

CDS 302 Assignment 2

Byron Washington

February 3, 2025

Problem 1

- Data independence is the concept of data structures being independent from their application programs to allow it to grow and mutate without harming the data or having to change the application programs themselves. This concept is important because if the data structure has to be changed, it would be very difficult and/or tedious to rework the entire application program just for a simple change. For example, if a user at a company had a folder with all their work but they left, and their folder gets deleted. All of that work and data within that folder would be gone because the data was dependent on the user being able to access their folder. Data independence also solves the problem of redundant and inaccurate data. If that user from before had worked in a different department of the same company previously, their file would show up in two spots. If someone wanted to access their data and it was different in either file, it would be nearly impossible to determine which data was accurate. Data independence solves this issue and is why most large databases are no longer managed with file-based systems.
- A key idea behind the relational model would be how the relations/tables are structured or how the schema looks like. Each column is called a tuple and each row is an n-tuple of a relation R. The schema is set up this way because it is based on relational algebra and set theory and it also allows for data independence. Another key idea behind the relational model would be the use of primary and foreign keys. The primary key is a domain or combination of domains that act as a unique identifier for the relation. A foreign key is a domain or combination of domains that are not the primary key of its own relation but the primary key for another relation.
- The relational model is an improvement on previous data models for many reasons. One of the biggest reasons would be data independence. Like said before, data independence is crucial for large datasets that are actively used and if the data structure needs to be changed at all, not changing the entire application program is a must. 3 big problems with older data

models are their inability to get rid of their ordering dependence, indexing dependence, and access path dependence. The relational model is able to solve all of these problems. Another reason would be readability. Many systems use a tree or graph structure which can make it hard to search for specific data when you have to follow the paths given. With the relational model, you are able to avoid this problem because you don't have to follow the said paths.

Problem 2

employee(person-name, street, city)
 works(person-name, company-name, salary)
 company(company-name, company-city)

1. $\Pi_{person-name}(\sigma_{city='Miami'}(employee))$
2. $\Pi_{person-name}(\sigma_{salary>100000}(works))$
3. $\Pi_{person-name}(\sigma_{city='Miami' \cap salary>100000}(employee \bowtie works))$