Crime Insights : A Data-Driven Approach to Public Safety

Rajashree Shanmuganathan
rajashre
rajashre@buffalo.edu

Sindhura Velmurugan
svelmuru
svelmuru@buffalo.edu

Sadhana Saravanan
sadhanas
sadhanas@buffalo.edu

I. Problem Statement

The increasing volume and complexity of crime data from 2020 to the present pose significant challenges for law enforcement agencies, policymakers, and researchers in analyzing trends, identifying patterns, and making data-driven decisions. While crime data is available in formats like Excel, these tools are insufficient for handling large datasets, performing complex queries, and enabling real-time analysis. This project aims to create a robust, scalable database system to store, manage, and analyze crime data efficiently. The database will enable users to answer critical questions such as:

- What are the trends and patterns in crime over time?
- Which areas or neighborhoods are most affected by specific types of crime?
- How do socioeconomic factors correlate with crime rates?
- What are the seasonal or temporal variations in criminal activity?

A database is essential for this project rather than an Excel file because a database is a superior choice over an Excel file for crime data management due to its scalability, data integrity, efficiency, multi-user access, and security. While Excel struggles with large datasets and can become sluggish or prone to crashes, a database efficiently handles millions of records with structured indexing and optimized storage. Databases enforce constraints, ensuring data accuracy and reducing redundancies that are common in spreadsheets. Additionally, SQL enables powerful querying capabilities, allowing users to filter, sort, and analyze data quickly—tasks that are manual and inefficient in Excel. Unlike Excel, which lacks proper concurrency control, a database allows multiple users to access and update records simultaneously without conflicts. Furthermore, databases provide robust security features, including access controls and user permissions, ensuring that sensitive crime data remains protected from unauthorized access or accidental modifications.

A. Background of the Problem

Crime is a major societal issue that affects public safety, economic stability, and community well-being. Law enforcement agencies and policymakers depend on accurate and timely crime data to detect patterns, allocate resources, and implement effective crime prevention strategies. However, traditional crime data management methods, such as Excel spreadsheets or paper-based records, pose significant challenges. These include data redundancy, human errors, difficulty in handling large datasets, and the inability to perform complex queries efficiently. As crime rates fluctuate and urban populations grow, the need for a more efficient and scalable system to manage and analyze crime data becomes increasingly critical.

One of the major challenges faced by law enforcement is the inability to quickly retrieve and analyze crime trends. Manual data handling can result in delayed responses to emerging threats and inefficient deployment of police resources. Additionally, the lack of integration between different agencies leads to fragmented data, making it difficult to track repeat offenders or identify crime hotspots. These inefficiencies can hinder proactive policing and result in ineffective crime control strategies. Given the critical role that data plays in crime analysis and prevention, adopting a structured database system is essential for improving decision-making and operational efficiency.

B. Significance of the Problem

This project addresses the growing need for a centralized, efficient, and accessible system to analyze crime data. By transforming raw data into actionable insights, the database will contribute to:

- Improving public safety through data-driven policing strategies.
- Informing policy decisions to reduce crime rates.
- Enhancing transparency and accountability in law enforcement.
- Supporting academic research on crime and its societal impacts.

C. Potential Contribution to the Problem

This project aims to develop a database-driven crime analysis system that will significantly enhance data management, retrieval, and analysis capabilities for law enforcement agencies. By transitioning from traditional spreadsheet-based management to a relational database, the project will improve data accuracy, reduce redundancy, and provide advanced analytical capabilities. Law enforcement officers will be able to efficiently filter, sort, and analyze crime reports, enabling them to identify patterns and trends that may otherwise go unnoticed.

The contribution of this project is crucial as it will facilitate real-time crime monitoring, allowing authorities to make data-driven decisions for crime prevention. Predictive analytics can help identify high-risk areas, optimize patrol routes, and allocate resources more effectively, ultimately leading to improved public safety. Furthermore, the system will enable better coordination between law enforcement agencies by providing centralized and structured access to crime data.

Beyond law enforcement, researchers and policymakers can utilize the database to study long-term crime trends, assess the effectiveness of crime prevention strategies, and develop policies that address root causes of crime. In contrast to traditional methods, which are often time-consuming and error-prone, this database-driven approach will provide a scalable, secure, and efficient solution for crime data analysis. By leveraging technology, this project has the potential to significantly enhance crime prevention efforts and contribute to the creation of safer communities.

