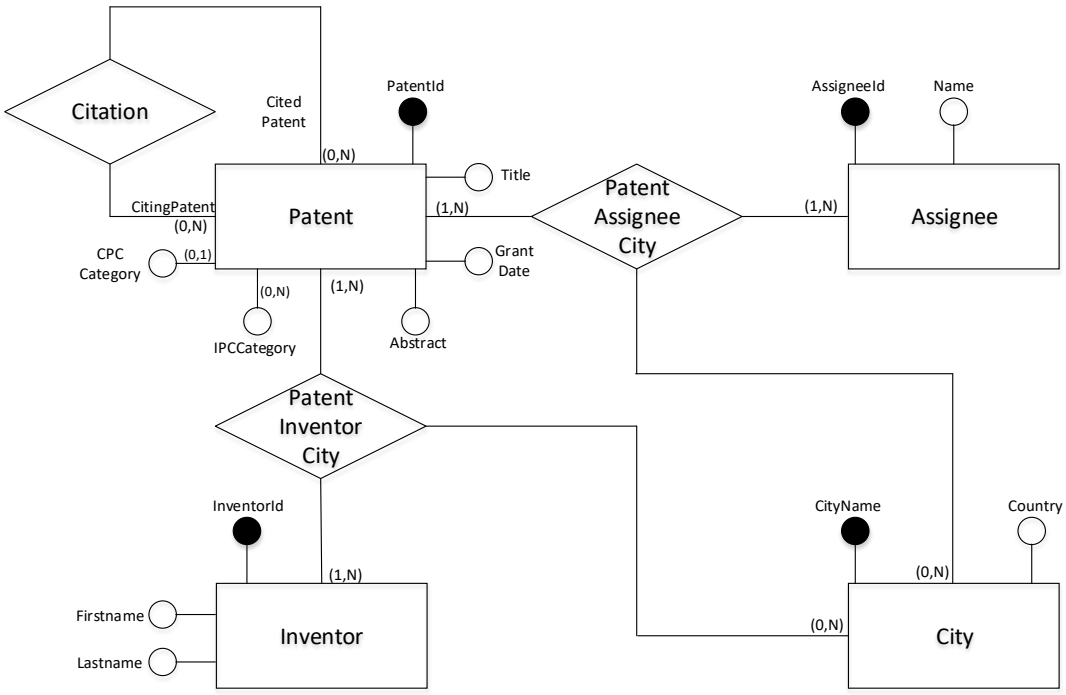
Relational schema
Patent (PatentId, Title, Summary, GrantDate, AssigneeId, AssigneeCityName)
PatentIPCCategory(PatentId, IPCCategoryName)
Assignee (AssigneeId, Name)
Inventor (InventorId, Name)
PatentInventorCity (PatentId, InventorId, CityName)
City (CityName, Country)

2. Schema integration

2a) Related concept identification + conflict analysis and resolution

PoliTwitter	UniTwitter	Conflict	Solution
Patent	Patent	Name conflicts	
		- Abstract → Summary	Abstract
		Cardinality conflicts	
		- A patent may have multiple assignees → A patent has just one assignee	A patent may have multiple assignees
Assignee	Assignee	Structure conflicts	
		- The city of the assignee is fixed → The city of the assignee may vary with the patent	The city of the assignee may vary with the patent
Inventor	Inventor	Structure conflicts	
		- The city of the inventor is fixed → The city of the inventor may vary with the patent	The city of the inventor may vary with the patent
		- Two distinct attributes for Firstname and Lastname → Just one attribute Name of the form *Firstname#Lastname*	Two distinct attributes for Firstname and Lastname

2b) Global conceptual schema



2c) Conceptual to logical translation

Patent (<u>PatentId</u>, Title, GrantDate, Abstract, CPCCategory*)
PatentIPCCategory(<u>PatentId</u>, <u>IPCCategoryName</u>)
City (<u>CityName</u>, Country)
Assignee (<u>AssigneeId</u>, Name)
Inventor (<u>InventorId</u>, Firstname, Lastname)
PatentAssigneeCity (<u>PatentId</u>, <u>AssigneeId</u>, CityName)
PatentInventorCity (<u>PatentId</u>, <u>InventorId</u>, CityName)
Citation (<u>CitingPatent</u>, <u>CitedPatent</u>)

3. Query answering and mapping definition

3a) Query formulation

Find the pairs (id of the patent, name of the assignee) associated with assignees from Milan and patents granted in 2017.

SELECT P.PatentId, A.Name
FROM Patent **AS** P, PatentAssigneeCity **AS** PAC, Assignee **AS** A
WHERE P.PatentId=PAC.PatentId **AND** PAC.AssigneeId=A.AssigneeId **AND** PAC.CityName=’Milan’ **AND** P.GrantDate **BETWEEN** ‘2017-01-01’ **AND** ‘2017-12-31’

3b) GAV mapping definition

```
CREATE VIEW UniPoliPatents.Patent (PatentId, Title, GrantDate, Abstract, CPCCategory) AS (
        SELECT PatentId, Title, GrantDate, Abstract, CPCCategory
        FROM PoliPatents.Patent

        UNION

        SELECT PatentId, Title, GrantDate, Summary, null
        FROM UniPatents.Patent
)


CREATE VIEW Assignee (AssigneeId, Name) AS (
        SELECT KeyGenAssignee(AssigneeId, 'PoliPatents'), Name
        FROM PoliPatents.Assignee

        UNION

        SELECT KeyGenAssignee(AssigneeId, 'UniPatents'), Name
        FROM UniPatents.Assignee
)


CREATE VIEW UniPoliPatents.PatentAssigneeCity (PatentId, AssigneeId, CityName) AS (
        SELECT PA.PatentId, KeyGenAssignee(A.AssigneeId, 'PoliPatents'), A.CityName
        FROM PoliPatents.PatentAssignee AS PA, PoliPatents.Assignee AS A
        WHERE PA.AssigneeId=A.AssigneeId

        UNION

        SELECT PatentId, KeyGenAssignee(AssigneeId, 'UniPatents'), AssigneeCityName
        FROM UniPatents.Patent
)
```

3c) Query rewriting

```
SELECT P.PatentId, A.Name
FROM PoliPatents.Patent AS P, PoliPatents.PatentAssignee AS PA, PoliPatents.Assignee AS A
WHERE P.PatentId=PA.PatentId AND PA.AssigneeId=A.AssigneeId AND A.CityName='Milan' AND
P.GrantDate BETWEEN '2017-01-01' AND '2017-12-31'

UNION

SELECT P.PatentId, A.Name
FROM UniPatents.Patent AS P, UniPatents.Assignee AS A
WHERE P.AssigneeId=A.AssigneeId AND P.AssigneeCityName='Milan' AND P.GrantDate BETWEEN '2017-
01-01' AND '2017-12-31'
```