Michael Wise
Dr. Alan Labouseur
CMPT 308 Database Management
19 November 2020

Lab #9 - Normalization Three

1. **Functional Dependencies:**

NASAEmployees Table
- nasaId → firstName, lastName, age, spouseFirstName, spouseLastName

Engineers Table
- nasaId → highestDegree, favVideoGame

Astronauts Table
- nasaId → yearsFlying, golfHandicap

FlightControlOperators Table
- nasaId → chairPref, preferredDrink, recHangoverCure

Spacecraft Table
- tailNum → name, weightTons, fuelType, crewCapacity

Systems Table
- systemId → name, description, costUSD

Parts Table
- partId → name, description, costUSD

Suppliers Table
- supplierId → name, address, ZIP, paymentTerms

ZIPAddress Table
- ZIP → city, state
    - We must create a ZIPAddress table to avoid the transitive dependency of supplierID → ZIP → city, state

Things to consider when dealing with cardinality and entity relationships:

Many suppliers can supply many parts, i.e. the same part can come from multiple (different) suppliers, often being interchangeable. If we considered the scenario in which each part can only come from one supplier, then we probably wouldn't need a Catalog table.

I created a NASAEmployees table because you can be an engineer, astronaut, flight control operator, a mix of the three, or none of them. Consider Clayton Anderson, who started as a NASA engineer and then became an astronaut later in his career[1]. Also, I think the janitors working there would feel left out otherwise.

[1] https://www.popularmechanics.com/space/interviews/a16978/clayton-anderson-interview/

Crew is an associative entity that deals with the many-to-many relationship between Astronauts and Spacecraft. Many astronauts have flown on more than one spacecraft, which usually coincides with being part of more than one crew/one mission (i.e. Chris Hadfield has been part of more than one “crew” / “mission” because he’s been to space multiple times and back, with different crews on different spacecraft.)

2. **E/R Diagram:**

Can be found as a separate pdf file on GitHub, because everyone deserves a high quality E/R diagram. When pasting it right into this document the quality was significantly affected.

3. **Explanation of why the database is in 3NF:**

This database is in 1NF because every intersection of a row and column is atomic. It is also in 2NF because it's in 1NF and there are no partial key dependencies. The two instances in which we have composite keys are in the Crew and Catalog table. In each of those tables, each part of the key doesn’t depend on another attribute in its respective table. In fact, there are no dependencies in those two tables. Finally, it is in 3NF because it is in 2NF and there are no multiple key dependencies. Every determinant is the key of a table. There are no non-prime attributes that determine other attributes. There are no transitive dependencies between non-key attributes or determinants, so this database is normalized.