II. Target Users

A. Primary Users

- Law Enforcement Agencies : Police departments will use the database to track crime trends, manage cases, and allocate resources effectively.
- General Public: Citizens and community organizations can access anonymized data to understand crime in their neighborhoods and advocate for change.
- Researchers: Academics and social scientists will use the database to study crime patterns, correlations, and impacts.

B. Database Admins

- Data Scientists and Analysts: Responsible for maintaining the database, performing complex queries, and generating insights.
- Data Modellers : Ensuring the database is secure, scalable, and accessible to authorized users.

1) Real Life Scenario: A city police department is struggling with a recent spike in burglaries. Using the database, crime analysts can quickly filter data by crime type, location, and time period to identify patterns. They discover that most burglaries occur in specific neighborhoods during late evenings. Based on this insight, the department increases patrols in those areas during peak times, leading to a significant reduction in burglary rates. Meanwhile, policymakers use the database to evaluate the effectiveness of this intervention and allocate funding for additional street lighting in high-risk areas.

III. Entity-Relationship Diagram

The ER diagram below represents the structure of the crime database, showcasing the relationships between different entities.

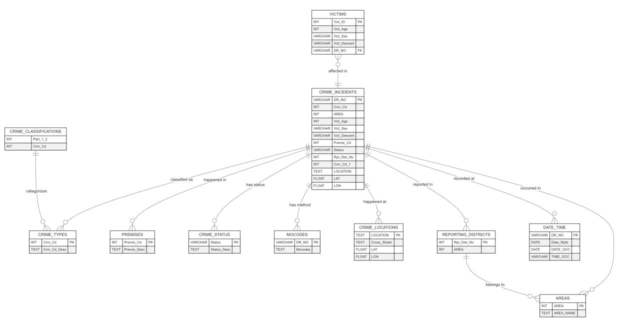


Fig. 1. Entity-Relationship Diagram for Crime Database

The following are the 11 relations present in the ER diagram:

The relationships between different tables in the ER diagram are as follows:

- **Crime_Incidents** (DR_NO, Crm_Cd, AREA, Vict_Age, Vict_Sex, Vict_Descent, Premis_Cd, Status, Rpt_Dist_No, LOCATION, LAT, LON)
  - **One-to-Many:** *Crime_Incidents* → *Victims* (DR_NO)
  - **Many-to-One:** *Crime_Incidents* → *Crime_Types* (Crm_Cd)
  - **Many-to-One:** *Crime_Incidents* → *Premises* (Premis_Cd)
  - **Many-to-One:** *Crime_Incidents* → *Reporting_Districts* (Rpt_Dist_No)
  - **Many-to-One:** *Crime_Incidents* → *Areas* (AREA)
- **Victims** (Vict_ID, Vict_Age, Vict_Sex, Vict_Descent, DR_NO *(FK)*)
  - **Many-to-One:** *Victims* → *Crime_Incidents* (DR_NO)
- **Crime_Types** (Crm_Cd, Crm_Cd_Desc)
  - **One-to-Many:** *Crime_Types* → *Crime_Incidents* (Crm_Cd)
- **Premises** (Premis_Cd, Premis_Desc)
  - **One-to-Many:** *Premises* → *Crime_Incidents* (Premis_Cd)
- **Crime_Status** (Status, Status_Desc)
  - **One-to-Many:** *Crime_Status* → *Crime_Incidents* (Status)
- **MOCodes** (DR_NO, MOCodes)
  - **One-to-Many:** *Crime_Incidents* → *MOCodes* (DR_NO)
- **Crime_Locations** (LOCATION, Cross_Street, LAT, LON)
  - **One-to-Many:** *Crime_Locations* → *Crime_Incidents* (LOCATION)
- **Reporting_Districts** (Rpt_Dist_No, AREA *(FK)*)
  - **One-to-Many:** *Reporting_Districts* → *Crime_Incidents* (Rpt_Dist_No)
- **Date_Time** (DR_NO, Date_Rptd, DATE_OCC, TIME_OCC, AREA *(FK)*)
  - **One-to-One:** *Crime_Incidents* → *Date_Time* (DR_NO)
- **Areas** (AREA, AREA_NAME)
  - **One-to-Many:** *Areas* → *Crime_Incidents* (AREA)
- **Crime_Classifications** (Crm_Cd, Part_1_2)
  - **One-to-Many:** *Crime_Classifications* → *Crime_Types* (Crm_Cd)

