# CITS5504 Data Warehousing Project 1

**Jiaze Li (23266049)**

## Is Atlanta Safe? Analysis of Crime Data in the city of Atlanta

**Q1:** Which type of building have the highest crime count?
**Building Type**

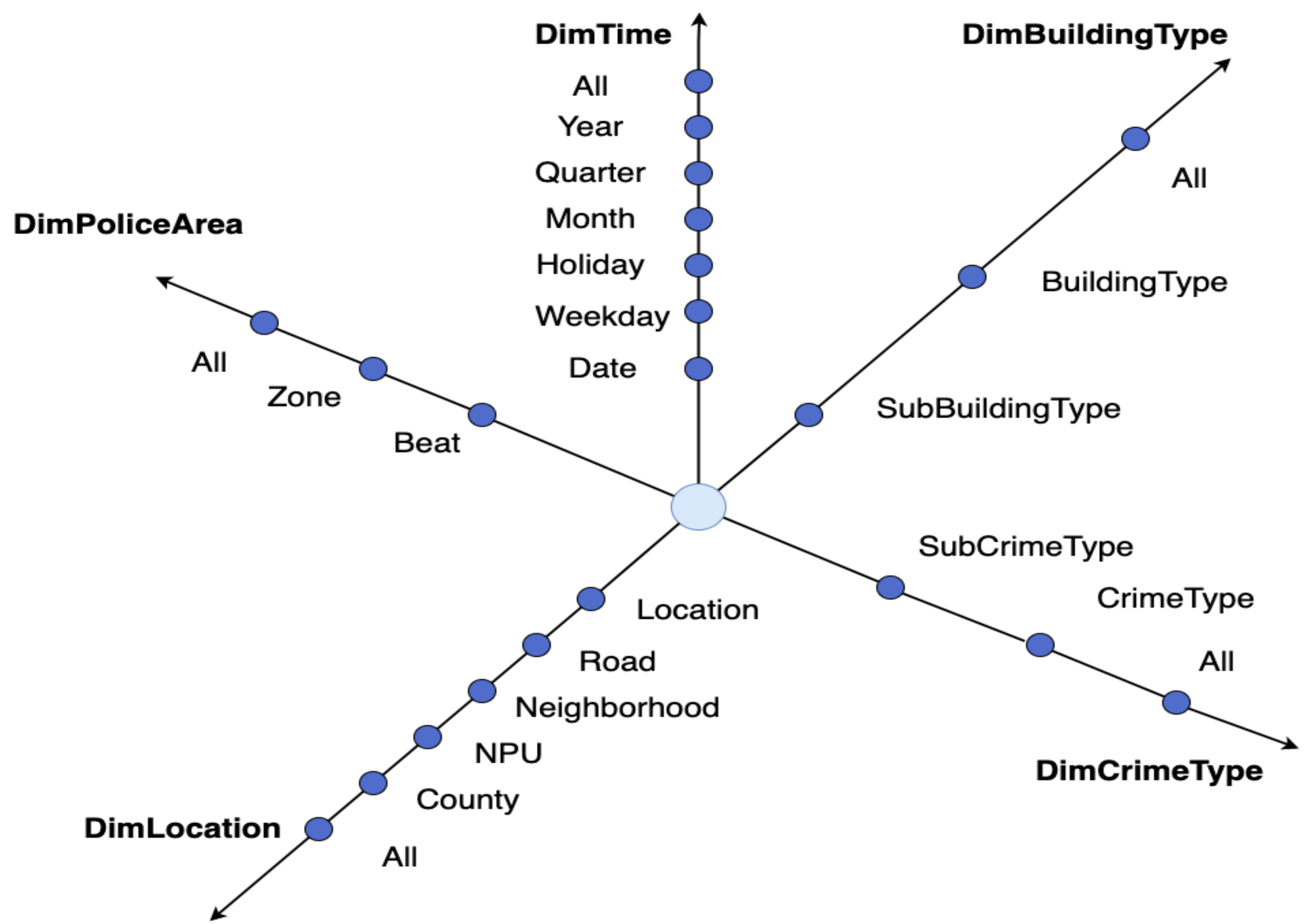
**Q2:** What the trend of residential area crime count per year?
**Extension:** Which month had the highest count of crime in residential areas?
**Building Type - Time**

**Q3:** What is the distribute of crime types in residential areas during July?
**Building Type - Time - Crime Type**

**Q4:** Which NPU had the highest count of property crime in residential areas during July?
**Building Type - Time - Crime Type - Locaion**

**Q5:** Which zone had the lowest count of property crime in residential areas during July?
**Building Type - Time - Crime Type - Police Area**

# Star Net

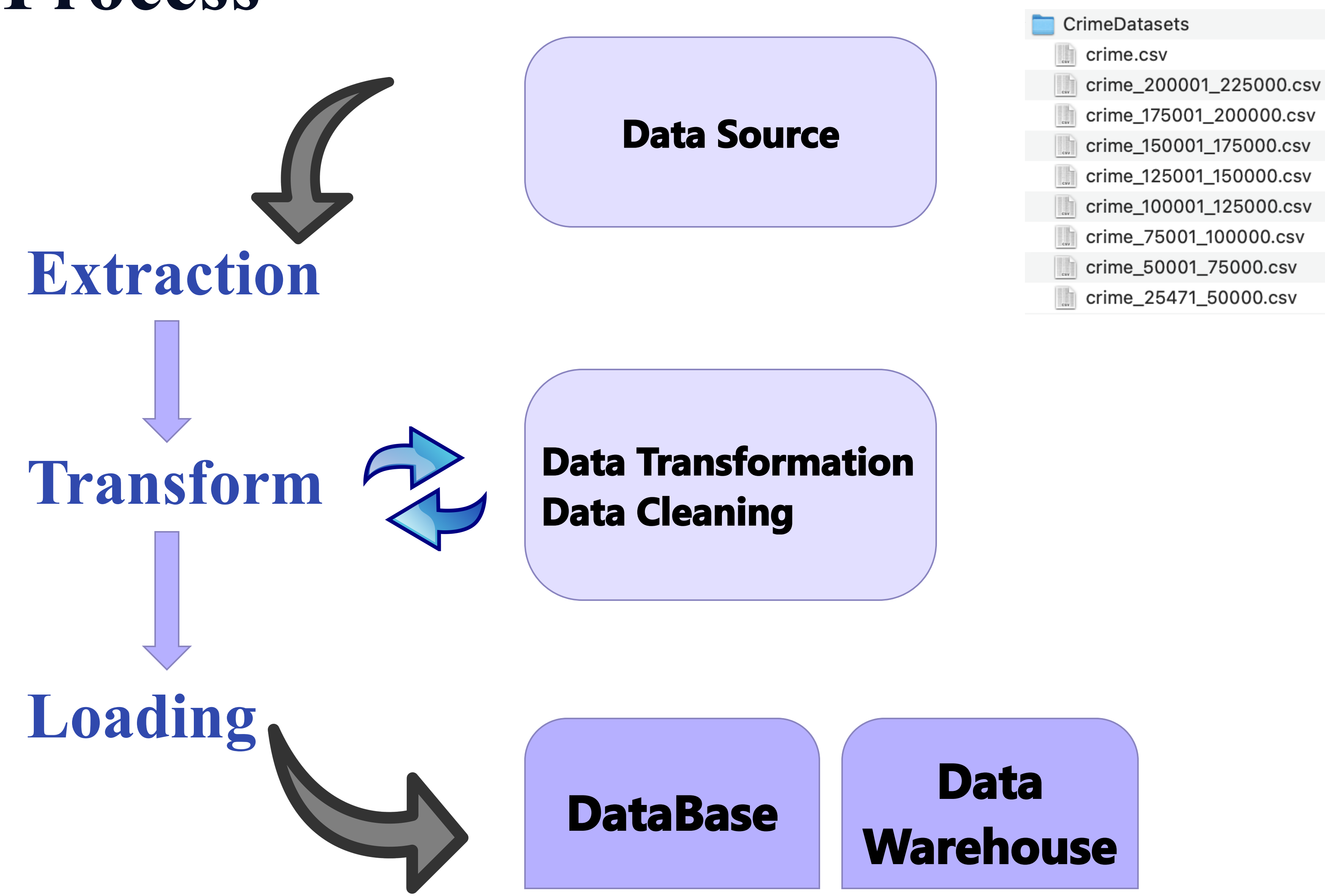



*StarNet Footprint*

For this project, the star schema was chosen due to its easily understandable queries, efficient query performance, and fewer foreign keys compared to snowflake schema.

Based on StarNet and business query analysis, the project includes **one fact** table and **five dimension** tables (**Time**, **Location**, **Crime Type**, **Police Area**, and **Building Type**), connected by five foreign keys.

# Data Process



# **Extract** data from **9 source files** by reading CSV files with **Python** and merging them into a single dataframe that contains data from all the source files.

CrimeDatasets
- crime.csv
- crime_200001_225000.csv
- crime_175001_200000.csv
- crime_150001_175000.csv
- crime_125001_150000.csv
- crime_100001_125000.csv
- crime_75001_100000.csv
- crime_50001_75000.csv
- crime_25471_50000.csv

```
crime1 = pd.read_csv("CrimeDatasets/crime.csv")
```

```
crime2 = pd.read_csv("CrimeDatasets/crime_25471_50000.csv")
crime3 = pd.read_csv("CrimeDatasets/crime_50001_75000.csv")
crime4 = pd.read_csv("CrimeDatasets/crime_75001_100000.csv")
crime5 = pd.read_csv("CrimeDatasets/crime_100001_125000.csv")
crime6 = pd.read_csv("CrimeDatasets/crime_125001_150000.csv")
crime7 = pd.read_csv("CrimeDatasets/crime_150001_175000.csv")
crime8 = pd.read_csv("CrimeDatasets/crime_175001_200000.csv")
crime9 = pd.read_csv("CrimeDatasets/crime_200001_225000.csv")
```

```
crime_full = pd.concat([crime_clean, crime2, crime3, crime4, crime5, crime6, crime7, crime8, crime9])
```

```
crime_full.head()
```

✓ 0.1s Python

| ed: 0 | crime | number | date | location | beat | neighborhood | npu | lat | long | type | road | neighbourhoo |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | LARCENY-NON VEHICLE | 103040029 | 10/31/2010 | 610 SPRING ST NW | 509 | Downtown | M | 33.77101 | -84.38895 | house_number | Spring Street Northwest | |
| 1 | AUTO THEFT | 103040061 | 10/31/2010 | 850 OAK ST SW | 401 | West End | T | 33.74057 | -84.41680 | office | Oak Street Southwest | |
| 2 | LARCENY-FROM VEHICLE | 103040169 | 10/31/2010 | 1344 METROPOLITAN PKWY SW | 301 | Capitol View Manor | X | 33.71803 | -84.40774 | shop | Metropolitan Parkway Southwest | Ca |
| 3 | AUTO THEFT | 103040174 | 10/31/2010 | 1752 PRYOR RD SW | 307 | Betmar LaVilla | Y | 33.70731 | -84.39674 | house_number | Pryor Street | |
| 4 | LARCENY-NON VEHICLE | 103040301 | 10/31/2010 | JOHN WESLEY DOBBS AVE NE / CORLEY ST NE | 604 | Old Fourth Ward | M | 33.75947 | -84.36626 | house_number | John Wesley Dobbs Avenue Northeast | In |

After the Extract phase is complete, we will have a single DataFrame containing all the crime data. We can then move on to the **Transform phase** to clean and process the data.

