Marist College

Lab 9
## Normalization 3

Dayna Eidle
Database Management
Professor Alan Labouseur
Due Date: November 20, 2017

1. *Functional dependencies*

   - **People**

     pid → firstName, lastName, age, spouseId

   - **Astronauts**

     aid → yearsFlying, golfHandicap

   - **Engineers**

     eid → highestDegree, favoriteVideoGame

   - **FlightControlOperators**

     oid → chairPreference, preferredDrink, hangoverCure

   - **Spouses**

     spouseid →sid

   - **Crew**

     aid, scid →

   - **Spacecrafts**

     scid → name, tailNum, weightInTons, fuelType, crewCap

   - **SpacecraftSystems**

     scid, sysid →

   - **Systems**

     sysid → name, description, costUSD

   - **SystemParts**

     sysid, partid →

   - **Parts**

     partid → name, description, costUSD

   - **Suppliers**

     supplierid → name, address, paymentTerms

   - **Catalog**

     supplierid, partid → description, costUSD

2. E/R Diagram
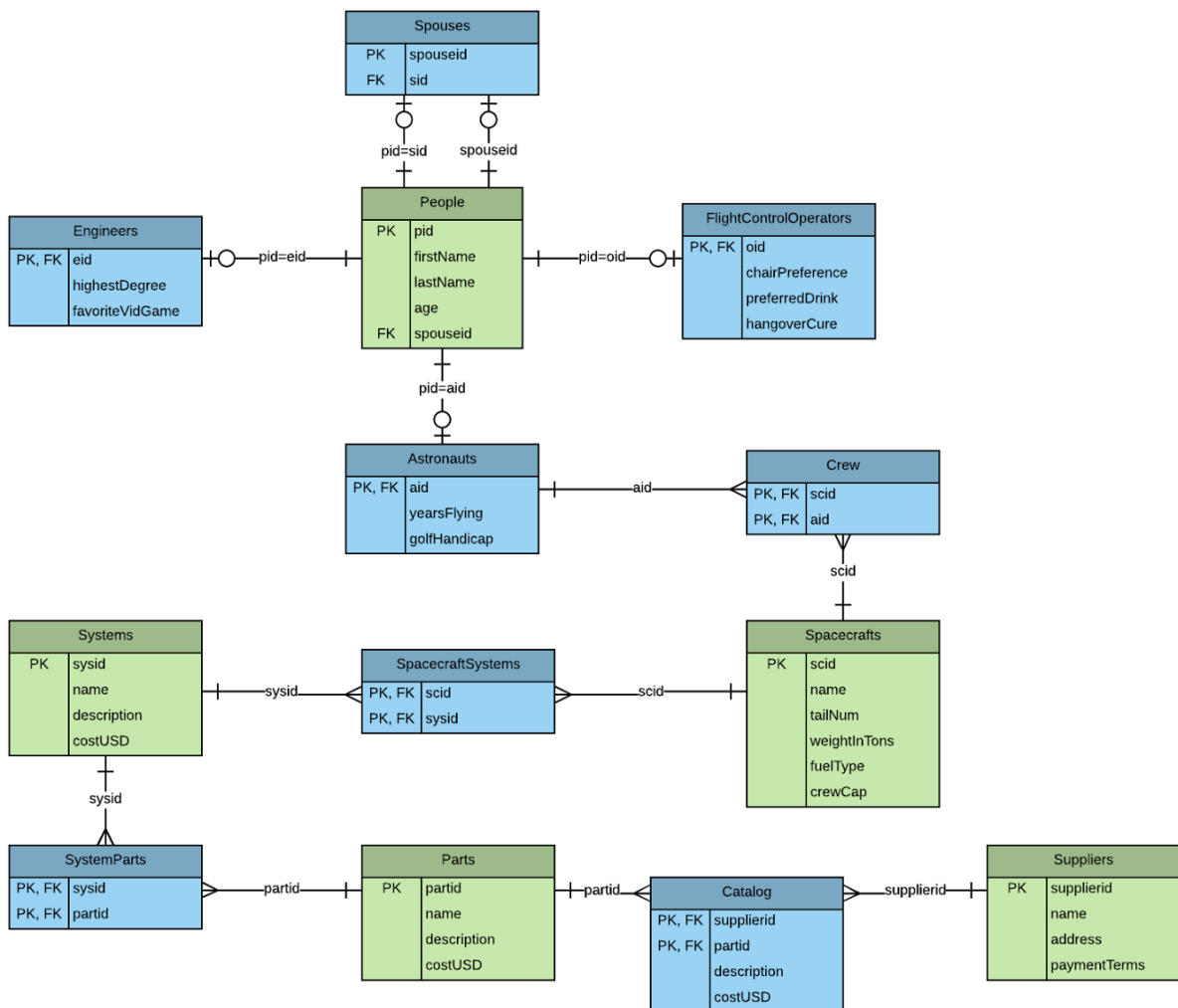
3. Explanation

In order for a database to be in 3NF, it needs to be in 1NF (atomic), 2NF (have no partial key dependencies), and have no multi-key dependencies. This database is atomic because every cell (cross section of a row and column) cannot be broken down into smaller elements. For example, rather than having just name of a person and

having to enter "James Bond" into one field, there is a column for first name and a column for last name so the name would be split as "James" and "Bond". This hold true for all other fields on the table as well. This database is in 2NF because it's in 1NF and there are no partial key dependencies among the tables. That means that every non-key attribute is dependent on the whole primary key of each table, rather than just a portion of it. An example of this can be found in the Catalog table. In this table, supplierid and partid are the composite primary key. The description and cost are dependent on both the supplierid and the partid. A part sold by two different suppliers would most likely have a different description for each supplier as well as the price which is why the description and cost depend on the whole primary key and not just one element of it. Finally, the table is in 3NF because it is in 1NF, 2NF, and has no multi-key dependencies. When a table has a multi-key dependency, it means that a non-key attribute depends on the primary key, in addition to some other non-key attribute. For many of the tables, there is an artificial key that is made to ensure uniqueness which almost makes it impossible to have a multi-key dependency in a situation like that. For the associative entities, there are composite primary keys that encompass all aspects of the table that are needed to ensure uniqueness so there would be no reason for the need of a multi-key dependency. With all of these factors combined, this database is in 3NF.