# T-202-GAG1: Project 4

## Readings

Ramakrishnan & Gehrke:  Chapters 2 and 3.

## Project Outline

*Disclaimer: The description of this project is entirely fictional.*

A political party, the Wily And Sinister Party or WASP, has found its support waning and decided to create a database to support its operations. The WASP have already hired consultants to interview potential users, coming up with the following requirements for the database, but they need you to design their database by creating an ER-diagram and transforming it into SQL tables.

## Database Requirements

The following requirements were determined through interviews with WASP:

1. For people, the database should keep track of their ID, name, address, phone number, date of birth, and date of death (default value is NULL).

2. People are further divided up into members of WASP and enemies; each person is either a member or an enemy, but not both. For members, the starting date of membership is registered. Note that WASP members cannot leave the party, even in death!

3. For each WASP enemy a list of their weaknesses is maintained, in case it could be used against them to achieve the WASP agenda. Each weakness of a person is identified with the name of the weakness, but text fields must also be maintained to a) describe the weakness in more detail and b) describe how the weakness could potentially be used against them.

4. For each WASP member a list of children is maintained. For each child of a particular member, the name is the identifier. Furthermore, data of birth and phone number of the child are registered, as well as a rating of their willingness to help WASP.

5. Furthermore, relationships between various people (e.g. business partnership, marriage, mortal enemies, …) are registered in a fairly flexible way. We call these "linkings" between people to avoid confusion with the relationship concept of ER-diagrams. Each such linking is considered an entity and assigned an ID, name, type and a more detailed description. Multiple people may then participate in each such linking, although the minimum number of people per linking should be two, and each person may participate in multiple linkings with different people.

6. The WASP party has a list of roles, each with a unique title. (Roles include party roles, such as chairman, and external roles, such as election candidate or senator.) Members take turns filling the roles: each member may fill multiple roles at any given time and multiple members may fulfill the same role. Members may, however, only be appointed to the same role once. The start and end dates of each appointment to a role are always known in advance and recorded, as well as the monthly salary for the role.

7. The WASP party also has several committees, identified by their names. For each committee, a text description of its agenda is stored, as well as the date the committee was established. Each committee may have any number of members, who of course must be WASP members, and each WASP member may serve on any number of committees. For each committee member, the start and end date of the appointment must be stored, as well as their role in the committee (chair, ...).

8. The WASP maintains a list of sponsors. For each sponsor, an ID is registered, as well as their name, their address and their industry.

9. Each sponsor may sponsor a number of people, but these people must serve in some role for the party to be eligible for sponsorship. For each sponsorship the date of the sponsorship is registered (and must fall within the start and end dates of the member's appoint to that role), as well as the amount and a text field called "payback" that describes what the sponsor expects in return. Each sponsor may sponsor a member in a particular role multiple times, with different amounts and different expected paybacks, but only once per day.

10. Each sponsorship is reviewed by one member of WASP. The date of the review is decided when the sponsorship is registered, typically one year in the future. At review time, the payback is assigned a numerical grade from 1 to 10, depending on how well the WASP member executed the payback.

## Project Restrictions

The database design should follow the requirements definition strictly. For example, ID columns should only be used where specifically mentioned. Where ID columns are mentioned, however, you can assume that they are a key.

There are requirements, however, which cannot be specified in the ER-diagram. Some of these can be addressed in the database creation script, while others may require more advanced techniques, such as assertions (which are not available in Postgres). Part of the project is to realize and document such restrictions in your report.

# Project Deliverables

*The project is a group project, with three (or at minimum two) students per group. The deadline is at 23:59 on Thursday October 20.*

**Late submissions will not be accepted, so make sure to submit your solutions on time. And note that one-student "groups" are not allowed!**

Submit three files:

a) A PDF file containing a complete ER-diagram. The diagram must follow the notation presented in the R&G book and in lectures, and must not be hand-drawn.

   The ER-diagram is the main deliverable of the project, as most of the entities and relationships are captured there. In order to fully specify the database design, however, the remaining two documents are also needed.

b) A text file (.txt) containing SQL commands to create the corresponding database tables.

c) A PDF file with a short report, which a) enumerates assumptions you have made that you felt were not clearly specified in the project description or cleared up in discussions on Piazza; b) enumerates all important design decisions made when converting the ER-diagram to tables; and c) enumerates all requirements which are *not* represented in the ER-diagram, specifying whether and how they can be supported in the database implementation.

   The report should have the title "Database Design for WASP", have your names, a short introduction of the contents, which should mention which software or web-site you used to generate the ER-diagram, and then three sections, one for each of the above enumeration.

Correctly supporting each of the 10 requirements above, or showing why (some part of it) cannot be supported, gives 10% of the final grade for the project. It is possible, however, to lose points due to unclear presentation in any one of the three files.