

Distributed Database for the Department of Education with MySQL and PHP

Tenzin Choesang

tchoesang@unibz.it

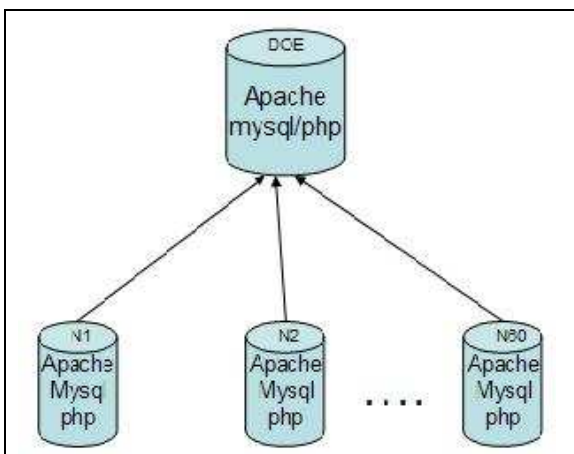
Problem Definition:

The Department of Education (DoE), Tibetan Government in exile wants to have a up to date data on the 60 different schools that are spread around 3 countries (India, Nepal and Bhutan) with approximately 30,000 students and 8000 staffs. Due to the lack of structured communication channels between the schools and the Department of Education, the data kept by DoE is not consistent and unreliable and many times too old. It is also important to securely transfer data from these schools to the DoE at fixed intervals of time during the year. Keeping in mind that most of the schools have computers but it is highly likely that many remote area schools are not connected by internet. These schools lack staffs with computer expertise to maintain any kind of sophisticated system or to conduct complicated tasks.

Up to date information is very much needed for

1. The management to take future decisions
2. To see ongoing progress of schools, students and staffs
3. Digitalised archive of history on each school and individuals (students and staff)
4. In Exile there are approximately 100,000 Tibetans and from them about 35-40% are related to the Educational institutions, which makes it a pressing need of the department

Figure 1: Structural overview of the Distributed Database System:



Problem Solution:

The idea was to use the resources each school has and to present a practical solution with keeping security of the data in mind. Due to the constraints regarding the budget on the project allocated until now and the time frame for the project, one of the most suitable solution was to use free open source software's (Apache as a web server, MySQL server as a backend database and php as a server side language). These three programs will be installed on each machine in every school. Considering the complexity of configuring Apache, MySQL and php, the solution is to use XAMPP (<http://www.apachefriends.org/en/xampp.html>).

It is a free packaged preconfigured Apache, MySQL, php bundle, which simply needs to be installed.

Each school communicates to its own local database via a web browser, so they can insert, delete, update and select data via the web browser. At every fixed point they make a dump file of the data that are not updated to the DoE database. This action is performed by first creating a duplicate of each table and inserting them with all the data that has not been updated to the DoE until that moment. Which is known from a column called copied (Boolean value), if copied 1 else 0. After the creation of new tables and insertion of non updated data, this new table is dumped into file using mysqldump command.

```
mysqldump -u %mysqluser% -p%mysqlpassword% node student_dump%workdir%\student_dump.sql
```

This is done via a batch file, which creates the dump file, zips the file and encrypts it. This data is now ready to be dispatched to the DoE.

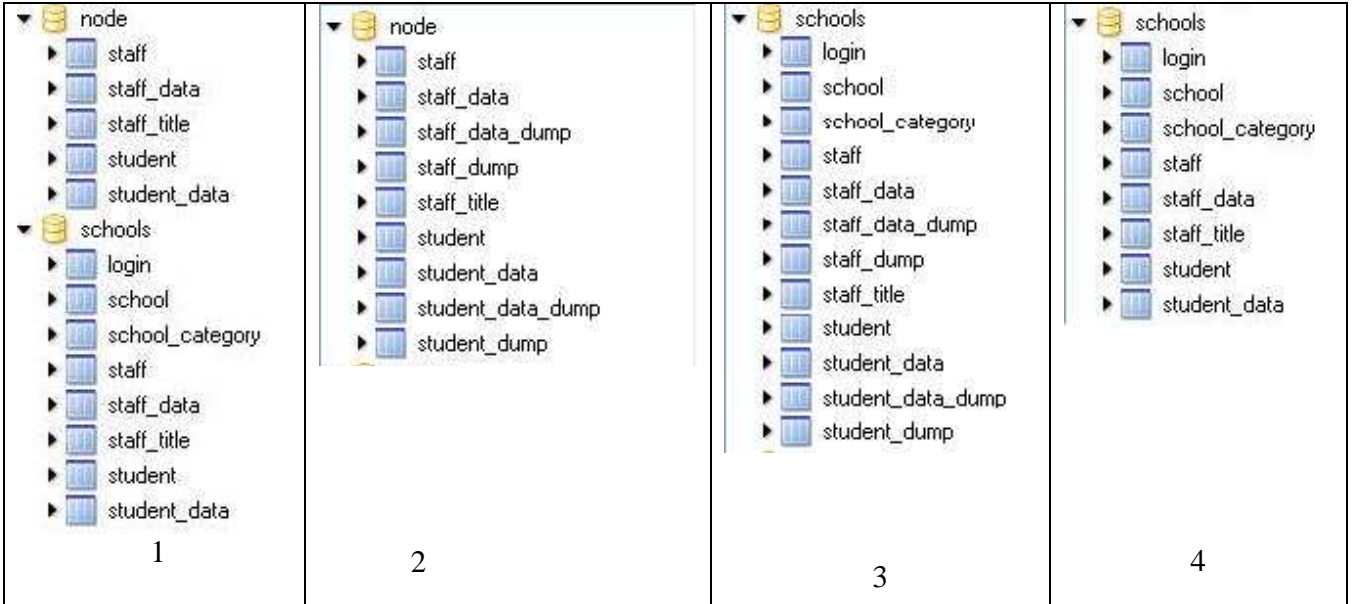
On the other side, the DoE, decrypts the data, unzips it and the data is updated in its database. This action is also performed through a batch file, which recreates these dump table from the dump file.

```
mysql -u %mysqluser% -p%mysqlpassword% schools <%workdir%\student_dump.sql
```