# Data Cleaning

## Single Value Columes

As **country** and **state** all have the **same single value**, the location attribute can be reduced to more specific information such as county, road, neighbourhood, npu etc. which reduces the complexity of storing and processing data and also improves accuracy and usability. This also reduces the complexity of data storage and processing, avoids data redundancy, and also improves the accuracy and availability of the data.

```
Unique values in column "country":
['United States']

Unique values in column "state":
['Georgia']
```

## Drop Highly Missing Rate Columns

Calculations show a missing rate of **neighbourhood_lookup** almost reach **40%**, well above the common standard for high deletion rates (greater than 5% or 10%), so deletion was considered.

## Drop Not Useful Columns

In Crime data files, not all columns are useful for this project; therefore, some columns need to be dropped. **Unnamed:0**, **postcode**, **number**, **Lat** and **Long** column need to be dropped since they are not used in this project.

```
crime_full = crime_full.drop(columns=['Unnamed: 0'])
crime_full = crime_full.drop(columns=['postcode'])
crime_full = crime_full.drop(columns=['number'])
crime_full = crime_full.drop(columns=['lat'])
crime_full = crime_full.drop(columns=['long'])
```

Eliminate '**City**' column; in this project we only focus on **Atlanta city**, disregard 'Sandy Springs' for less redundancy.

```
Unique values in column "city":
['Atlanta' nan 'Sandy Springs']
```

# Data Cleaning

## Missing Value Issue

Fill missing values with '**unknown**' when absence of data has meaning or context, and imputation may cause bias. *(Little & Rubin, 2002)*

```
crime_full['location'] = crime_full['location'].fillna('Unknown')
crime_full['neighborhood'] = crime_full['neighborhood'].fillna('Unknown')
crime_full['npu'] = crime_full['npu'].fillna('Unknown')
crime_full['road'] = crime_full['road'].fillna('Unknown')
```

## Location Values Issue

Although the symbols @, /, (), etc. appear in the location, they all have their own specific meaning, e.g. **after @** it means the specific type of building in the location, **after /** it may mean that the location is uncertain or comes between the two locations, and **in ()** it means the specific location. So we should keep them and **only remove** the '**;**', which affects the analysis, and remove the **extra spaces**

## Next Step

After all cleaning processes, the selecting data need to be filled into **dimension tables and fact table**.

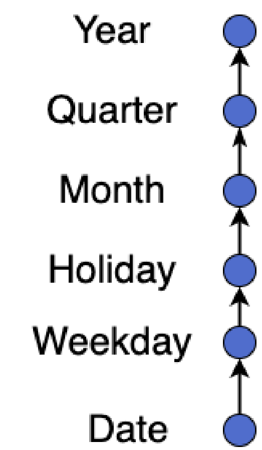
# Dimension Design

**Time**

## Schema Hierarchy

**Total order**:

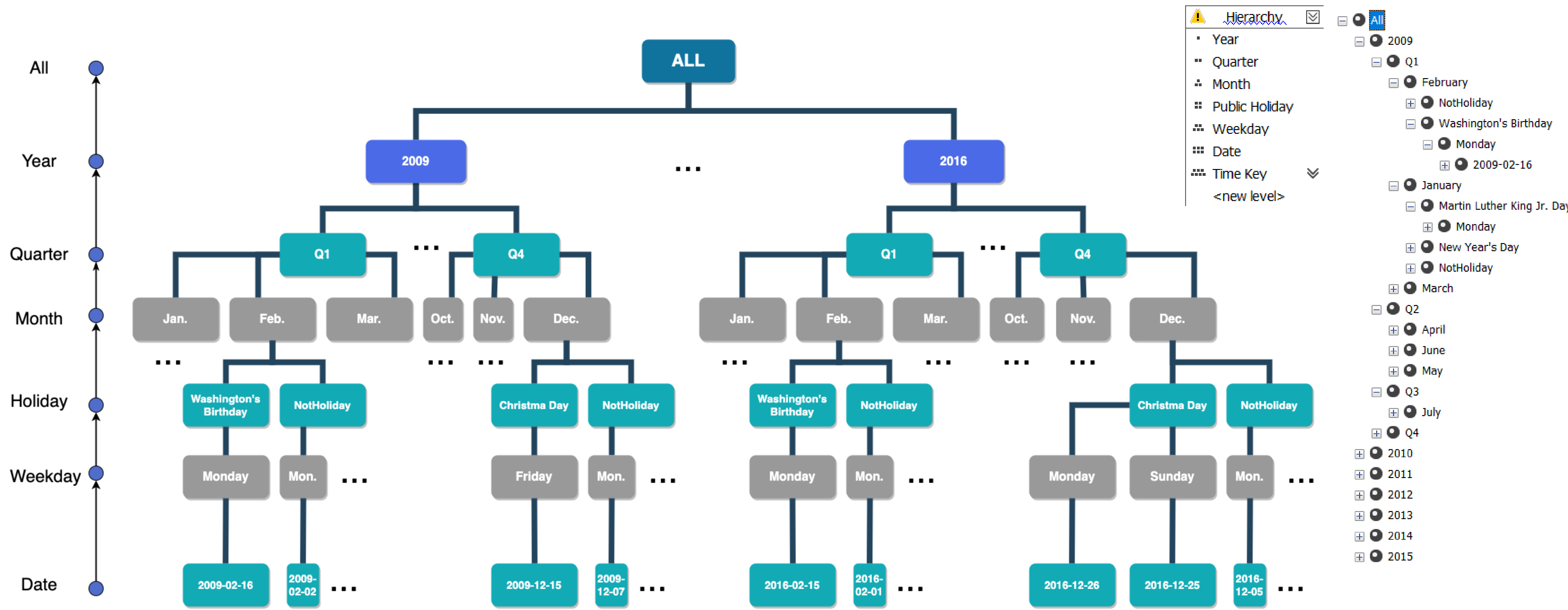
**Date < Weekday < Holiday < Month < Quarter < Year**




"**Date**" represents a specific day and serves as the foundation.
"**Weekday**" groups data based on days of the week to analyze patterns.
"**Holiday**" identifies significant dates, allowing for holiday impact analysis.
"**Month**" and "**Quarter**" enable concise analysis of trends and patterns within different time segments.
"**Year**" provides the highest level of aggregation for long-term trends and comparisons.

The Time Dimension is designed to facilitate in-depth analysis and insights from Crime dataset. It provides a structured hierarchy and these sub-dimensions **roll up** to the Year dimension, that allows for easy data aggregation and filtering across various time-related attributes.

# Time Hierarchy




*Schema Hierarchy* *Concept Hierarchy* *Hierarchy in Visual Studio*

# Dimension Design

## Schema Hierarchy

**Total order**:

**Location**

**Location <  Road < Neighborhood < NPU < County**

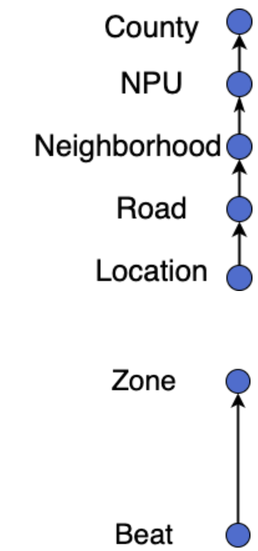
**NPU** & **Neighborhood** relationship backed by City of Atlanta; both included in Location dimension. *(City of Atlanta, n.d.)*

**Zone**

**Beat <  Zone**

* Atlanta Police Department (APD) divides the city into **Zones** and **Beats**.
* Zones consist of several Beats, with each Beat representing a specific geographical area.

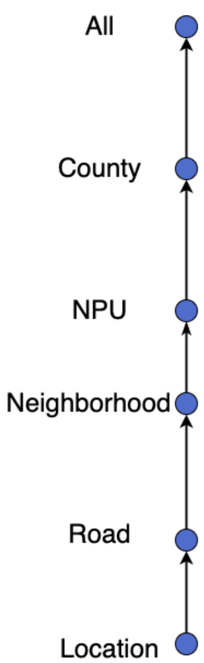
*(City of Atlanta Police Department, n.d.)*



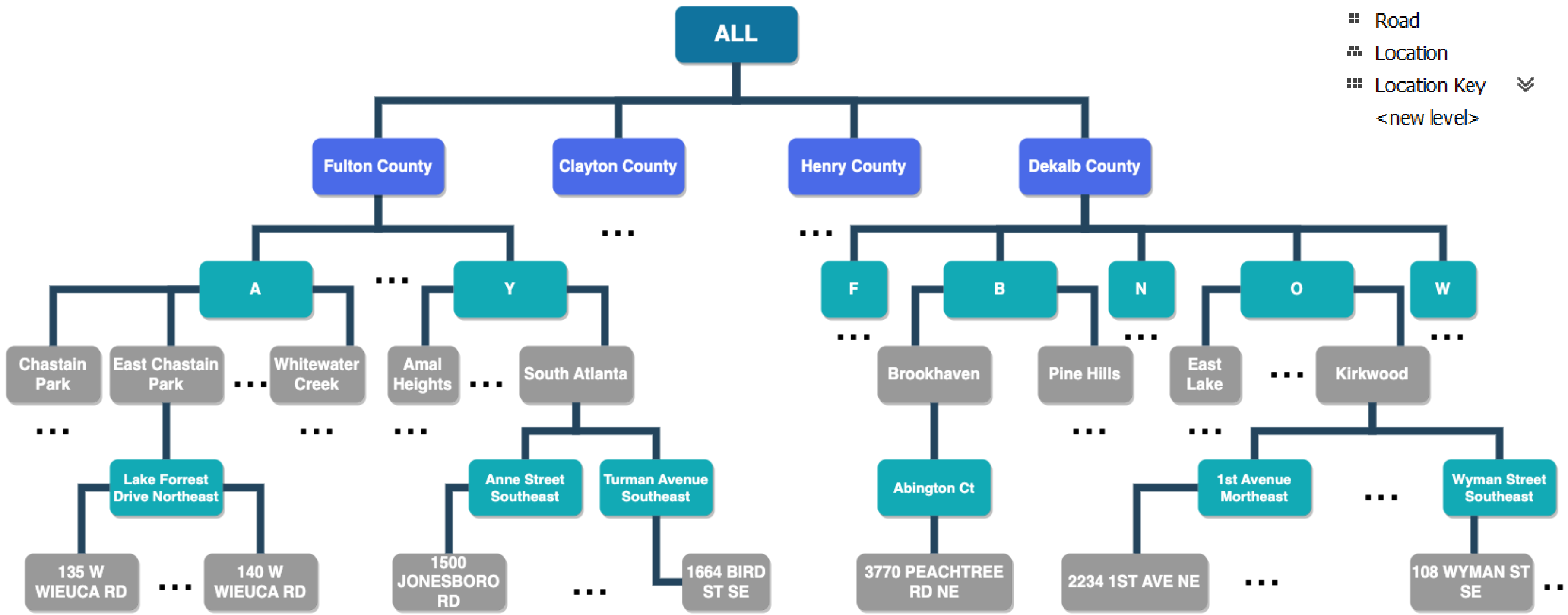

To solve business queries related to analyzing **security situations in different areas**, LocationDim and PoliceAreaDim are designed as **two separate dimensions**, even though both contain geographical information.