IV. Functional Dependencies (FDs) for Each Relation

A. Crime_Incidents (Primary Key: DR_NO)

FDs:

DR_NO → Crm_Cd, AREA, Vict_Age, Vict_Sex, Vict_Descent, Premis_Cd, Status, Rpt_Dist_No, LOCATION, LAT, LON

BCNF Check:
Since DR_NO is a superkey, this relation is in BCNF.

B. Crime_Types (Primary Key: Crm_Cd)

FDs:

$$\text{Crm\_Cd} \rightarrow \text{Crm\_Cd\_Desc}$$

BCNF Check:
Since Crm_Cd is a superkey, this relation is in BCNF.

C. Areas (Primary Key: AREA)

FDs:

$$\text{AREA} \rightarrow \text{AREA\_NAME}$$

BCNF Check:
Since AREA is a superkey, this relation is in BCNF.

D. Victims (Primary Key: Vict_ID)

FDs:

$$\text{Vict\_ID} \rightarrow \text{Vict\_Age, Vict\_Sex, Vict\_Descent, DR\_NO}$$

BCNF Check:
Since Vict_ID is a superkey, this relation is in BCNF.

E. Premises (Primary Key: Premis_Cd)

FDs:

$$\text{Premis\_Cd} \rightarrow \text{Premis\_Desc}$$

BCNF Check:
Since Premis_Cd is a superkey, this relation is in BCNF.

F. Crime_Status (Primary Key: Status)

FDs:

$$\text{Status} \rightarrow \text{Status\_Desc}$$

BCNF Check:
Since Status is a superkey, this relation is in BCNF.

G. Mocodes (Primary Key: DR_NO)

FDs:

$$\text{DR\_NO} \rightarrow \text{Mocodes}$$

BCNF Check:
Since DR_NO is a superkey, this relation is in BCNF.

H. Crime_Locations (Primary Key: LOCATION)

FDs:

$$\text{LOCATION} \rightarrow \text{Cross\_Street, LAT, LON}$$

BCNF Check:
Since LOCATION is a superkey, this relation is in BCNF.

I. Reporting_Districts (Primary Key: Rpt_Dist_No)

FDs:

$$\text{Rpt\_Dist\_No} \rightarrow \text{AREA}$$

BCNF Check:
Since Rpt_Dist_No is a superkey, this relation is in BCNF.

J. Crime_Classifications (Primary Key: Crm_Cd)

FDs:

$$\text{Crm\_Cd} \rightarrow \text{Part\_1\_2}$$

BCNF Check:
Since Crm_Cd is a superkey, this relation is in BCNF.

K. Date_Time (Primary Key: DR_NO)

FDs:

$$\text{DR\_NO} \rightarrow \text{Date\_Rptd, Date\_Occ, Time\_Occ}$$

BCNF Check:
Since DR_NO is a superkey, this relation is in BCNF.

V. Primary Keys and Foreign Keys for Each Relation

A. Crime_Incidents

Primary Key: DR_NO
Foreign Keys:

- Crm_Cd → Crime_Types (Crime classification)
- AREA → Areas (Crime location)
- Premis_Cd → Premises (Location type)
- Rpt_Dist_No → Reporting_Districts (District association)

Justification: Each crime incident has a unique identifier (DR_NO). The foreign keys ensure referential integrity by linking crime incidents to crime types, areas, premises, and reporting districts.

B. Crime_Types

Primary Key: Crm_Cd
Justification: Each crime type is uniquely identified by its code.

C. Areas

Primary Key: AREA
Justification: Each geographical area has a unique identifier.

D. Victims

Primary Key: Vict_ID
Foreign Keys:

- DR_NO → Crime_Incidents (Linking victims to crime incidents)

Justification: Each victim has a unique ID, and the foreign key ensures that each victim is linked to a crime.

E. Premises

Primary Key: Premis_Cd
Justification: Each type of premise where crimes occur has a unique identifier.

F. Crime_Status

Primary Key: Status
Justification: Each crime status (e.g., pending, closed) is uniquely identified.

