

CS434 – Data Base Theory and Design

Project #2

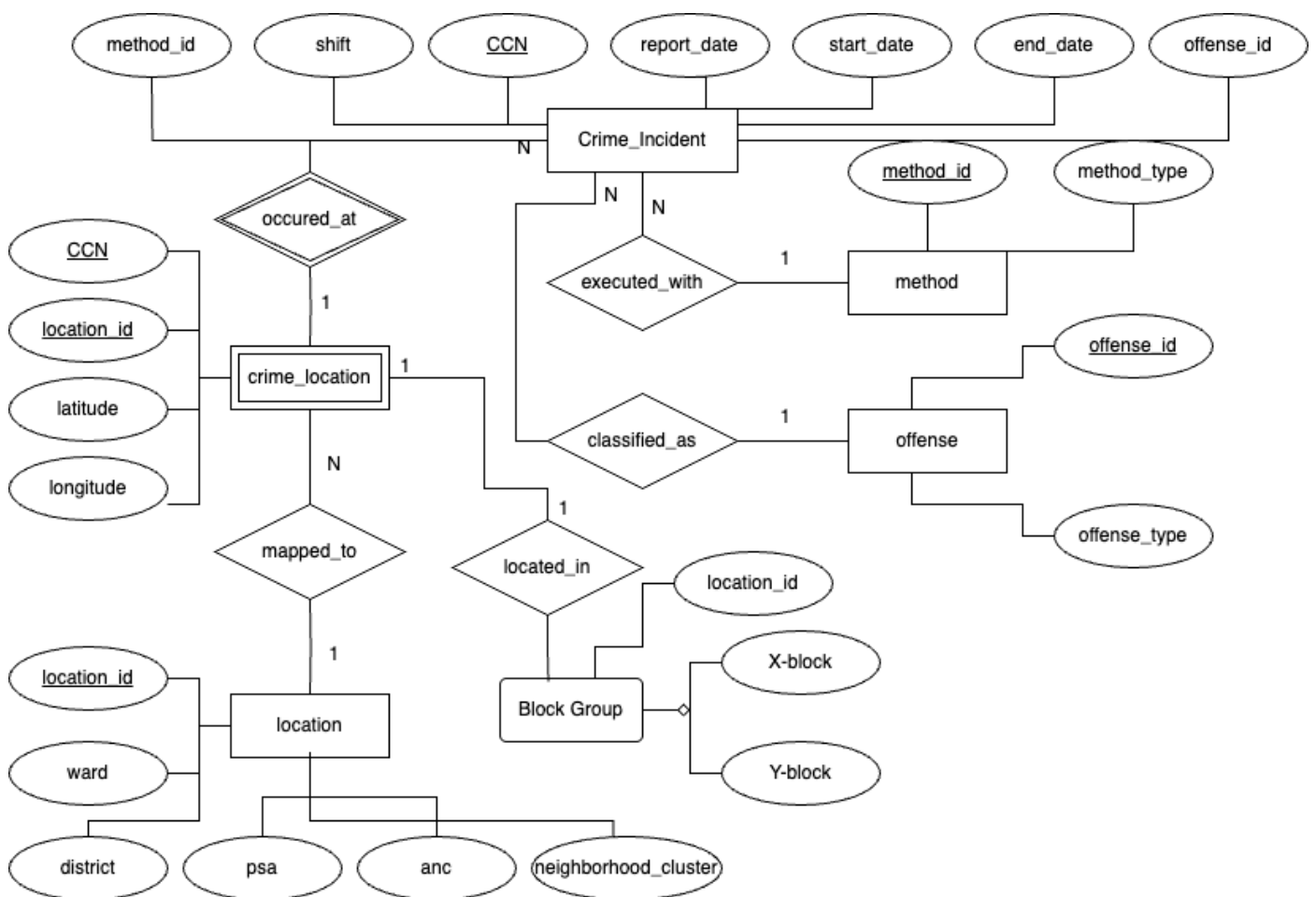
Team Database Application (TDA): Part 2 - Relational Database Design

Team

Lipika Baniya | 800794205 | lbaniya@siue.edu

The domain I would like to manage with the TDA is **Washington DC Crime Datasets 2024** by the District of Columbia Metropolitan Police Department (MPD).

