**The Association of Computing Machinery (ACM) Database**

**Project Member**

Caleb Hamilton

Step : Create an imaginary scenario. Your scenario should satisfy following conditions:

The Association of Computing Machinery (ACM), one of the largest computing associations in the world, is having difficulty maintaining its membership information because its information system is old and no longer reflects its growing needs. The following is a typical national organizational hierarchy of the ACM in the US.

a. One-to-one binary relationships.

Coordinator has a 1 to 1 relationship with sub\_entity

“This includes different types of sub\_types, and ACM coordinator title, since it was hard to implement elsewhere”

Coordinator has a 1 or 0 to 1 relationship with itself

“ACM coordinator in charge of the rest”

b. One-to-many binary relationships.

Country Chairman has a 1 to Many relationship with Coordinator

“1 country for all the coordinators”

Coordinator has a many to 1 relationship with STATECHAPTER

“Since you can only be in 1 state”

State Chapter has a 1 to many relationship with City Chapter

“This is because many cities are in a state”

c. Associative relationships.

Member is an associative relationship between Sub Entity and City chapter, with a many to 1 relationship with city chapter, and a many to 1 or 0 relationship with sub entity

“The reason for 1 or 0 is because not all members have to be a part of a subtype, but there can be many members in a certain city or type”

Step : Explain the story behind the scenario, and all your assumptions, which are required to support relationships given above.

In this scenario, the ACM is big organization with a complex leadership structure. The organization currently has chapters in 17 states. These are Texas, Tennessee, Missouri, Nebraska, Minnesota, Iowa, South Dakota, North Dakota, New York, District of Columbia, Alaska, Maine, Washington, Colorado, Arizona, California, and Utah. To be an ACM member, you apply for ACM membership in the city or state where you reside. Once membership is granted and membership dues are paid, you have a full ACM membership with all its privileges. In addition, it is possible to apply for a sub-membership. There are four sub-membership types available. These are student, youth, veteran, and women. Each of the four sub-entities has its own chapters in the 17 states as well. Because each sub-membership comes with its perks, there are additional membership subscription fees to become a member, but you must be a full ACM member before you can qualify for a sub-membership. Also, some members are employed in the offices of the Representative and the Chapter Coordinator and may be required to pay occasional minimal dues. Reporting mechanism are upward. This means the lowest entity reports to the one above it. To be specific, ACM city chapters report to their state chapter leadership. For instance, state leaders for students, youth, women, and veteran chapters report to their national coordinators; ACM state chapter chairman/chairwomen reports to the ACM Chapter Coordinator; the ACM Chapter Coordinator reports to the Country ACM Representative – who also receives reports from the student, youth, women, and veteran Coordinators. However, other relationships – aside from reporting, are possible both ways. ACM maintains the following data (attributes) from each member: Full name, ID number, position, address, age, country, state, county, citizenship, Other Citizenship, tribe, region, date of birth, date of joining, university attended, highest degree earned, veteran status, marital status, whether they work in the office of the Representative or not, whether they work in the office of the Chapter Coordinator or not, and submembership.

Step :

Immediately I went to design a diagram to better understand what kind of relational database this would be, in doing so I came across a lot of challenges to successfully create an ER diagram that I was satisfied with.

Step : Show ER..

Diagram

Description automatically generated

Step :

One the diagram was made I put everything into MYSQL to create the relational database.

Step :

I then forward engineered the database into my local MYSQL node so I could manipulate data in the database.

Step :

I started with the chairman to add his data, since he isn’t reliant on anyone else, and seemed like a logical start.

Step :

I then added all the states that were mentioned in the above text, to the STATECHAPTERS table, since this is also a non-reliant node.

Step :

Once the data was added to the two nodes with no reliance, I added data to my coordinators table, since they are only reliant on themselves, and the state and chairman, so it was possible to make a head ACM coordinator, as well as coordinators for each different sub type for a specific state.

Step :

Once the coordinators are added, it is possible to now interact with the sub entity table, where we can specify the types of sub relationships being youth, veteran, women, and student, and apply which coordinator is in charge of which type.

Step :

Before we can mess with the members, we need to add the different city chapters to the state, none were given, so I simply went with a few cities in Texas as my example in the database.

Step :

Once all other data is added, the Members table can be filled out with data, since it is an associative relationship between the city chapters and the sub types.

Step :

Another important part of the members table is that it shows whether or not someone is working for a ACM coordinator, or for a city chapter, so a position column was added to show whether or not someone is working for the organization or simply a member.

Step :

Once all data is filled out you can begin to ensure quality of the database, making sure all tables are properly linked and have proper references to eachother.

Step :

Once all tables are quality inspected, we can begin to write MYSQL code to interact with the database.

Step :

The SQL code written, comes directly from the objectives of the company (which were cited in the requirements document), and make simple operations to receive the data necessary.

Step :

Once this document is tested with the MYSQL code, we can consider the database finished and eligible to be used by the company in their future projects.

Step :

If ever the company wants to come in and change any data, or add any new relationships, this could be done very simply by adding data via MYSQL code, or going into the database and manually adding new data.

Step 19:

If a new relationship is to be added, then the whole database must be forward engineered again, and start from Step 6. Once this is done, we should be able to transfer most data we have into the new database, and just fill in the gaps where we are missing from the new relationships.