This design enables efficient data analysis by allowing users to **drill down** from higher levels to lower levels of geographic granularity or roll up from lower to higher for aggregation, helping in the identification of **patterns and trends** in specific regions.

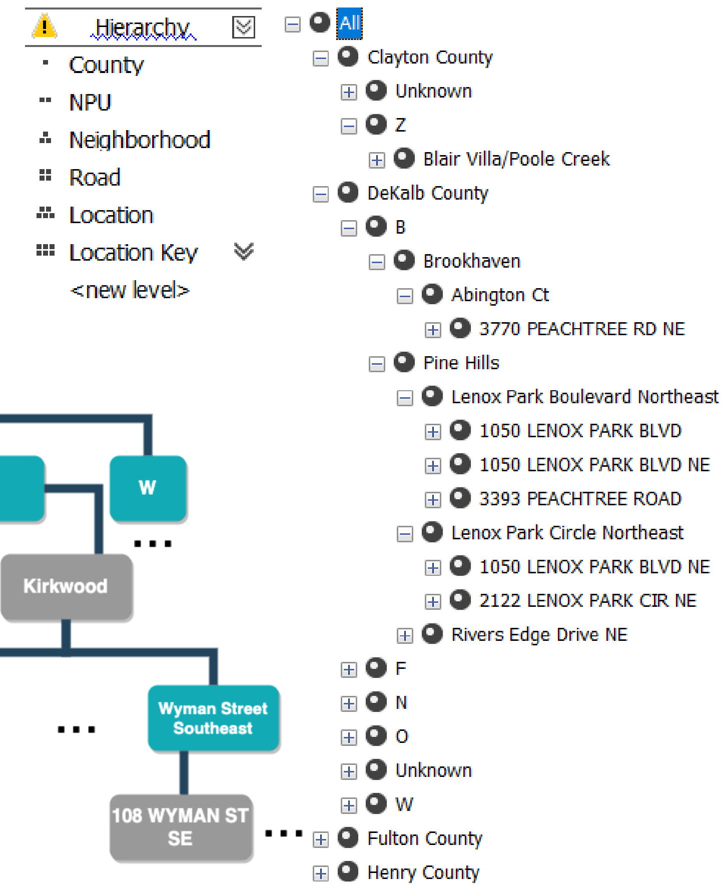
# Location Hierarchy




*Schema Hierarchy*

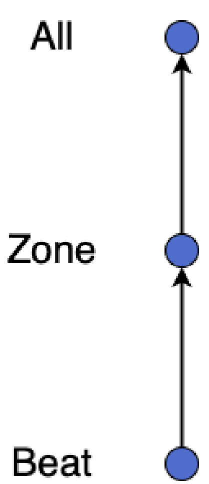



*Concept Hierarchy*

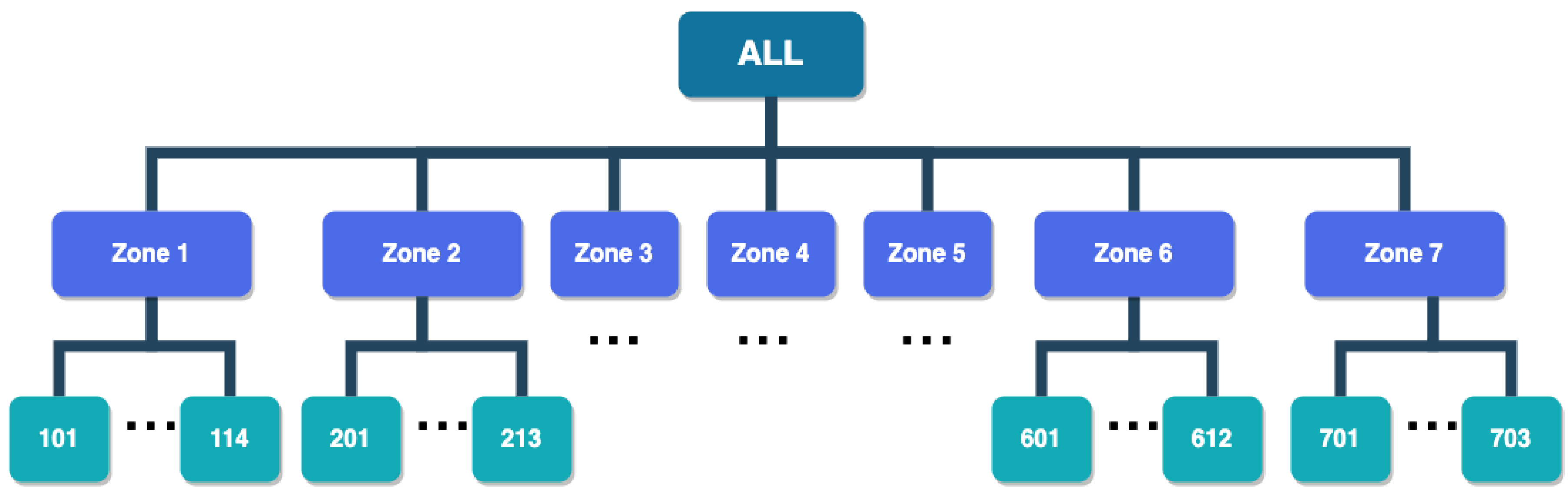



*Hierarchy in Visual Studio*

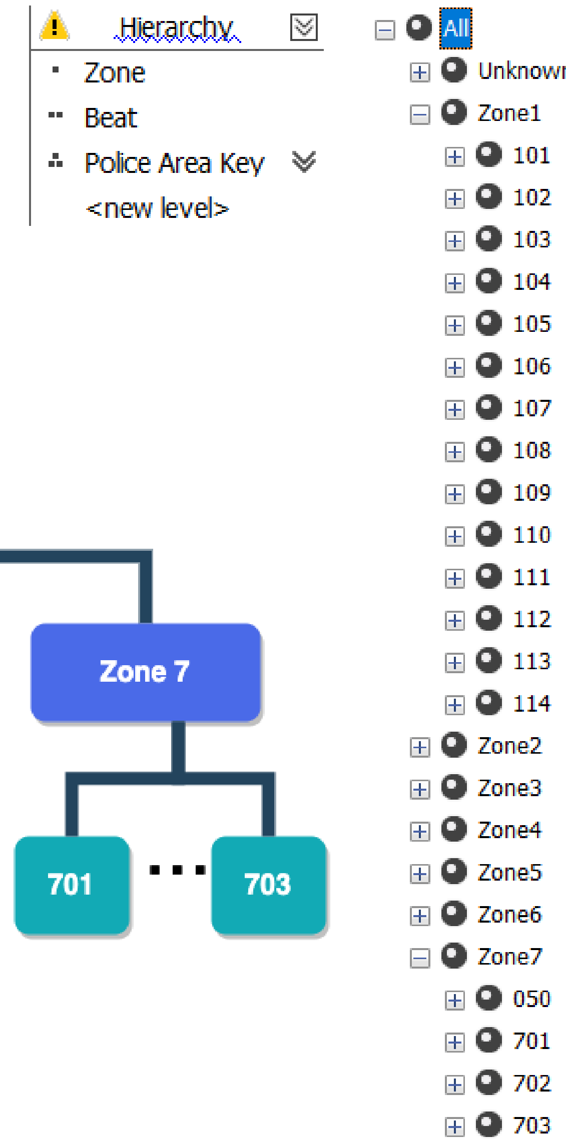
# Police Area Hierarchy




*Schema Hierarchy*

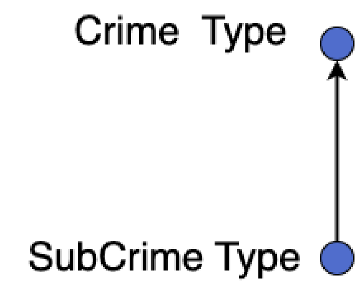



*Concept Hierarchy*




*Hierarchy in Visual Studio*

# Dimension Design




**Crime Type**

## Schema Hierarchy

**Total order**:

**SubCrimeType < CrimeType**

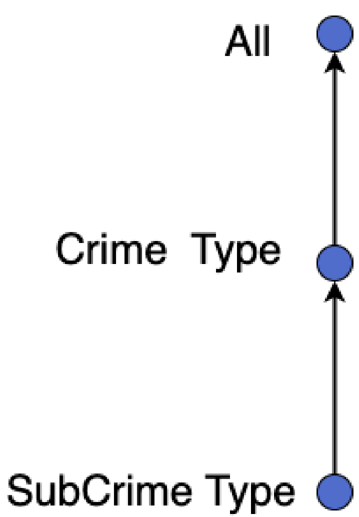
```
def get_crime_type(crime):
    if crime in ['LARCENY-NON VEHICLE', 'LARCENY-FROM VEHICLE', 'BURGLARY-RESIDENCE', 'BURGLARY-NONRES', 'AUTO THEFT']:
        return 'Property Crime'
    elif crime in ['ROBBERY-PEDESTRIAN', 'ROBBERY-RESIDENCE', 'ROBBERY-COMMERCIAL']:
        return 'Robbery'
    elif crime in ['AGG ASSAULT', 'RAPE']:
        return 'Violent Crime'
    elif crime in ['HOMICIDE']:
        return 'Homicide'
    else:
        return 'Other'
```

This hierarchical classification allows us to analyze the public safety situation more effectively by considering the **varying severity of crimes**.