The data from the dump file is now inserted into the final table and the dump file is dropped from the DoE database.

That way, it avoids the need of a constant internet connection to the central database, complexity of the task is kept to minimum and security of the data is constantly maintained.

Figure 2: A step by step actions performed on the Databases:



1. Node DB: school database and schools DB: Department of Education DB. Tables in the DB normally.
2. Node DB: 4 new tables are created with TableName_dump. These new tables contain a copy of the non updated data.
3. Schools DB: After the DoE receives the dump files, it imports the dump file in its DB, which creates the 4 new tables with the data in it.
4. Schools DB: After the data has been inserted into the final tables, the dump files are dropped.

Figure 3: Entity Relationship diagram of the Central Database (DoE)

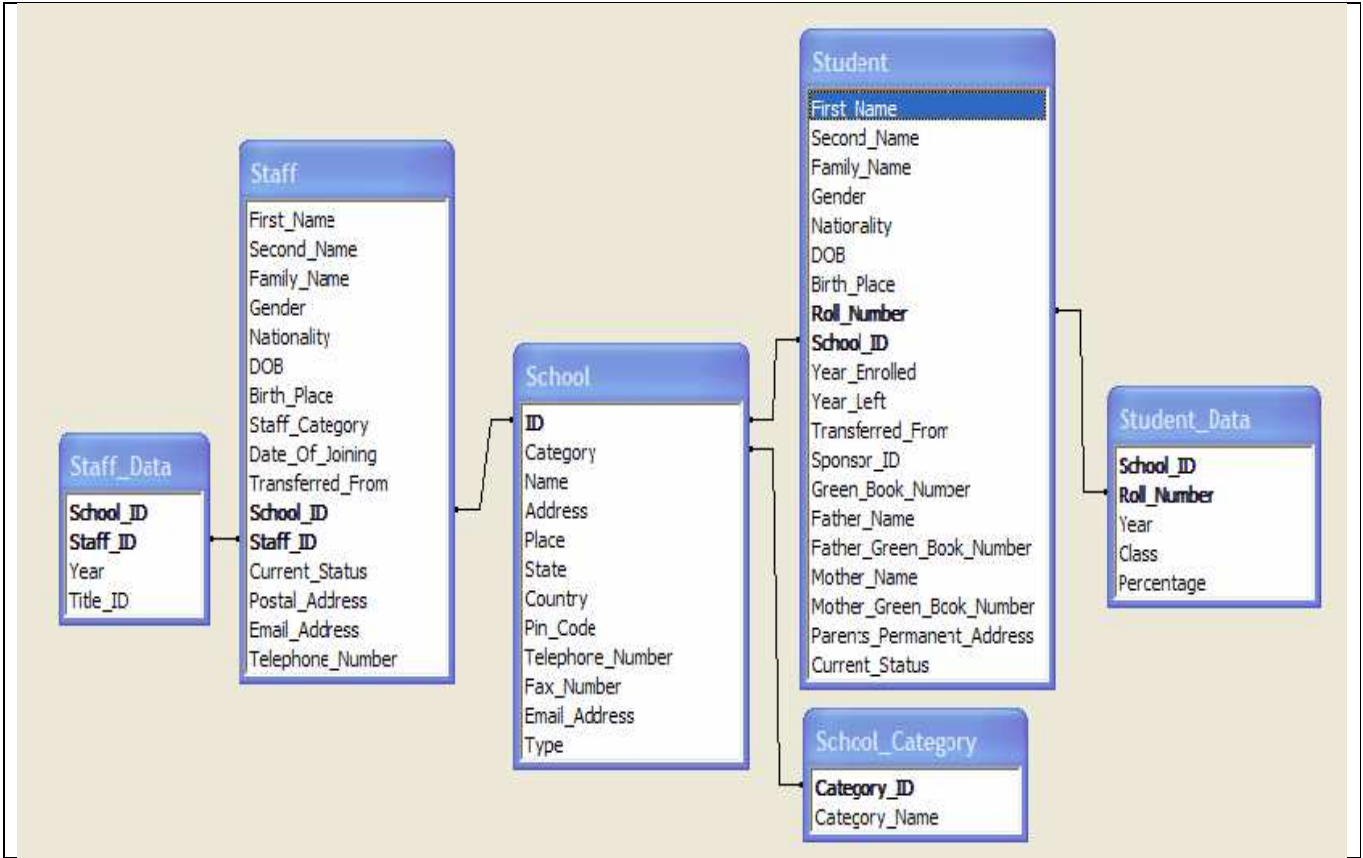


Figure 4: Entity Relationship diagram of Node Database (School):

