Phase 1 Report: DBMS Installation and Working Dataset

Shanshan Zhang, tuf14438@temple.edu

I. Data Description

The conference chosen for analysis is CIKM in 3 consecutive years.

In the preliminary analysis phase, there should be at least individual tables for the following entities: **Papers, Authors, PCMembers** for program committee members. A paper has columns like title, author, year, track (*IR, KM or DB*), topic, paper number, type (*full, long, demo, poster*). An author has columns like first name, last name, affiliation. A committee member has the same columns with an author. While when analyzing in depth, there are following relationships exist among these entities:

- Many-to-many relationship: 1) one paper is co-authored by multiple authors and one author can write multiple papers. 2) one PC member may appear in multiple conferences and one conferences have multiple PC members.

To address the two many-to-many relationship, I added a table called **Paper_Author** which stores only the paper-author pairs and using the PaperID and AuthorID as the foreign keys. And another table called **PC_Conf** table which stores the PCMember-conference pairs and using **PCMemberID** as the foreign key.

There then comes another question: where to put the year/conference information. For papers, one paper can appear in only one year, while for a committee member, he/she can appear in several years.

I considered three ways to add the year/conference information:

1. Add a column called year to **Papers** table and **PC_Conf** table.
2. Add a column called year to **Paper_Author** table and **PC_Conf** table.
3. Add a table called **Conferences** with two columns conferene name and year. The primary key contains the two columns and the two columns are also added to **Paper_Author** and **PC_Conf** tables as froreign keys.

Either the three is enough for my application, because I chose the same conference in 3 consecutive years. While finally I chose strategy 3 for the following reasons:

1. More efficiency for some query task. For example, if I want to analyze how many papers are published by every author in a specific year, I can only refer to the **Paper_Author** table without any other merging or joining operations.
2. Scalability. If later on I need to parse more conferences, strategy 3 will be more scalable since what I need to do is adding more rows in every table without altering the structure of tables.

One last minor consideration is whether there is a need to separate **Authors** and **PCMembers** table because they have exactly the same columns. So far, I didn't see any hurt of separating them, so I will keep them separated now.

II. Tables

```
mysql> show tables;
+--------------------+
| Tables_in_PubWorld |
+--------------------+
| Authors            |
| Conference         |
| PCMembers          |
| PC_Conf            |
| Paper_Author       |
| Papers             |
+--------------------+
6 rows in set (0.01 sec)
```

```
mysql> describe Authors;
+-------------+--------------+------+-----+---------+-------+
| Field       | Type         | Null | Key | Default | Extra |
+-------------+--------------+------+-----+---------+-------+
| AuthorID    | int(11)      | NO   | PRI | NULL    |       |
| FirstName   | varchar(255) | YES  |     | NULL    |       |
| LastName    | varchar(255) | YES  |     | NULL    |       |
| Affiliation | varchar(255) | YES  |     | NULL    |       |
+-------------+--------------+------+-----+---------+-------+
4 rows in set (0.00 sec)
```

```
mysql> describe Conference;
+----------+--------------+------+-----+---------+-------+
| Field    | Type         | Null | Key | Default | Extra |
+----------+--------------+------+-----+---------+-------+
| ConfName | varchar(255) | NO   | PRI | NULL    |       |
| Year     | year(4)      | NO   | PRI | NULL    |       |
+----------+--------------+------+-----+---------+-------+
2 rows in set (0.00 sec)
```

```
mysql> describe Papers;
+---------+--------------+------+-----+---------+-------+
| Field   | Type         | Null | Key | Default | Extra |
+---------+--------------+------+-----+---------+-------+
| PaperID | int(11)      | NO   | PRI | NULL    |       |
| Title   | varchar(255) | NO   |     | NULL    |       |
| PaperNo | varchar(255) | NO   |     | NULL    |       |
| Track   | varchar(255) | YES  |     | NULL    |       |
| Topic   | varchar(255) | YES  |     | NULL    |       |
| Type    | varchar(255) | YES  |     | NULL    |       |
+---------+--------------+------+-----+---------+-------+
6 rows in set (0.00 sec)
```

```
mysql> describe PCMembers;
+-------------+--------------+------+-----+---------+-------+
| Field       | Type         | Null | Key | Default | Extra |
+-------------+--------------+------+-----+---------+-------+
| PCMemberID  | int(11)      | NO   | PRI | NULL    |       |
| FirstName   | varchar(255) | YES  |     | NULL    |       |
| LastName    | varchar(255) | YES  |     | NULL    |       |
| Affiliation | varchar(255) | YES  |     | NULL    |       |
+-------------+--------------+------+-----+---------+-------+
4 rows in set (0.00 sec)
```

```
mysql> describe Paper_Author;
+----------+--------------+------+-----+---------+----------------+
| Field    | Type         | Null | Key | Default | Extra          |
+----------+--------------+------+-----+---------+----------------+
| ID       | int(11)      | NO   | PRI | NULL    | auto_increment |
| PaperID  | int(11)      | NO   | MUL | NULL    |                |
| AuthorID | int(11)      | NO   | MUL | NULL    |                |
| ConfName | varchar(255) | NO   | MUL | NULL    |                |
| Year     | year(4)      | NO   |     | NULL    |                |
+----------+--------------+------+-----+---------+----------------+
5 rows in set (0.03 sec)
```

```
mysql> describe PC_Conf;
+------------+--------------+------+-----+---------+----------------+
| Field      | Type         | Null | Key | Default | Extra          |
+------------+--------------+------+-----+---------+----------------+
| ID         | int(11)      | NO   | PRI | NULL    | auto_increment |
| PCMemberID | int(11)      | NO   | MUL | NULL    |                |
| ConfName   | varchar(255) | NO   | MUL | NULL    |                |
| Year       | year(4)      | NO   |     | NULL    |                |
| Track      | varchar(255) | YES  |     | NULL    |                |
| Title      | varchar(255) | YES  |     | NULL    |                |
+------------+--------------+------+-----+---------+----------------+
6 rows in set (0.00 sec)
```