G. Mocodes

Primary Key: DR_NO
Justification: Each crime (DR_NO) can have multiple modus operandi codes.

H. Crime_Locations

Primary Key: LOCATION
Justification: Each crime location is uniquely identified.

I. Reporting_Districts

Primary Key: Rpt_Dist_No
Foreign Keys:

- AREA → Areas (Each district belongs to an area)

Justification: Each reporting district has a unique number, and it belongs to an area.

J. Crime_Classifications

Primary Key: Crm_Cd
Justification: Each crime classification has a unique identifier.

K. Date_Time

Primary Key: DR_NO
Justification: Each crime incident (DR_NO) has an associated date and time.

VI. List of Relations and Attributes

A. Crime_Incidents

Attribute: DR_NO
Datatype: VARCHAR(20)
Description: Unique identifier for each crime incident
Null Allowed: No
Default Value: -

Attribute: Crm_Cd
Datatype: INT
Description: Crime classification code
Null Allowed: No
Default Value: -

Attribute: AREA
Datatype: INT
Description: Area where the crime happened
Null Allowed: No
Default Value: -

Attribute: Vict_Age
Datatype: INT
Description: Age of the victim (if available)
Null Allowed: Yes
Default Value: NULL

Attribute: Vict_Sex
Datatype: VARCHAR(10)
Description: Gender of the victim (M, F, X)
Null Allowed: Yes
Default Value: NULL

Attribute: Vict_Descent
Datatype: VARCHAR(10)
Description: Ethnicity of the victim
Null Allowed: Yes
Default Value: NULL

Attribute: Premis_Cd
Datatype: INT
Description: Code for the premise where crime occurred
Null Allowed: No
Default Value: -

Attribute: Status
Datatype: VARCHAR(20)
Description: Crime status (e.g., pending, closed)
Null Allowed: No
Default Value: -

Attribute: Rpt_Dist_No
Datatype: INT
Description: Reporting district number
Null Allowed: No
Default Value: -

Attribute: LOCATION
Datatype: VARCHAR(255)
Description: Description of the crime location
Null Allowed: Yes
Default Value: NULL

Attribute: LAT
Datatype: FLOAT
Description: Latitude coordinates
Null Allowed: Yes
Default Value: NULL

Attribute: LON
Datatype: FLOAT
Description: Longitude coordinates
Null Allowed: Yes
Default Value: NULL

Foreign Key Constraints:
Crm_Cd → Crime_Types(Crm_Cd) (ON DELETE CASCADE)
AREA → Areas(AREA) (ON DELETE SET NULL)
Premis_Cd → Premises(Premis_Cd) (ON DELETE SET NULL)
Rpt_Dist_No → Reporting_Districts(Rpt_Dist_No) (ON DELETE SET NULL)

B. Crime_Types

Attribute: Crm_Cd
Datatype: INT (PK)
Description: Unique identifier for crime type
Null Allowed: No
Default Value: -

Attribute: Crm_Cd_Desc
Datatype: TEXT
Description: Description of the crime type
Null Allowed: No
Default Value: -

Foreign Key Constraints: None

C. Victims

Attribute: Vict_ID
Datatype: INT (PK)
Description: Unique ID for each victim
Null Allowed: No

Default Value: -

Attribute: Vict_Age
Datatype: INT
Description: Age of the victim
Null Allowed: Yes
Default Value: NULL

Attribute: Vict_Sex
Datatype: VARCHAR(10)
Description: Gender of the victim (M, F, X)
Null Allowed: Yes
Default Value: NULL

Attribute: Vict_Descent
Datatype: VARCHAR(10)
Description: Ethnicity of the victim
Null Allowed: Yes
Default Value: NULL

Attribute: DR_NO
Datatype: VARCHAR(20)
Description: Crime incident ID related to the victim
Null Allowed: No
Default Value: -

Foreign Key Constraints:
DR_NO → Crime_Incidents(DR_NO) (ON DELETE CASCADE)

D. Premises

Attribute: Premis_Cd
Datatype: INT (PK)
Description: Unique ID for premise type
Null Allowed: No
Default Value: -

Attribute: Premis_Desc
Datatype: TEXT
Description: Description of premise type
Null Allowed: No
Default Value: -