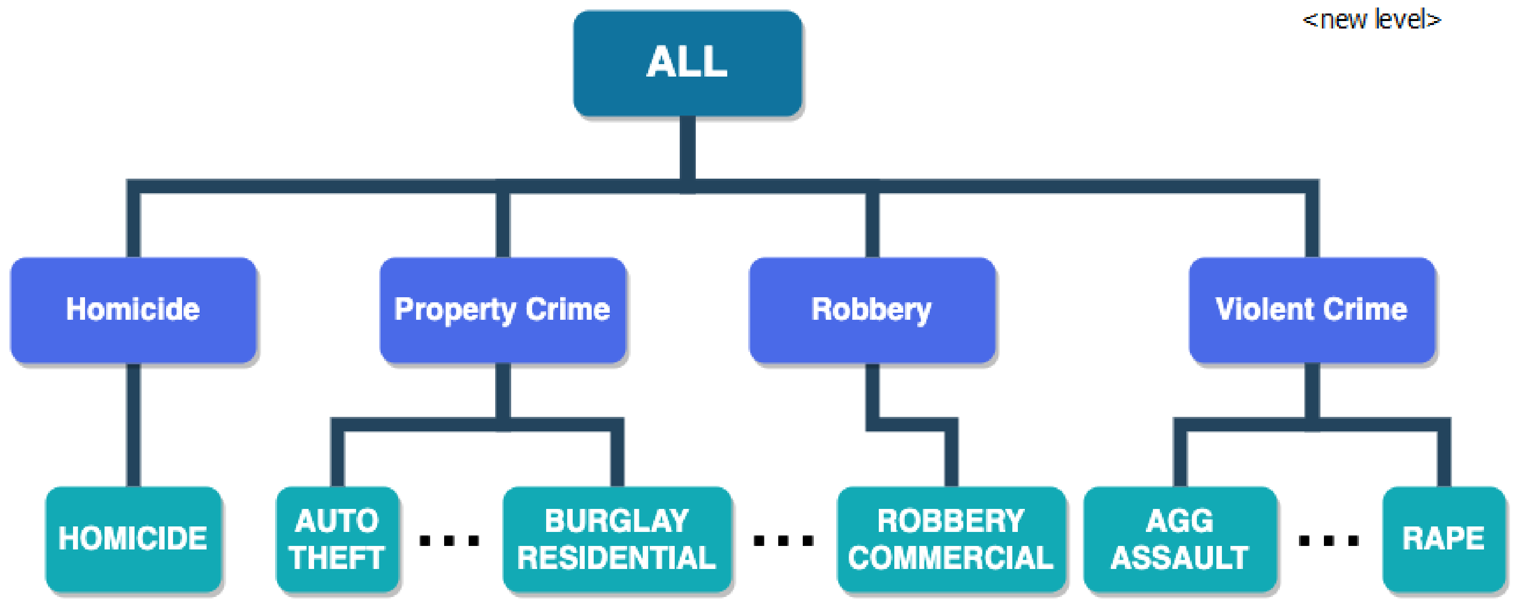
By **grouping similar offenses together**, we can identify patterns and trends in criminal activity, enabling more targeted interventions and policy decisions.

*(Federal Bureau of Investigation [FBI], n.d.)*

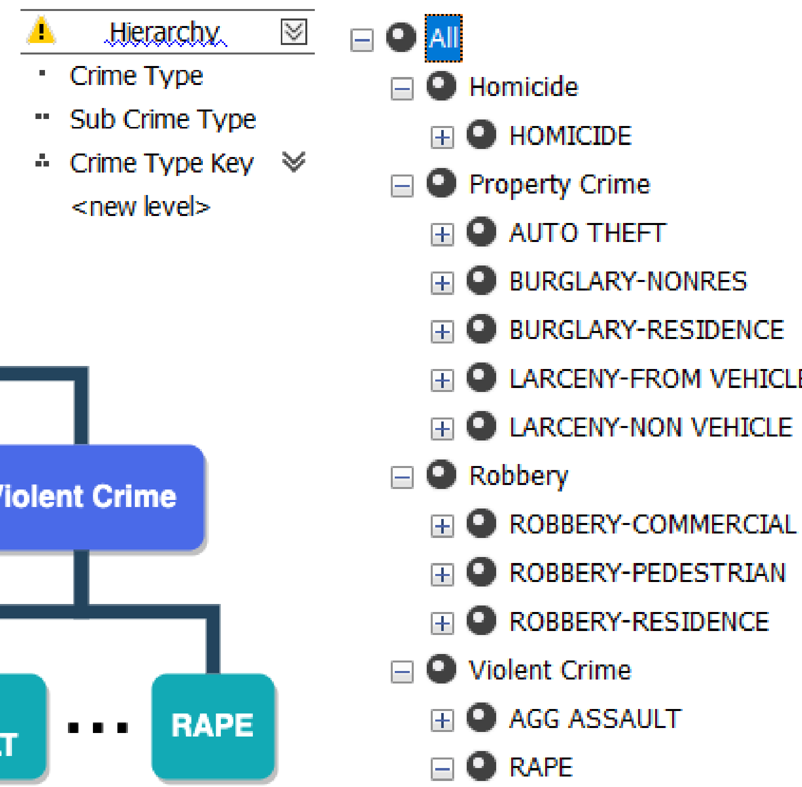
# Crime Type Hierarchy




*Schema Hierarchy*




*Concept Hierarchy*



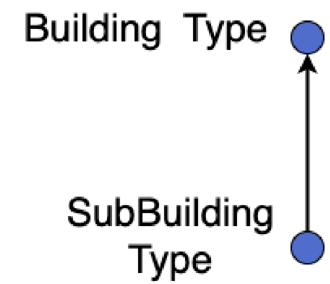

*Hierarchy in Visual Studio*

# Dimension Design

**Total order**:

**SubBuildingType < BuildingType**

**Building Type**



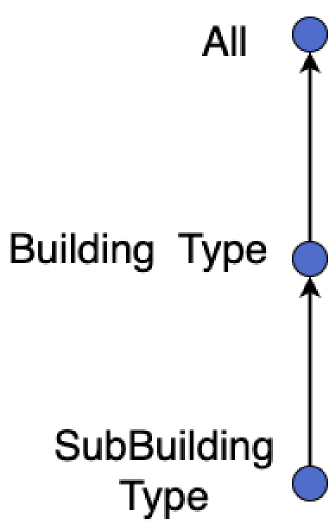

```python
def classify_type(x):
    if x in ['house_number', 'residential', 'neighbourhood', 'quarter']:
        return 'Residential'
    elif x in ['amenity', 'shop', 'retail', ]:
        return 'Commercial/Service'
    elif x in ['building', 'office', 'man_made']:
        return 'Office'
    elif x in ['healthcare', 'emergency']:
        return 'Medical'
    elif x in ['road', 'highway', 'railway', 'aeroway']:
        return 'Transportation'
    elif x in ['tourism', 'historic']:
        return 'Tourism'
    elif x in ['club','leisure']:
        return 'Entertainment'
    elif x in ['city', 'town', 'suburb', 'hamlet', 'county']:
        return 'Urban/Rural'
    else:
        return 'Other'
```

This two-level categorization helps in analyzing crime patterns across **various types of structures**, providing valuable insights into the security landscape in different building environments.

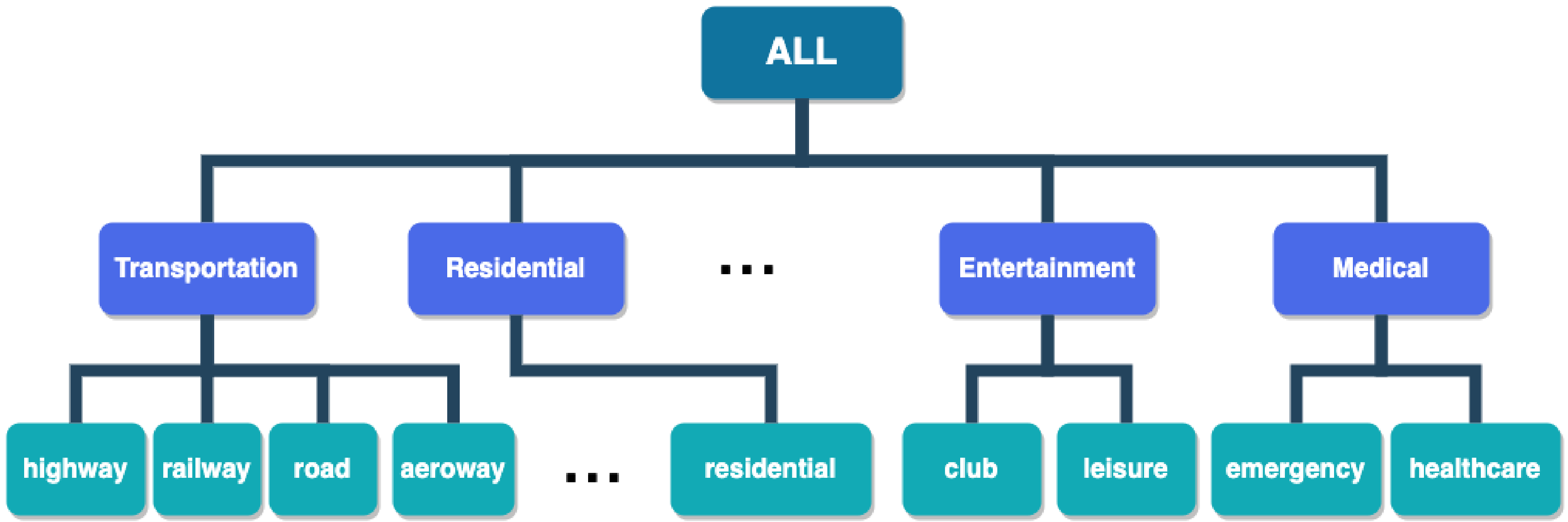
By categorizing buildings into groups like Residential, Commercial/Service, and Office, we can identify and focus on **specific areas that may require targeted crime prevention measures**.

