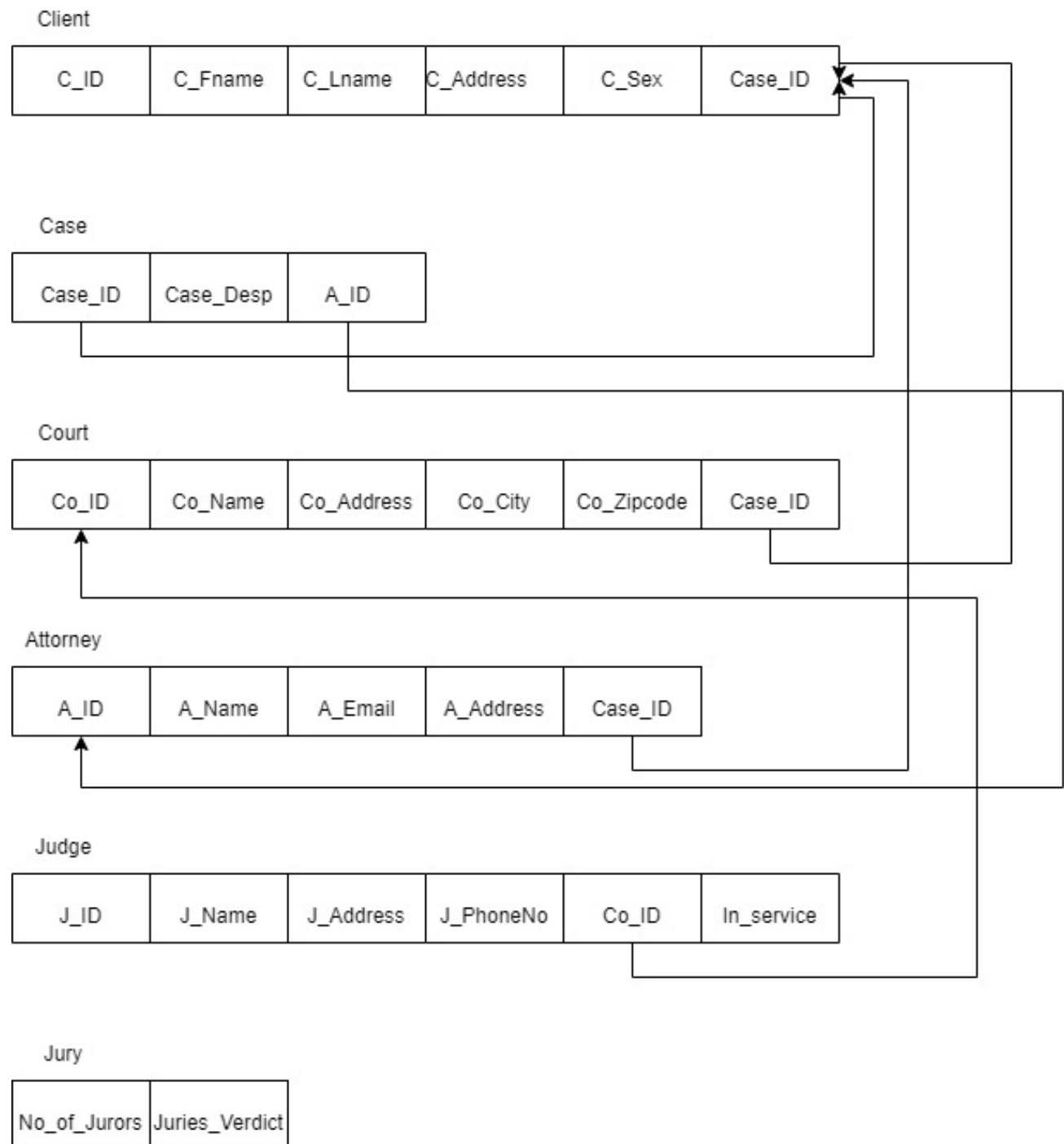


Court Case Management System

- S. Sandeep Srivatsava
- N. Sai Aravind
- Sourab Deb

Relational Schema:



Normalization:

Entities: • Clients • Case • Attorney • Court • Judge • Jury

Relationships: • Clients – Fname, LName, Address, Sex • Case – Case_Description, Case_Id • Attorney – Attorney_Id, Attorney_Name, Address • Court – City, ZipCode, Address, Case_Id, Court_Name • Judge – Judge_Id, Judge_Name, Address, Judge_PhnNo • Jury – No_of_jurors, Juries_verdict

Transactions: • Case Filing: Create a new case record in the database • Case Updating: Update existing records in the database with new information • Case Closing: To mark a case closed in the database • Case Searching: Search the court database for specific cases or information • Document Retrieval: Authorized users can retrieve and view electronic copies of court • Fine/Penalty: Record the payment of fines and penalties • Case Assignment: Assign a case to a specific judge or court • Electronic filing: Allows users to file or update court documents electronically.

Normalization is a process of organizing data in a database so that data is stored efficiently and redundancies are eliminated. The goal is to reduce data redundancy and improve data integrity.

Based on the entities, relationships, and transactions provided, we can create the following normalization:

First Normal Form (1NF)

- Each table has a primary key
- Columns are atomic and do not contain multiple values

Clients

- Client_Id (Primary Key)
- Fname
- LName
- Address
- Sex

Case

- Case_Id (Primary Key)
- Case_Description

Attorney

- Attorney_Id (Primary Key)
- Attorney_Name
- Address

Court

- Court_Id (Primary Key)

- City
- ZipCode
- Address
- Court_Name

Judge

- Judge_Id (Primary Key)
- Judge_Name
- Address
- Judge_PhnNo

Jury

- Jury_Id (Primary Key)
- No_of_jurors
- Juries_verdict

Second Normal Form (2NF)

- Tables are in 1NF
- All non-key attributes are dependent on the primary key

Clients

- Client_Id (Primary Key)
- Fname
- LName
- Address
- Sex

Case

- Case_Id (Primary Key)
- Case_Description

Attorney

- Attorney_Id (Primary Key)
- Attorney_Name
- Address

Court

- Court_Id (Primary Key)
- City
- ZipCode
- Address
- Court_Name

Judge

- Judge_Id (Primary Key)
- Judge_Name
- Address
- Judge_PhnNo

Jury

- Jury_Id (Primary Key)
- No_of_jurors
- Juries_verdict

Third Normal Form (3NF)

- Tables are in 2NF
- There are no transitive dependencies

Clients

- Client_Id (Primary Key)
- Fname
- LName
- Address
- Sex

Case

- Case_Id (Primary Key)
- Case_Description

Attorney

- Attorney_Id (Primary Key)
- Attorney_Name
- Address

Court

- Court_Id (Primary Key)
- City
- ZipCode
- Address
- Court_Name

Judge

- Judge_Id (Primary Key)
- Judge_Name
- Address
- Judge_PhnNo

Jury

- Jury_Id (Primary Key)
- No_of_jurors
- Juries_verdict

The above normalization provides a clear and concise way to organize the database. Each entity has its own table with a primary key, and all attributes are atomic. There are no transitive dependencies between tables. This ensures data integrity and eliminates redundancies in the database.