Foreign Key Constraints: None

E. Reporting_Districts

Attribute: Rpt_Dist_No
Datatype: INT (PK)
Description: Reporting district number
Null Allowed: No
Default Value: -

Attribute: AREA
Datatype: INT
Description: Area ID related to this district
Null Allowed: No
Default Value: -

Foreign Key Constraints:
AREA → Areas(AREA) (ON DELETE CASCADE)

F. Crime_Status

Attribute: Status
Datatype: VARCHAR(20) (PK)
Description: Status of the crime (e.g., open, closed)
Null Allowed: No
Default Value: -

Attribute: Status_Desc
Datatype: TEXT
Description: Description of crime status
Null Allowed: No
Default Value: -

Foreign Key Constraints: None

G. Crime_Locations

Attribute: LOCATION
Datatype: TEXT (PK)
Description: Unique identifier for location
Null Allowed: No
Default Value: -

Attribute: Cross_Street
Datatype: TEXT
Description: Cross street information
Null Allowed: Yes
Default Value: NULL

Attribute: LAT
Datatype: FLOAT
Description: Latitude coordinates
Null Allowed: Yes
Default Value: NULL

Attribute: LON
Datatype: FLOAT
Description: Longitude coordinates
Null Allowed: Yes
Default Value: NULL

Foreign Key Constraints: None

VII. Foreign Key Action Summary

- **ON DELETE CASCADE**:
  - **Crime_Incidents → Victims**: If a crime incident is deleted, the corresponding victim record is also removed. Each crime record is associated with a victim, and if the crime is removed, victim details should not persist independently.
  - **Crime_Incidents → Crime_Types**: If a crime type is deleted, all crime incidents of that type are also deleted. A crime must always have a valid type. If a type is removed, related crimes should also be removed to maintain consistency.
  - **Crime_Incidents → Reporting_Districts**: If a reporting district is deleted, all crimes reported

in that district are deleted. Crimes are linked to a reporting district, and if the district no longer exists, it makes sense to remove associated incidents rather than leaving them with an invalid reference.

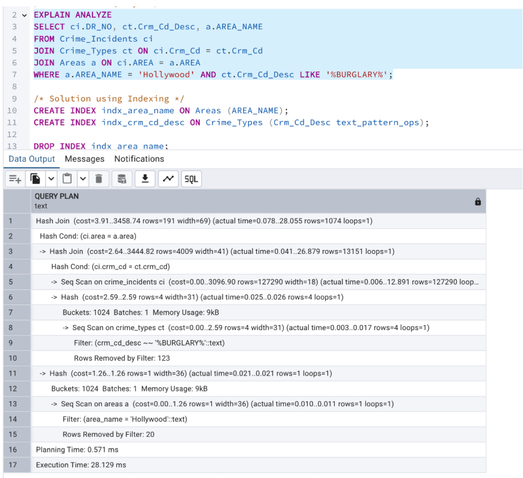
- **ON DELETE SET NULL**:
  - **Crime_Incidents** → **Areas**: If an area is deleted, the corresponding area reference in the crime incidents table is set to NULL. This ensures that crime incidents are not deleted but are instead disassociated from the deleted area.
  - **Crime_Incidents** → **Premises**: If a premise is deleted, the corresponding premise reference in the crime incidents table is set to NULL. This ensures that crime incidents are not deleted but are instead disassociated from the deleted premise.

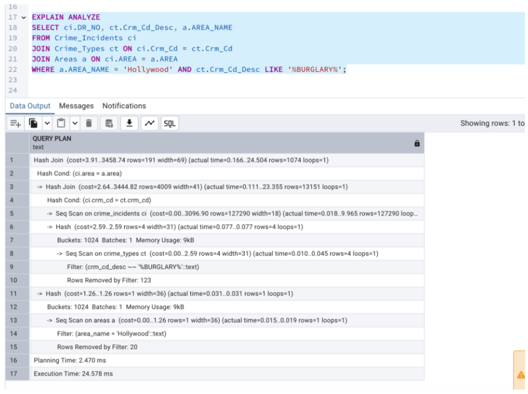
VIII. Indexing and Query Execution Analysis

A. Problematic Query 1: Search Crimes by Area Name and Crime Description

1) Query Description: We want to find crimes reported in a particular area (e.g., `Hollywood`) with specific keywords (e.g., `BURGLARY`) in the crime description.