*Category suggestions courtesy of ChatGPT. (OpenAI, 2021)*

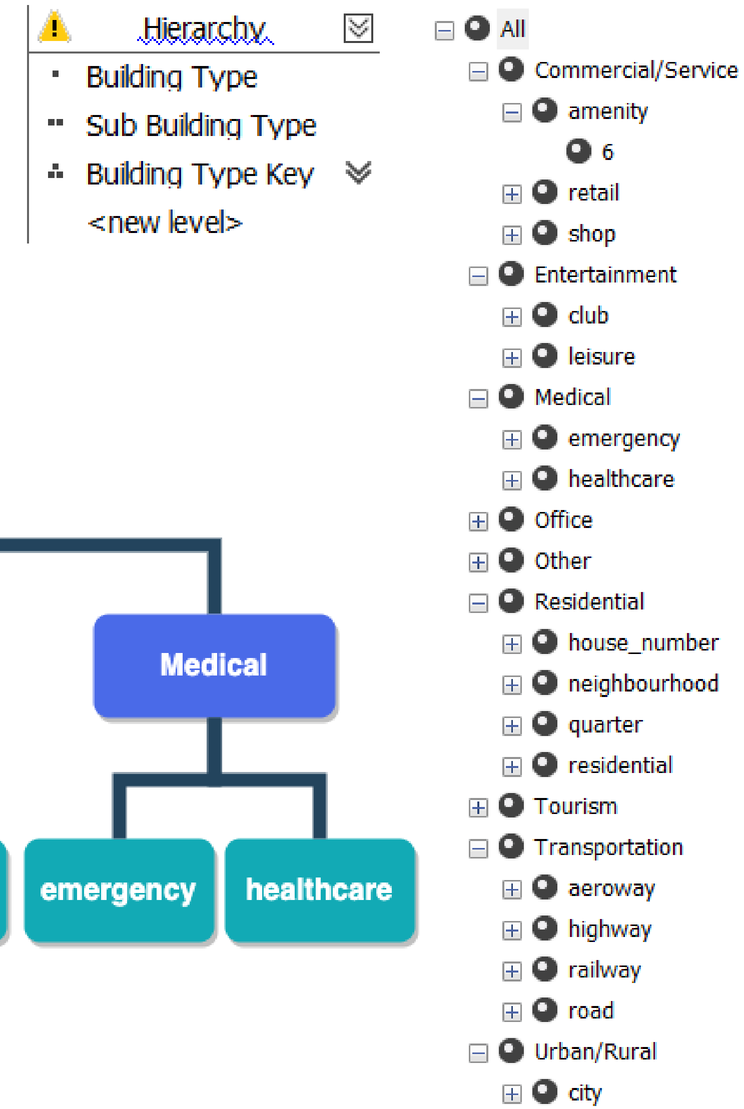
# Building Type Hierarchy




*Schema Hierarchy*



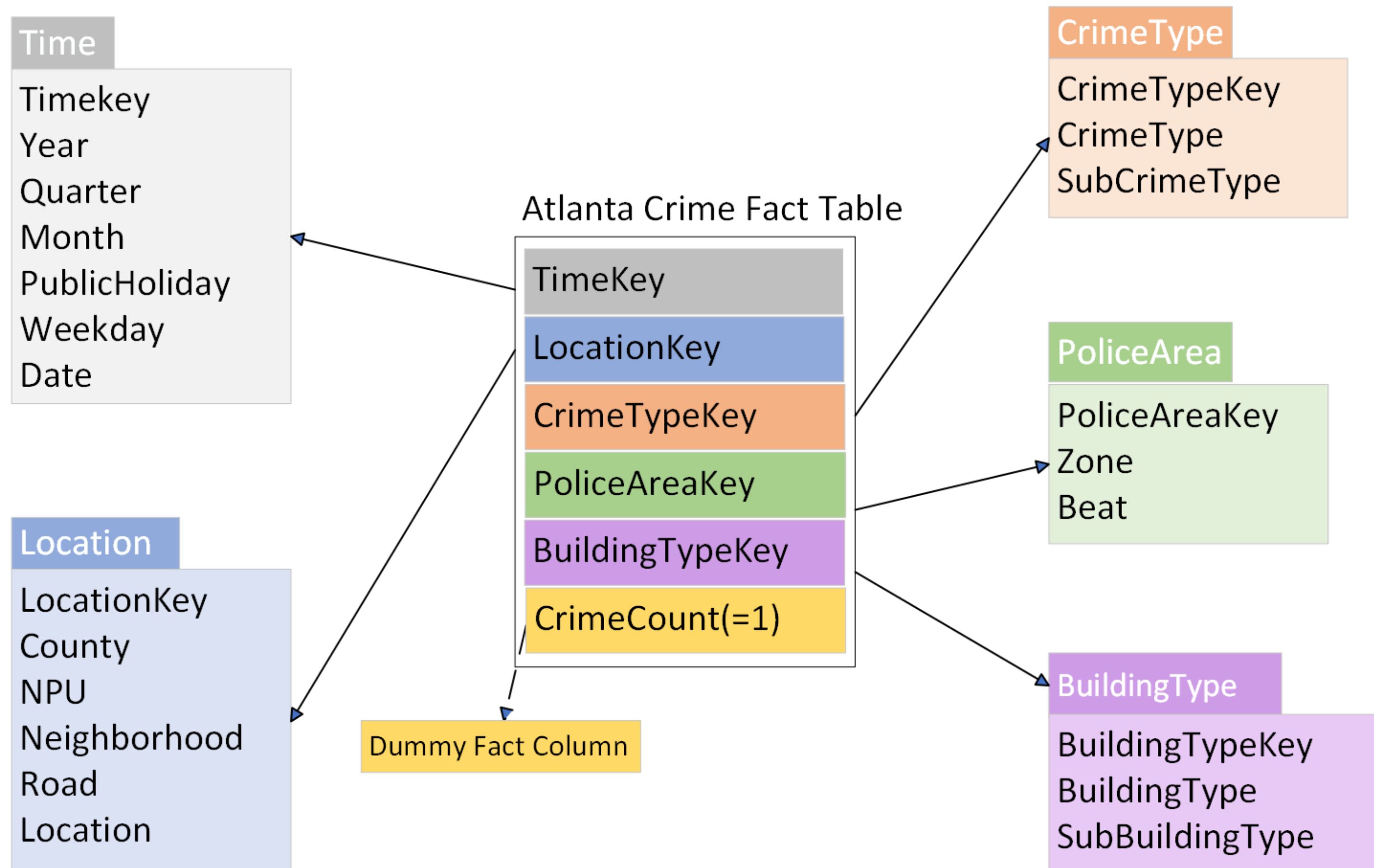

*Concept Hierarchy*




*Hierarchy in Visual Studio*

# Fact Table Design

## Fact Table




This structure allows for efficient querying and analysis of crime data across multiple dimensions.

## Surrogate Key

**Each table** is assigned a **unique surrogate key**, enhancing data consistency, integrity, and management. These system-generated keys improve data warehouse performance and enable efficient querying across **all dimensions**.

# Transform

**Time**

| date |
|---|
| 10/31/2010 |

## DimTime

| Year | Quarter | Month | PublicHoliday | Weekday | Date | TimeKey |
|---|---|---|---|---|---|---|
| 2010 | Q4 | October | NotHoliday | Sunday | 2010-10-31 | 1 |

**Zone**

| beat |
|---|
| 509 |
| 401 |
| 301 |
| 307 |
| 604 |

## DimPoliceArea

| Zone | Beat | PoliceAreaKey |
|---|---|---|
| Zone5 | 509 | 1 |
| Zone4 | 401 | 2 |
| Zone3 | 301 | 3 |
| Zone3 | 307 | 4 |
| Zone6 | 604 | 5 |

**Building Type**

| type |
|---|
| house_number |

## DimBuildingType

| BuildingType | SubBuildingType | BuildingTypeKey |
|---|---|---|
| Residential | house_number | 1 |

# Transform

**Crime Type**

| crime |
|---|
| LARCENY-NON VEHICLE |

## DimCrimeType

| CrimeType | SubCrimeType | CrimeTypeKey |
|---|---|---|
| Property Crime | LARCENY-NON VEHICLE | 1 |

**Location**

| location | beat | neighborho | npu | lat | long | type | road | neighbourhood_lc | city | county | state | postcode | country |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 610 SPRING | 509 | Downtown | M | 33.77101 | -84.38895 | house_num | Spring Street Northwest | | Atlanta | Fulton County | Georgia | 30308 | United States |

## DimLocation

| | County | NPU | Neighborhood | Road | Location | LocationKey |
|---|---|---|---|---|---|---|
| 0 | Fulton County | M | Downtown | Spring Street Northwest | 610 SPRING ST NW | 1 |

**Fact Table**

| TimeKey | LocationKey | CrimeTypeKey | PoliceAreaKey | BuildingTypeKey | CrimeCount | AtlantaCrimeID |
|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 2 | 2 | 2 | 2 | 1 | 2 |

# Loading

After data extraction and transformation, the data needs to be loaded into the **Microsoft SQL Server Management Studio (SSMS)** by SQL codes.

**Insert the data into the created tables:**

1) Create the targeted **database AtlantaCrimeDW** and **six tables** (**BuildLoadTables.sql**)
2) Set the environment variable to data path and insert the data into tables (**InsertData.sql**)

- AtlantaCrimeDW
  - Database Diagrams
  - Tables
    - System Tables
    - FileTables
    - External Tables
    - Graph Tables
    - dbo.DimBuildingType
    - dbo.DimCrimeType
    - dbo.DimLocation
    - dbo.DimPoliceArea
    - dbo.DimTime
    - dbo.FactAtlantaCrime

```sql
SELECT TOP (1000) [Year]
      ,[Quarter]
      ,[Month]
      ,[PublicHoliday]
      ,[Weekday]
      ,[Date]
      ,[TimeKey]
  FROM [AtlantaCrimeDW].[dbo].[DimTime]
```

100 %

Results | Messages

| | Year | Quarter | Month | PublicHoliday | Weekday | Date | TimeKey |
|---|---|---|---|---|---|---|---|
| 1 | 2010 | Q4 | October | NotHoliday | Sunday | 2010-10-31 | 1 |
| 2 | 2010 | Q4 | October | NotHoliday | Saturday | 2010-10-30 | 2 |
| 3 | 2010 | Q4 | October | NotHoliday | Friday | 2010-10-29 | 3 |
| 4 | 2010 | Q4 | October | NotHoliday | Thursday | 2010-10-28 | 4 |
| 5 | 2010 | Q4 | October | NotHoliday | Wednesday | 2010-10-27 | 5 |

*Time Dimension*

```sql
SELECT TOP (100) [TimeKey]
      ,[LocationKey]
      ,[CrimeTypeKey]
      ,[PoliceAreaKey]
      ,[BuildingTypeKey]
      ,[CrimeCount]
      ,[AtlantaCrimeID]
  FROM [AtlantaCrimeDW].[dbo].[FactAtlantaCrime]
```

sults | Messages

| TimeKey | LocationKey | CrimeTypeKey | PoliceAreaKey | BuildingTypeKey | CrimeCount | AtlantaCrimeID |
|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 2 | 2 | 2 | 2 | 1 | 2 |
| 1 | 3 | 3 | 3 | 3 | 1 | 3 |
| 1 | 4 | 2 | 4 | 1 | 1 | 4 |
| 1 | 5 | 1 | 5 | 1 | 1 | 5 |

