1 Purpose of the project

You are to analyze the requirements, design, implement, document, and demonstrate a database system that could answer a wide class of queries on the history of the FIFA World Cup.

2 Overview

The main goal of this project is to populate a relational database, called MondialDB, from data readily available on the Web. In this project you will learn how to transform web HTML and XML data into a relational database that supports SQL querying and processing. The real challenge in this project is to learn how to select from hundreds or thousand possible sites and integrate the data into a single relational database.

2.1 The MondialDB

The MondialDB allows the users to retrieve data about the city that organized the World Cup. From now on we will use the year it took place to refer to specific “cup”, such as 1958, 1972, 1990, 1994. MondialDB will have data about the national teams that participated, the scores in the these soccer games, the players in each team, the scorers, the Cup champions, the rank each team obtain, statistics for each team i.e. wins, losses, draws (ties), points, and other related information.

The games are organized in groups first. Then, in most of the Cups, the teams that qualify from each group play in a tournament where at each step one team advances to the next round and the other is eliminated from the Cup. You can search the web to find out more if you do not know on the organization of the Cup.

2.2 Tasks & Queries

This section gives only a few of the tasks and queries. These are suggestions but you need to expand, modify, and/or substitute some of them with other more interesting ones depending on the emphasis of your project.

- Extract-Transform-Load (ETL) Populating the database (data entry) is an important task that you have to do. This is the task that is responsible for extracting the data from the web, transforming it to match the format of your database schema, and loading it in. The data will come from the web which most likely will be in the form of HTML or XML documents. Therefore, parsing such documents is an essential aspect of the project and will give you a unique experience that will be useful in the increasing number of databases that depend on data coming from the web. Automatic data entry of such documents is necessary.
You should break this task in several subtasks for dealing with different types of data, sources, and, other peculiarities that you will be confronted. It is almost certain that you will need to do some “data cleansing” during the transformation and loading. One such cleansing would be the discovery and elimination of duplicate data coming from different sites, elimination of wrong data, etc. Another could be discrepancies in the form of the data. An example could be the variations of dates, European vs American, measurement units, etc. All these reconciliations are done during ETL.

- **Query for a given Cup:** Typically, we would like to have queries to return information on a given Cup, such as year and country, the participating countries, the position of each of the teams, scores, etc. In the result, goals scored during the regular game period have to be reported separately from those scored during penalty kicks (the tie breaker).

- **Query for a given Player:** This query will return a player’s participation in all Cups, the country he paid with, goals he scored, age he was and perhaps some other interesting data. For example, by entering the player Eusebio, it should return that he played for Portugal in 1966 and was the highest scorer with 9 goals.

- **Query for Super Stars:** This query will find, if any, all the players that played in more than a Cup and the goals they scored in each of them.

- **Team Historical Query:** This query should return the results of a team throughout the history of the games along with the position it obtained in a given Cup. Along with it, some statistics about the performance of the team: the total number of times the team participated, the total number of goals scored/received, points obtained, etc.

- **Country’s Players Query:** This query should produce for each country the scorers in all participation, the number of goals the scored during the normal and penalty shout-outs.

- **Misc Queries:** You come up with 4 additional “interesting” queries on your own (subject to the approval of the TA).

There are many directions that you can explore to enhance your project for extra credit. For example you can enrich your database with urls to bring in audio-visual tools and display picture and/or play audio stored in the database or elsewhere on the web. Under no circumstances the extra credit can exceed 20% of the projects highest score or 6% of the total grade.

### 3 Rules of the game

- **Groups:** The project is to be done in groups of 2 students. A roster for each group must be submitted to the TA by the date specified in the “Due Dates” section of class schedule. The groups are “self-policing” (e.g., each group is responsible for its own division of labor, scheduling, etc.). *Note: If an irreconcilable problem arises in your group, it is your responsibility to contact both the professor and the TA as soon as possible. After the project is due, it will be too late.*

- **Assumptions:** In cases where you have questions on the above description, it is acceptable to make assumptions about the application providing that: 1) they are explicitly stated in the report, 2) they
don’t terribly conflict with any of the requirements specified above, and 3) they are “reasonable”. If you have a question about the acceptability of any of your assumptions, check with the TA or the professor.

- **Reports:** A report should be handed in for checking at the end of each phase. The report must be formatted in a reasonable manner (i.e., using a text processor and a decent printer). Reports are due during class on the date specified in the “Due Dates” section below.

- **Implementation:** The final phase of the project requires a working implementation of the system to be built, tested, and demonstrated. A large part of the project grade depends on the quality of this implementation. The implementation will be done as a client-server system in which a web server runs on your cluster unix account or another machine, accepts queries through a web interface, and connects to a DBMS to retrieve from the database. On the unix cluster, you will be using your student login account. ORACLE accounts on the cluster will be provided. If you like, you can use for the project mySQL running on any machine you have access and can set it up on your own. Unfortunately, the class cannot provide assistance in installing your own DBMS. *A small portion of the project grade will be based on the Web-based user interface. A bigger portion of the grade will depend on the quality of data entry which will assure robustness of the database.*

## 4 Project Phases

The three phases of the project cover the following work-processes from the paper *An Adaptable Methodology for Database Design* [1]:

<table>
<thead>
<tr>
<th>Phase</th>
<th>Phase Name</th>
<th>Due Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Group names to the TA</td>
<td>Sep 22, 2011</td>
</tr>
<tr>
<td>II</td>
<td>Conceptual Modeling and Task Emulation</td>
<td>Nov 10, 2011</td>
</tr>
<tr>
<td>III</td>
<td>Implementation and Testing</td>
<td>Dec 13, 2011</td>
</tr>
</tbody>
</table>

## 5 Reports

*The Phase I report must contain:*

1. a description of the purpose of the project and of this phase.

2. a description of the technical/conceptual problems encountered in this phase and justification for the solutions.

3. the assumptions that you have made about the enterprise.

4. a description of the procedures in the enterprise, as you imagine they happen.

5. all the documentation produced in this phase, i.e.

   - the top-level information flow diagram, *(very important)*
the list of tasks, subtasks, and the task forms
the list of documents and their form (important)

The Phase II report must contain:

1. Phase I- with corrections addressing the TA’s feedback.
2. a short description of the purpose of this phase of the project.
3. a description of the problems encountered in this phase and justification for the solutions.
4. the documentation produced in this phase, i.e.,
   • the graphical schema using the E-R model,
   • list of the attributes for each entity and relationship,
   • the relational schema obtained by mapping the E-R to relations, and their Boyce-Codd or 3rd Normal Form with keys.
   • the code for each task: pseudo-code and the embedded DML code.

The Phase III report must contain:

1. Phase I and Phase II reports with corrections addressing TA’s feedback.
2. a description of the purpose of this phase of the project,
3. a description of the problems encountered in this phase and justification for the solutions.
4. any revisions made to the relational schema definition from Phase II,
5. documentation produced in this phase, i.e.:
   • a source program listing.
   • a users manual for the system.
   • your testing efforts: erroneous cases that your system can detect and handle reasonably.
   • a description of the system’s limitations and the possibilities for improvements.
6. In addition, a demo of the system is required. All members of the group should attend this demo, to explain the aspects of the project for which they were responsible.

References