Updated ER Diagram:



General Nature of application

The main goal of an Entity Relationship Diagram (ER Diagram) is to explain the relationship between entities, it is a structural design of the database. Through the help of specialized symbols it helps to define the relationship between entities. It is based on three main principles entities, attributes and relationships, these help to design the database that would be required before implementing the database.

About Data

Washington, D.C. has been facing significant challenges in ensuring public safety due to the varying and growing crime rates in different neighbourhoods and time periods. It is important for law enforcement agencies to understand when and where crimes occur so that it can respond efficiently and allocate limited resources wisely. Imagine a robust database system that is designed to handle this task effectively, because without a data-driven approach, policing efforts may remain reactive, which would result in delays or gaps in coverage in high-risk areas.

crime_incident entity, captures important information about report crime, including Criminal Complaint Number (CCN), report date, start date, end date, method_id, offense_id and what shift of the day it occurred.

These crime incident occur in certain location and each crime's location is recorded in the **crime_location** entity, which contains attributes such as CCN, location_id (forming composite key) latitude and longitude of the location.

To get more in depth and understand which area or neighbourhood the crime occurred the **location** entity provides a more descriptive layer, including attributes such as ward, district, psa, ans, and neighbourhood clusters.

There are different types of crimes that occur every day in Washington D.C., ranging in severity from minor infractions to serious violent offenses. To classify and understand this **offense** entity categorizes each crime, by assigning it a specific offense type. This enables law enforcement and analysts to systematically group and evaluate criminal activity. Similarly, **method** entity defines what tool was used for the crime, such as knife or gun or other object providing additional context and supporting detailed analysis of criminal behaviour patterns.

Lastly, the **block_group** entity ties to the specific block the crime occurred, helping to identify and address local crime patterns.

Together, these entities form a well-rounded system that allows the authorities and law enforcement get a better idea of these pattern ahead, so they can plan patrols more effectively and avoid waiting for the incident to occur. Through the help of the structured database, the state can ensure its citizen are well- protected and informed.

Entities and Attributes

Crime Incident

- **Attributes:** CCN (Primary Key), report_date, start_date, end_date, shift, offense_id, method_id,
- **Description:** The entity records primary details of each crime incident.
- **Relationships:**

- offense_id is a foreign key referencing the offense table
- method_id is a foreign key referencing the method table

Crime Location

- **Attributes:** Crime_ID, location_id (composite key), latitude, longitude
- **Description:** The entity records crime location for each crime.
- **Relationships:**
 - crime_ID is a foreign key referencing the Crime Details table.
 - location_id is a foreign key referencing the location table.

Location

- **Attributes:** location_ID (Primary Key), ward, district, psa, ans, neighborhood_cluster
- **Description:** The entity provides detailed description of the location where each crime occurred.

Offense

- **Attributes:** offense_ID (Primary Key), offense_type
- **Description:** This entity classifies the type of crimes reported.

Method

- **Attributes :** method_ID (Primary Key), method_type
- **Description:** This entity describes the tool used to commit the crime.

Block_Group

- **Attributes :** x-block, y-block, location_id
- **Description :** This entity provides information on which block the crime was reported.
- **Relationships:**
 - location_id is a foreign key referencing the location table.

Functional Dependencies

Functional Dependency is a type of constraint that helps to differentiate between a good or bad database design, it depicts the relationships between attributes.

Crime Incident

- CCN -> report_date, start_date, end_date, shift, offense_id, method_id
 - The CCN determines all the other attributes in the Crime incident entity.

Crime Location

- Location_id, CCN -> latitude, longitude

- The location_id, CCN uniquely determines all the other attributes in the Crime Location entity.