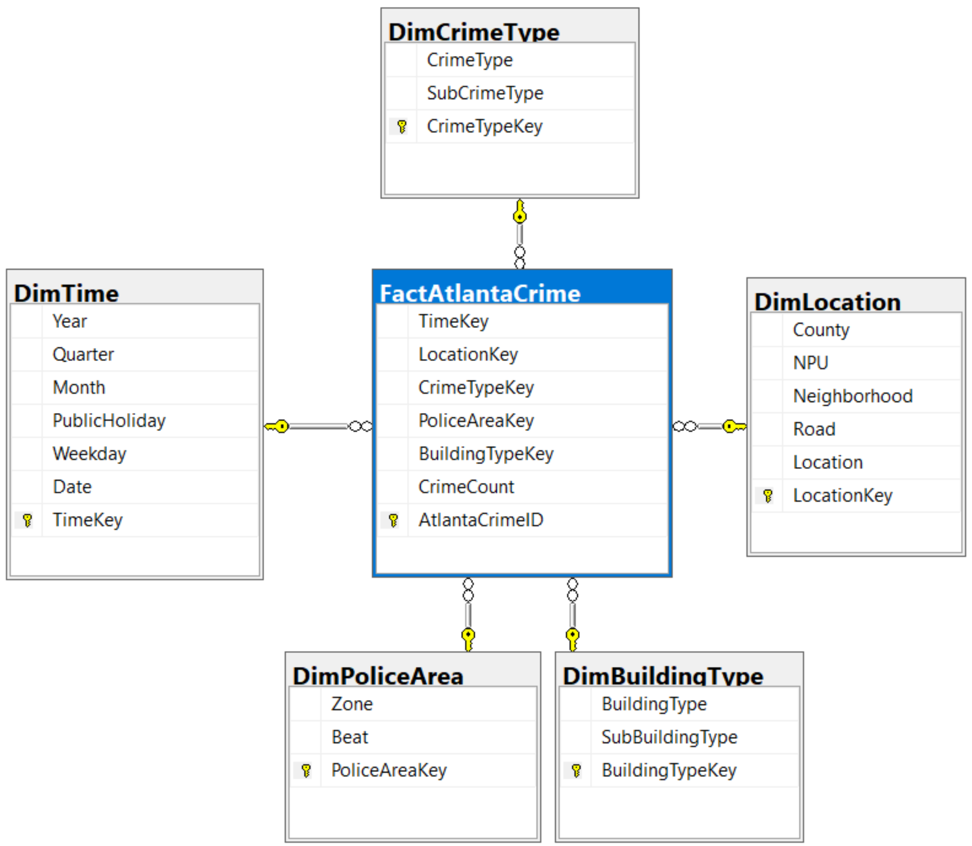
*Fact Table*

# OLAP Cube Design

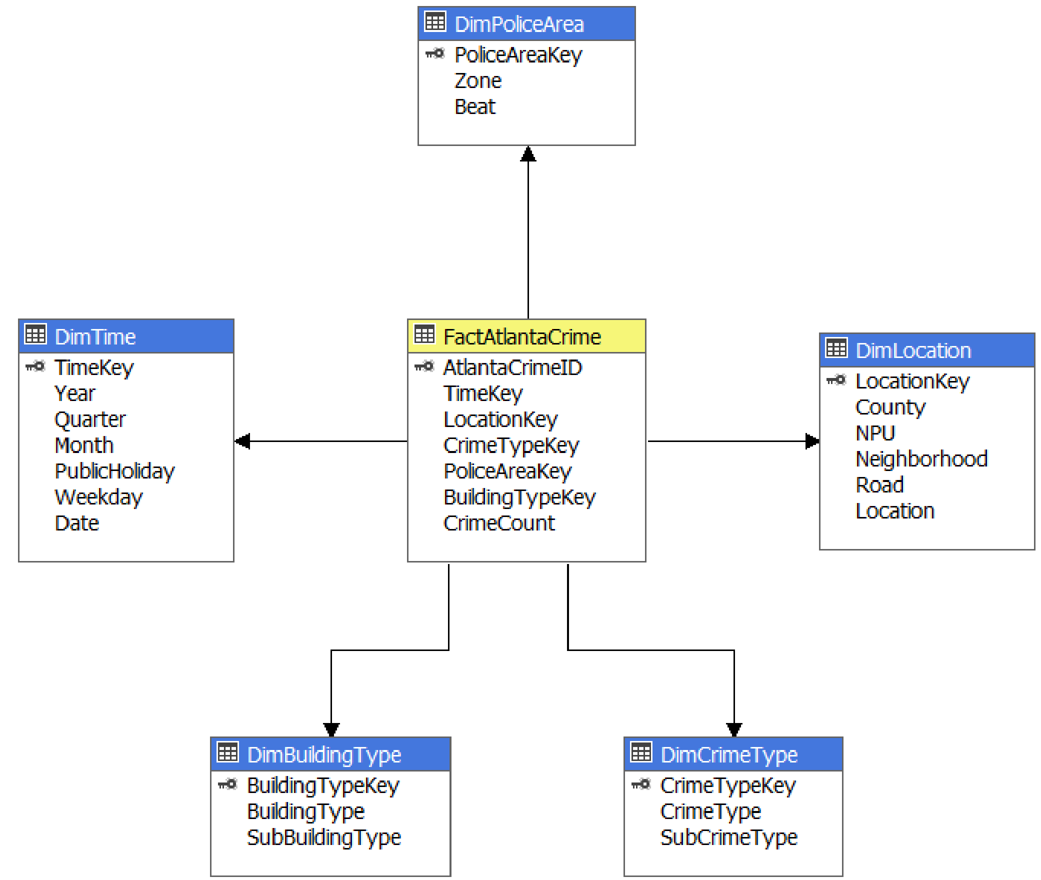
In order to answer the business queries, a new multi-dimensional project with a cube of five dimensions was created in **Microsoft Visual Studio**.

Unique values were populated for each dimension, and concept hierarchies were established.

## ER Diagram

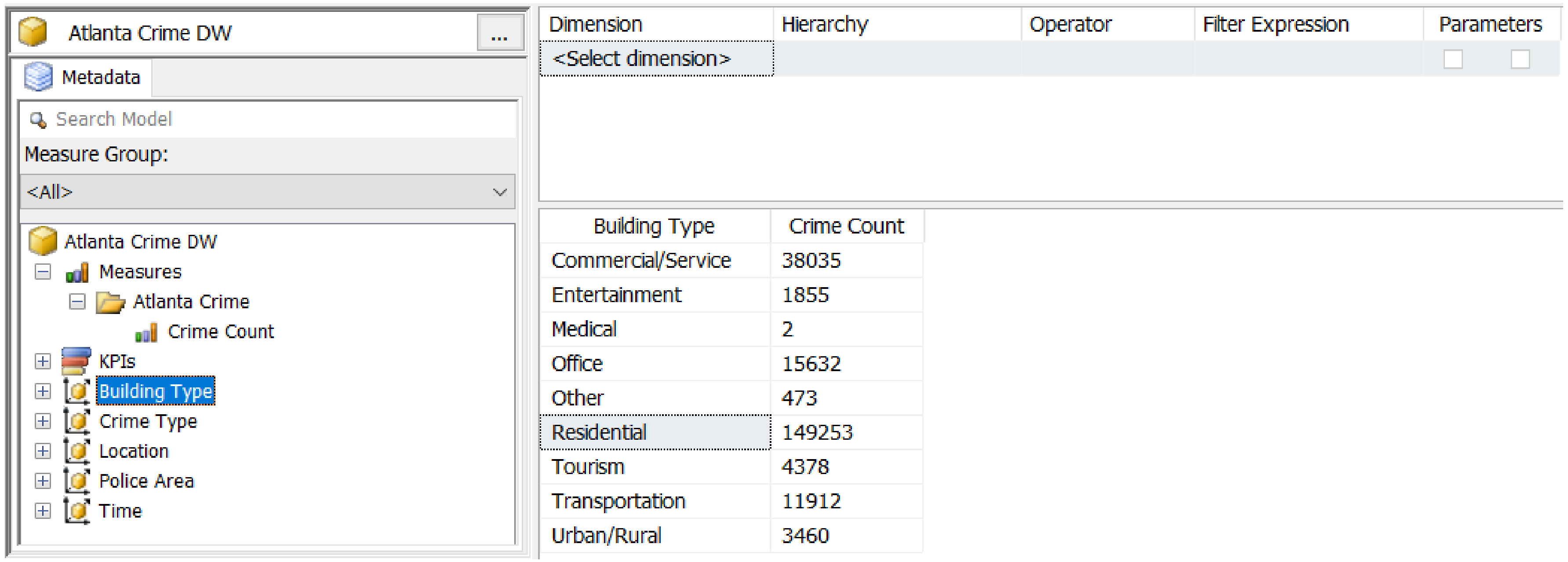



## Data Cube



# Q1: Which type of building have the highest crime count?

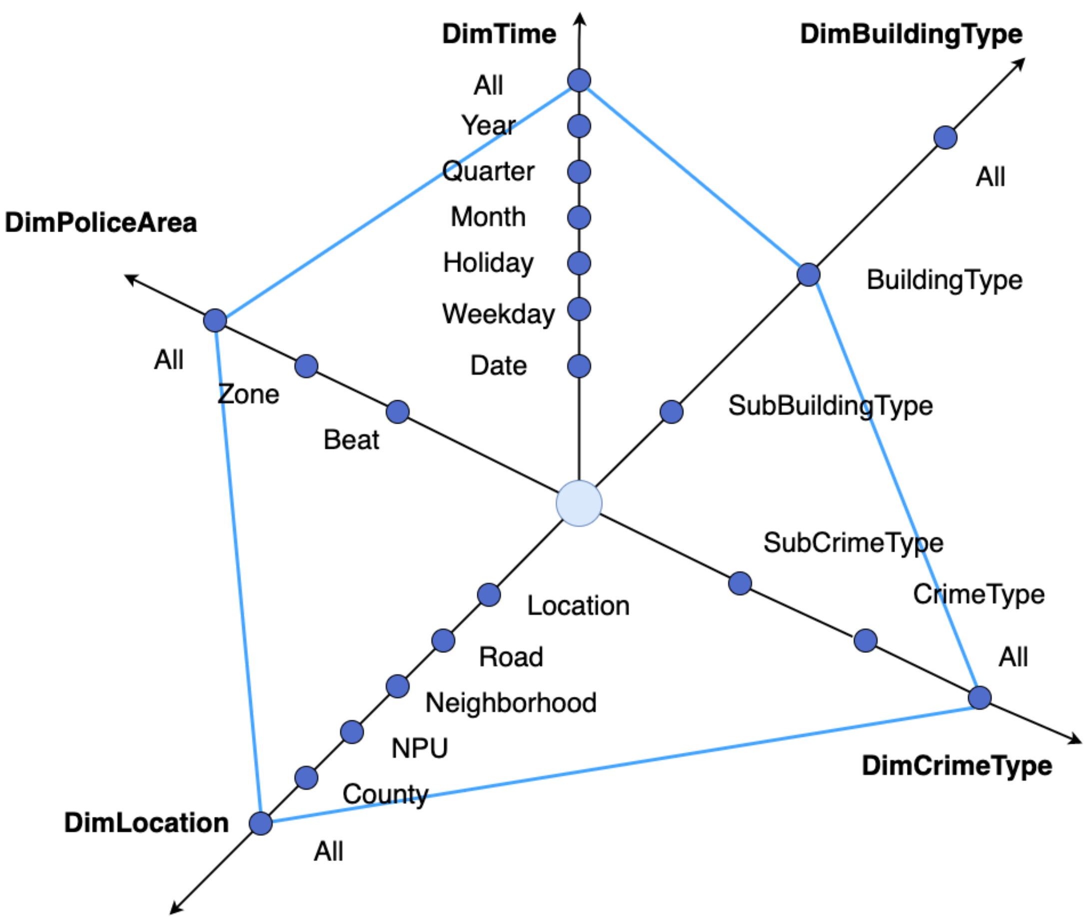
## Use multi-dimensional cube to Analyze Building Types with Highest Crime Count

