

Ryan Fredericks

Data Management

Normalization 3

### **FUNCTIONAL DEPENDENCIES**

- a. Staff
  - i. staffID -> fname, lname, age
- b. Engineers
  - i. staffID -> degree, videogame
- c. Flight Control Operators
  - i. staffID -> chair, drink, hangover
- d. Astronauts
  - i. staffID -> yearsFlying, handicap, spouse
- e. Crew
  - i. staffID, sid ->
- f. Spacecrafts
  - i. sid -> name, tailNum, weight, fuelType, capacity
- g. Systems
  - i. sysID -> sid, name, description, costUSD
- h. Parts
  - i. partID -> sysID, name, description, costUSD
- i. Catalog
  - i. partID, suppID ->
- j. Suppliers
  - i. suppID -> name, address, paymentTerms

### **Why this is in 3NF/Boyce Codd Normal Form**

This is in 3NF and Boyce Codd Normal form because there are no anomalies within the table, which means that if information is updated or deleted from one place, the same is done through all relevant tables. It is also in third normal form because every table has a primary key or candidate key that are unique, which means that there is no redundancies throughout the table. The staff, astronauts, engineers, and flight control operators tables have the same primary key because they are entity subtypes, which help to define people within the database and can be a useful tool to avoid having nulls within the table. There is always a connection between the primary key of the table and the attributes, which means that the functional dependencies are correct and there are correct relationships between the superkey and the non-prime attributes. For example, if a system is changed, those changes will be noted within the spacecrafts and parts table.