Location

- **Location_ID** -> ward, district, psa, ans, neighborhood_cluster
 - The location_id determines all the other attributes in the Location entity.

Offense

- **Offense_ID** -> offense_type
 - The offense_id uniquely determines all the other attributes in the Offense entity.

Method

- **Method_ID** -> method_type
 - The method_id uniquely determines all the other attributes in the method entity.

Attributes and its Data Types

Attributes are database entity characteristics. It defines column values like integers, characters, dates, etc.

Crime Incident

- CCN : INT (25)
- Report_date: DATETIME
- Start_date: DATETIME
- End_date: DATETIME
- Shift: VARCHAR (20)
- Method_ID: INT (20)
- Offense_ID: INT(20)

Crime Location

- Crime_ID: INT (25)
- Location_id : INT (25)
- Latitude: FLOAT
- Longitude: FLOAT

Location

- Location_ID: INT(25)
- Ward: INT (1)
- District: INT (1)
- Psa: INT (3)

- Ans: VARCHAR (5)
- Neighborhood_cluster: VARCHAR (15)

Block Group

- X-block: FLOAT
- Y-block: FLOAT
- Location_id : INT (25)

Offense

- Offense_ID: INT(25)
- Offense_type: VARCHAR(20)

Method

- Method_ID: INT(25)
- Method_type: VARCHAR(20)

Integrity Constraints

Crime Incident

- CCN : PRIMARY KEY NOT NULL
- Report_date: NOT NULL
- Start_date: NOT NULL
- End_date: NOT NULL
- Shift: NOT NULL
- Method_ID: NOT NULL
- Offense_ID: NOT NULL

Crime Location

- Crime_ID: PRIMARY KEY, FOREIGN KEY (references Crime Incident) and NOT NULL
- Location_id : FOREIGN KEY (references Location) NOT NULL
- Latitude: NOT NULL
- Longitude: NOT NULL

Location

- Location_ID: PRIMARY KEY and NOT NULL
- Ward: NOT NULL
- District: NOT NULL
- Psa: NOT NULL

- Ans: NOT NULL
- Neighborhood_cluster: NOT NULL

Block Group

- X-block: NOT NULL
- Y-block: NOT NULL
- Location_id : FOREIGN KEY (references Location) NOT NULL

Offense

- Offense_ID: PRIMARY KEY and NOT NULL
- Offense_type: NOT NULL

Method

- Method_ID: PRIMARY KEY and NOT NULL
- Method_type: NOT NULL

Question

Are there any flaws in the relational database schema you get from part 2? Are there opportunities to combine relations without introducing redundancy? If so, indicate which, and if not, tell us there are none. Are there examples of non-BCNF relation schemas? If so, do you want to decompose them? For each opportunity to combine or decompose relations (i.e., each non-BCNF relation), decide whether or not to do so, and explain your reasoning briefly (e.g., tell us what queries you expect will be typical for your database, and tell how the design you pick facilitates them). Is there anything you still don't like about the schema (e.g., attribute names, relation structure, duplicated information, etc.)? If so, modify the relational schema to something you prefer. You will be working with this schema quite a bit, so it's worth spending some time to make sure you're happy with it

Reply

The relational database schema appears well-structured. All the entities involved have primary keys, and relationships which are defined well. Yes, there are some areas where some combinations could possibly reduce complexity. For instance crime_location and crime_incident, if each crime_incident only occurred at one crime location latitude and longitude could be removed and directly embedded to crime_incident however since that does not reflect the real-world data I have kept them separate. There could be one potential non-BCNF relation schema location → ward, district, psa, anc, neighborhood_cluster. These attributes are full functionally dependent on location_id but there could be possibility for example, ward → district, or psa → district, then those transitive dependencies violate BCNF. But this would come at the cost of query complexity so denormalization benefits performance and simplicity.

Relational Schema

RELATIONAL SCHEMA