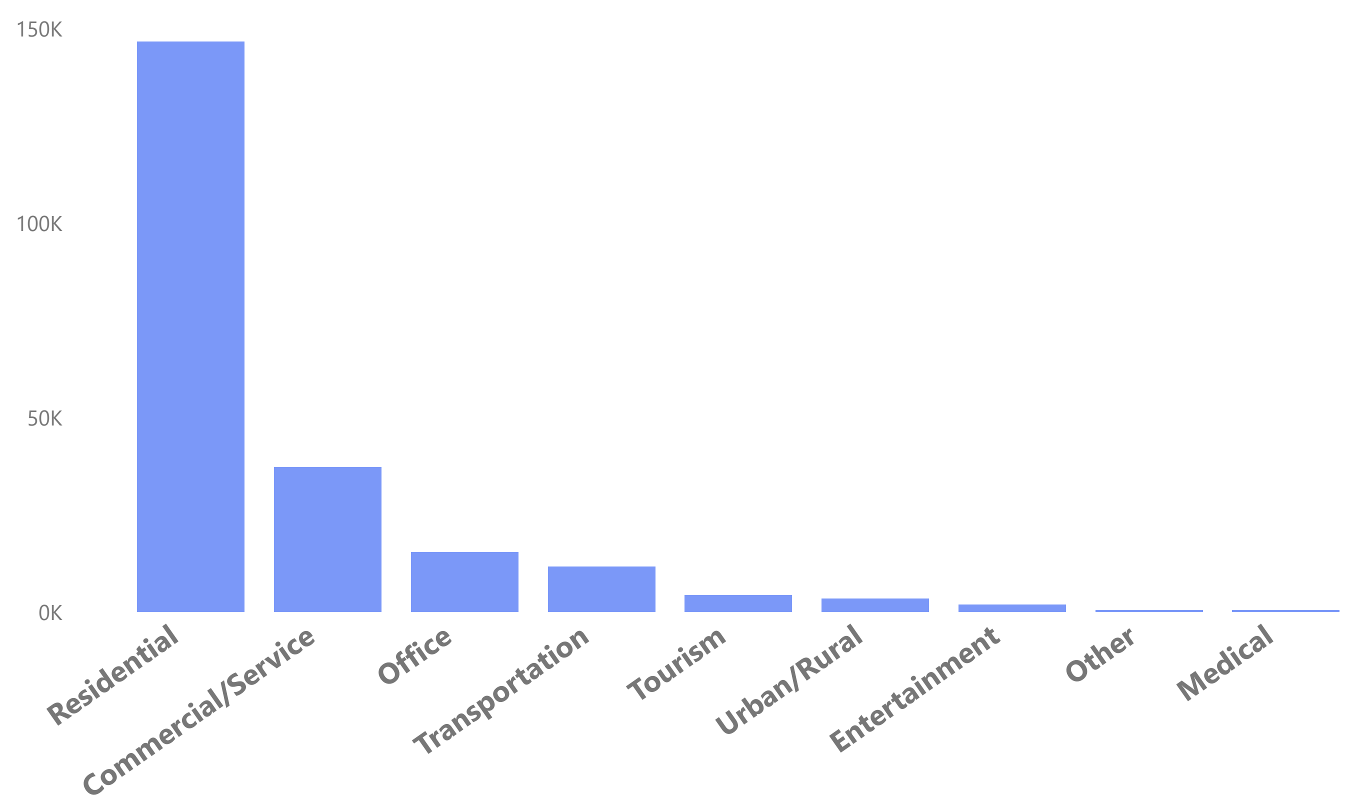



Performing a **roll-up** operation on the BuildingTypeDim by aggregating the CrimeCount at the BuildingType level, allowing us to identify the **building category** with the highest recorded crime incidents.

# Q1: Which type of building have the highest crime count?

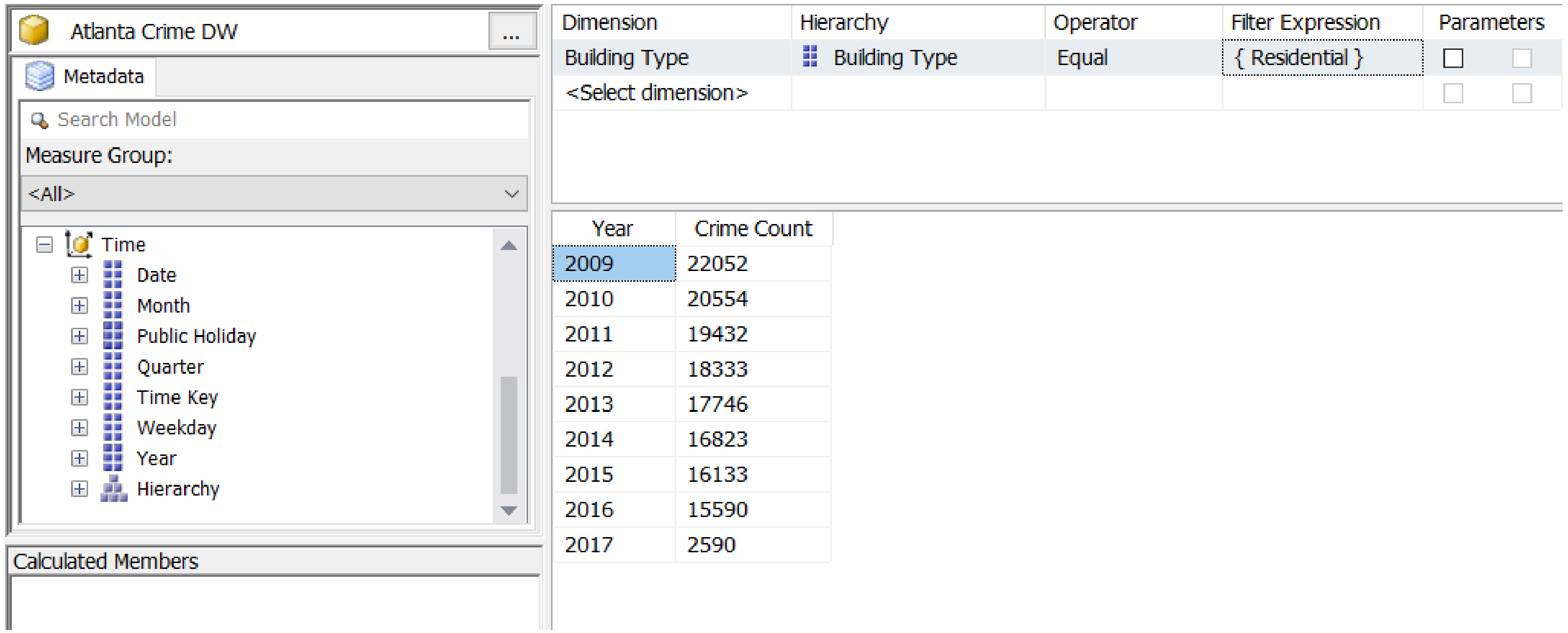
In this dataset, the number of crimes occuring in **Residential** areas is significantly higher than other building types, which total count is almost reach **150k**.