2) Initial Query Performance (Before Optimization): Initially, no proper indexes existed on the `AREA_NAME` or `Crm_Cd_Desc`, resulting in sequential scans and slower performance.





Execution Time: Approximately 28 ms.

3) Optimization Strategy: To improve query performance, the following indexes were created:

- `CREATE INDEX idx_area_name ON Areas(AREA_NAME);`
- `CREATE INDEX idx_crm_cd_desc ON Crime_Types(Crm_Cd_Desc text_pattern_ops);`

4) Query Performance After Indexing: After applying the indexes, index scans were performed instead of sequential scans, significantly improving performance.

Execution Time: Approximately 24 ms.

B. Problematic Query 2: Search by Latitude and Longitude Range

1) Query Description: We want to find crimes that occurred within a specific latitude and longitude boundary.

2) Initial Query Performance (Before Optimization): Initially, the `Crime_Incidents` table had no spatial indexes, leading to full table scans and slower query execution.

Execution Time: Approximately 25 ms.

3) Optimization Strategy: To improve query performance, a composite index on latitude and longitude was created:

- `CREATE INDEX idx_lat_lon ON Crime_Incidents(LAT, LON);`

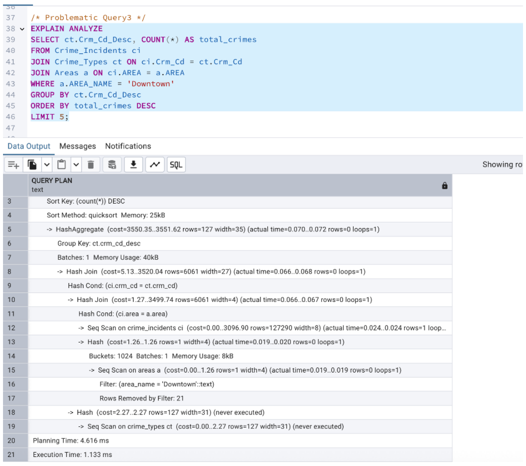
4) Query Performance After Indexing: After creating the composite index, the database used index range scans, which reduced the query time substantially.

Execution Time: Approximately 9 ms.

C. Problematic Query 3: Find Top 5 Crime Types in a Specific Area

1) Query Description: We want to find the **Top 5 most frequently reported crime types** in a specific area, for example, `Downtown`.

2) Initial Query Performance (Before Optimization): Initially, no indexes existed on `AREA` or `Crm_Cd` columns, leading to sequential scans and slower performance, especially when grouping and ordering by crime type.



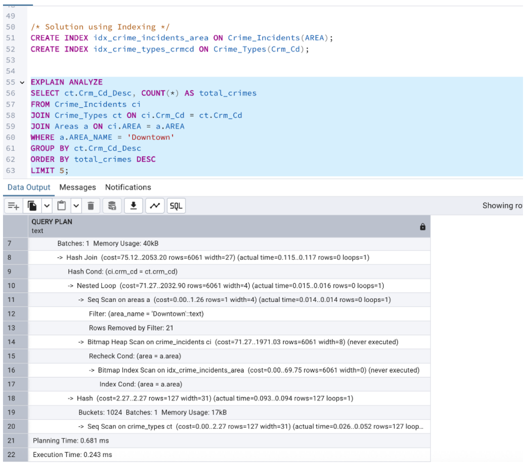
Execution Time: Approximately 1.2 ms.

3) Optimization Strategy: To optimize the query, the following indexes were created:

- `CREATE INDEX idx_crime_incidents_area ON Crime_Incidents(AREA);`
- `CREATE INDEX idx_crime_types_crmcd ON Crime_Types(Crm_Cd);`

These indexes improved the join operations between `Crime_Incidents`, `Crime_Types`, and `Areas`, enabling faster filtering, grouping, and sorting.

4) Query Performance After Indexing: After creating the indexes, the query execution utilized index scans instead of full sequential scans, which significantly improved the performance.



Execution Time: Approximately 0.2 ms.

IX. References

The dataset used for this project is sourced from the official U.S. government open data portal. It contains crime incident reports from 2020 to the present, including various attributes such as crime type, location, victim details, and more.

The dataset can be accessed at the following link: **Crime Data from 2020 to Present**