III. Scripts

```
################################################
# @Author: Shanshan Zhang
# @Date: 09/16/2014
# @Class: Principle of Data Management
# @Title: MySQL script for Phase 1.
################################################

CREATE DATABASE PubWorld;
USE PubWorld;
SHOW TABLES;

CREATE TABLE IF NOT EXISTS Conference
(
ConfName VARCHAR(255) NOT NULL,
Year YEAR(4) NOT NULL,
PRIMARY KEY (ConfName, Year)
) ENGINE=INNODB;

CREATE TABLE IF NOT EXISTS Papers
(
PaperID INT NOT NULL,
Title VARCHAR(255) NOT NULL,
PaperNo VARCHAR(255) NOT NULL,
Track VARCHAR(255),
Topic VARCHAR(255),
Type VARCHAR(255),
PRIMARY KEY (PaperID)
) ENGINE=INNODB;

CREATE TABLE IF NOT EXISTS Authors
(
AuthorID INT NOT NULL,
FirstName VARCHAR(255),
LastName VARCHAR(255),
Affiliation VARCHAR(255),
PRIMARY KEY (AuthorID)
) ENGINE=INNODB;

CREATE TABLE IF NOT EXISTS PCMembers
(
PCMemberID INT NOT NULL,
FirstName VARCHAR(255),
LastName VARCHAR(255),
Affiliation VARCHAR(255),
PRIMARY KEY(PCMemberID)
) ENGINE=INNODB;
CREATE TABLE IF NOT EXISTS Paper_Author
(
```

```
ID INT NOT NULL AUTO_INCREMENT,
PaperID INT NOT NULL,
AuthorID INT NOT NULL,
ConfName VARCHAR(255) NOT NULL,
Year YEAR(4) NOT NULL,
PRIMARY KEY (ID),
INDEX (PaperID),
INDEX (AuthorID),
INDEX (ConfName, Year),
FOREIGN KEY (PaperID)
    REFERENCES Papers (PaperID)
    ON UPDATE CASCADE ON DELETE RESTRICT,
FOREIGN KEY (AuthorID)
    REFERENCES Authors (AuthorID)
    ON UPDATE CASCADE ON DELETE RESTRICT,
FOREIGN KEY (ConfName, Year)
    REFERENCES Conference (ConfName, Year)
    ON UPDATE CASCADE ON DELETE RESTRICT
) ENGINE=INNODB;

--
CREATE TABLE IF NOT EXISTS PC_Conf
(
ID INT NOT NULL AUTO_INCREMENT,
PCMemberID INT NOT NULL,
ConfName VARCHAR(255) NOT NULL,
Year YEAR(4) NOT NULL,
Track VARCHAR(255),
Title VARCHAR(255),
PRIMARY KEY(ID),
INDEX (PCMemberID),
INDEX (ConfName, Year),
FOREIGN KEY(PCMemberID)
    REFERENCES PCMembers(PCMemberID)
    ON UPDATE CASCADE ON DELETE RESTRICT,
FOREIGN KEY(ConfName, Year)
    REFERENCES Conference(ConfName, Year)
    ON UPDATE CASCADE ON DELETE RESTRICT
) ENGINE=INNODB;
```