# Q2: What the trend of residential area crime count per year?

## Use multi-dimensional cube to Analyze Annual Trend of Crime Count in Residential Areas

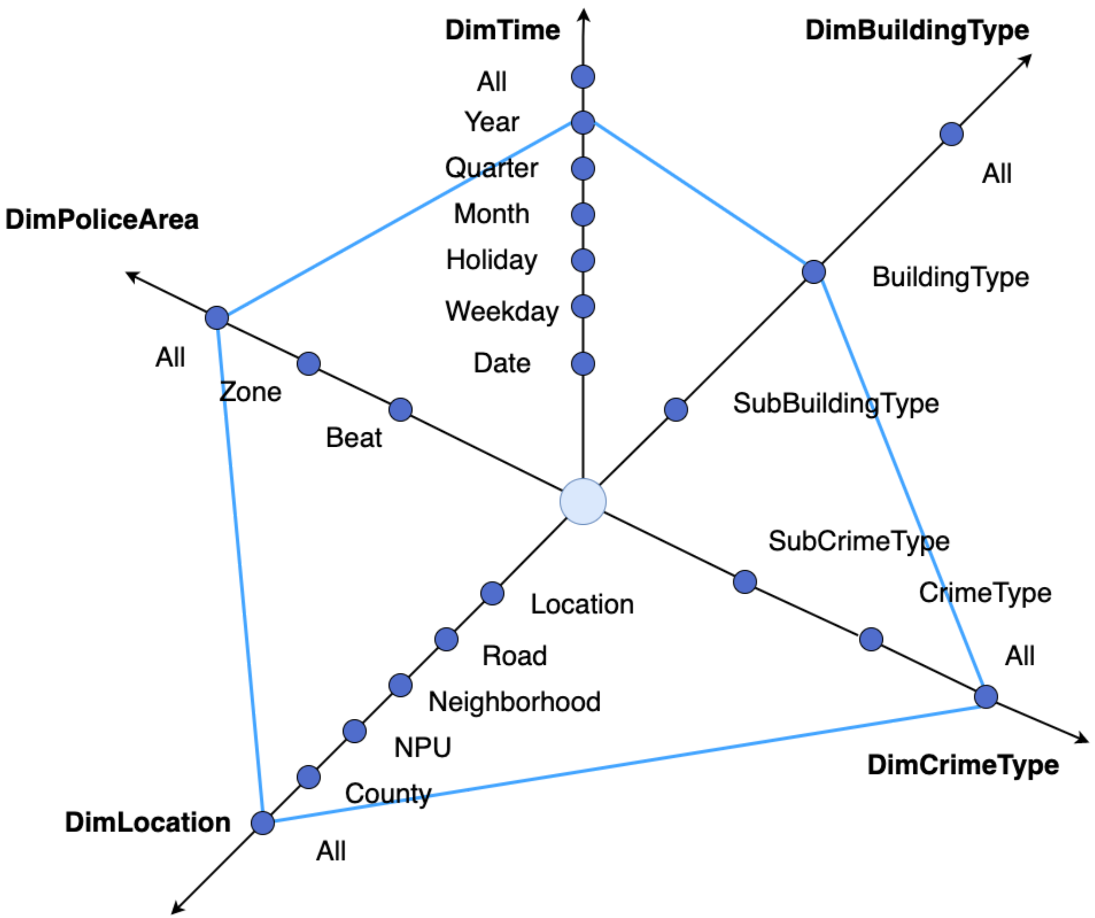


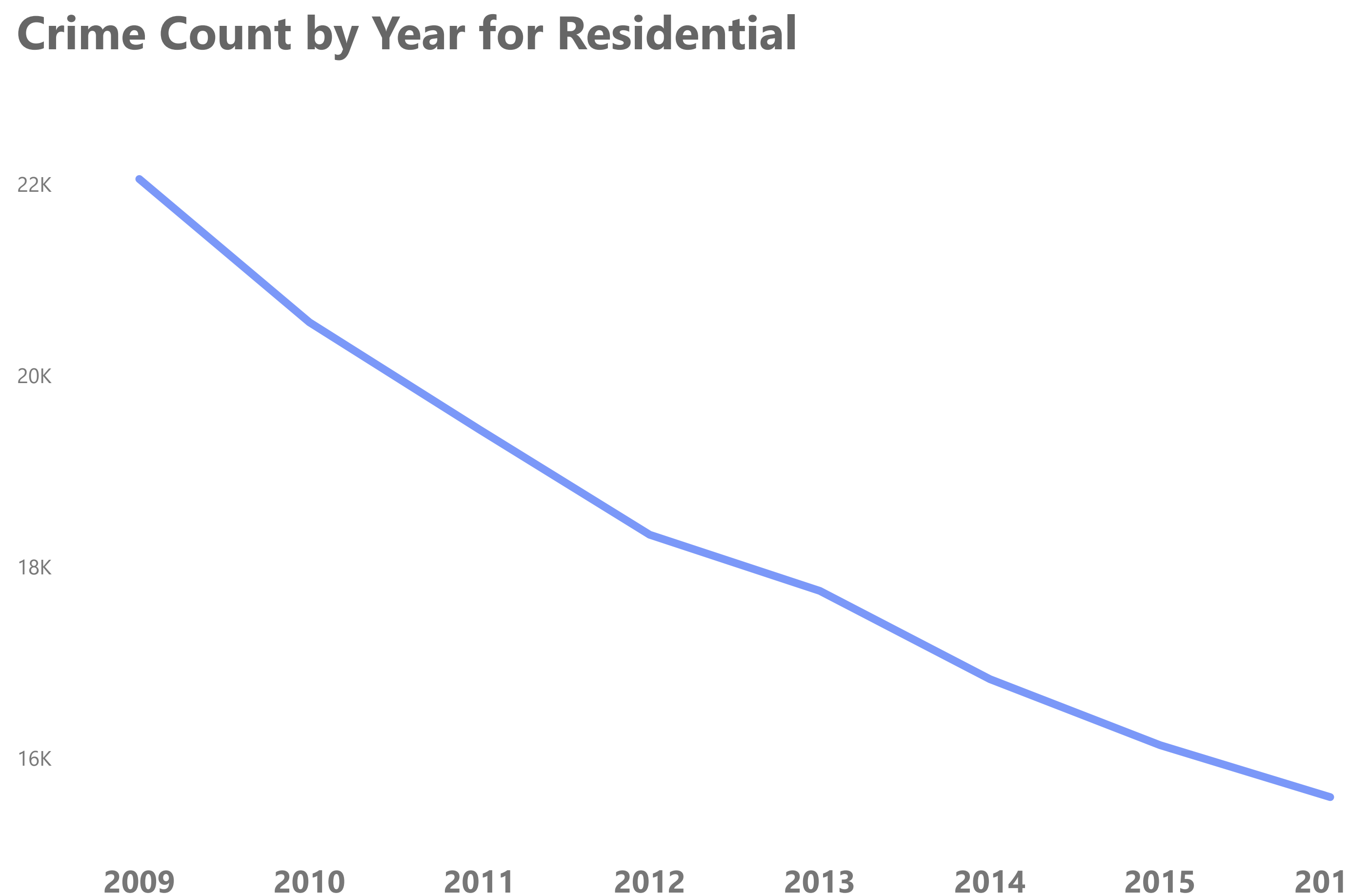

| Year | Crime Count |
|---|---|
| 2009 | 22052 |
| 2010 | 20554 |
| 2011 | 19432 |
| 2012 | 18333 |
| 2013 | 17746 |
| 2014 | 16823 |
| 2015 | 16133 |
| 2016 | 15590 |
| 2017 | 2590 |

To analyze the **yearly trend** of crime count in residential areas, applying a filter to the BuildingTypeDim, selecting only '**Residential**' buildings. Then, we performed a **roll-up** operation on the TimeDim at the Year level, aggregating the CrimeCount to visualize the **annual crime trend** in residential areas.

## Q2: What the trend of residential area crime count per year?

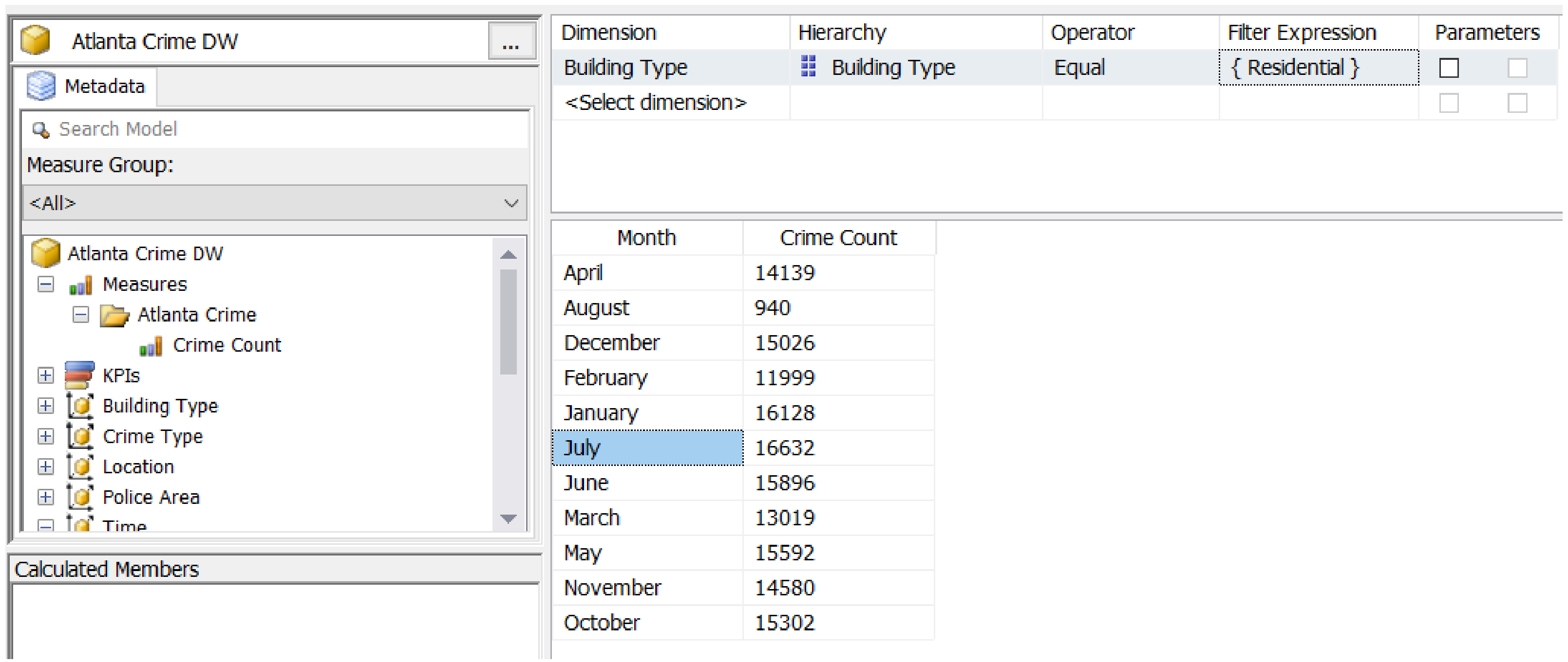
It is apparent that the number of crimes in residential areas has been **decreasing annually** as the years progress.



# Q2 Extension: Which month had the highest count of crime in residential area?

## Multi-dimensional cube Analyzing Month with the Highest Crime Count in Residential Area

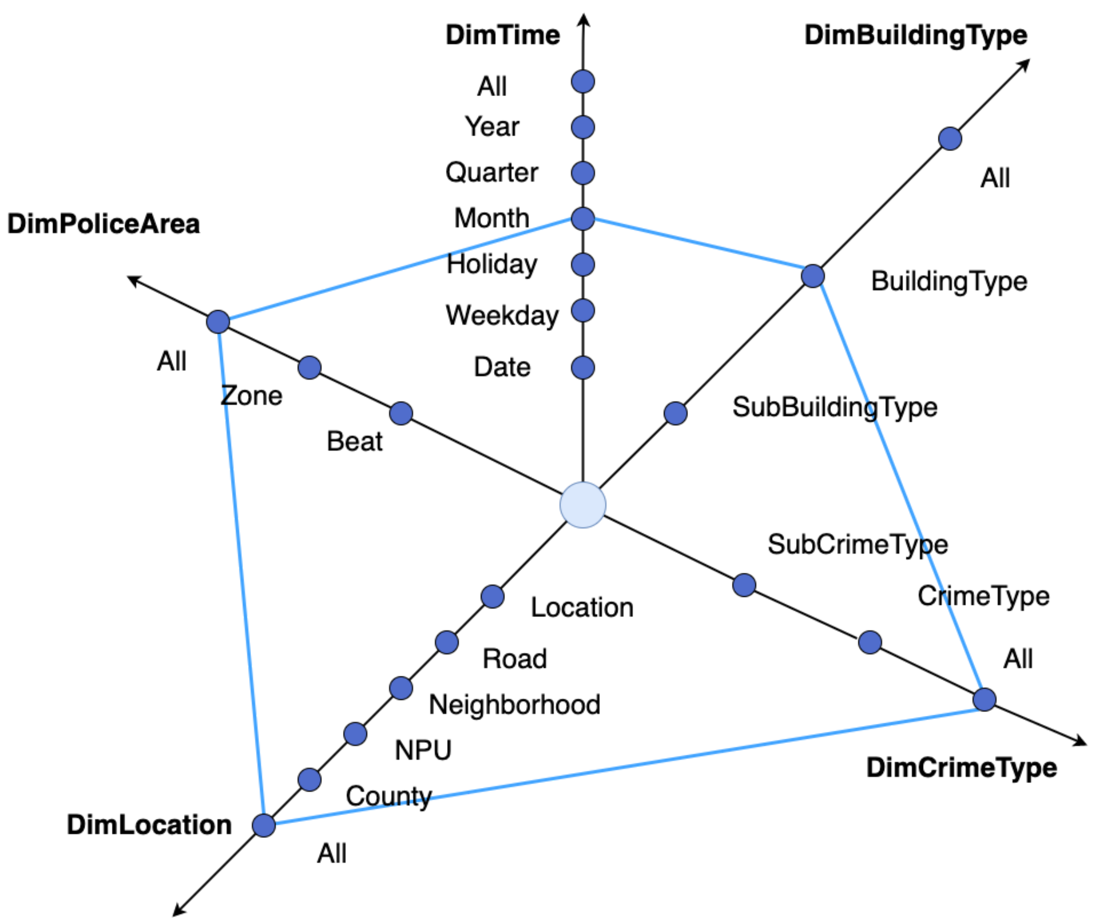


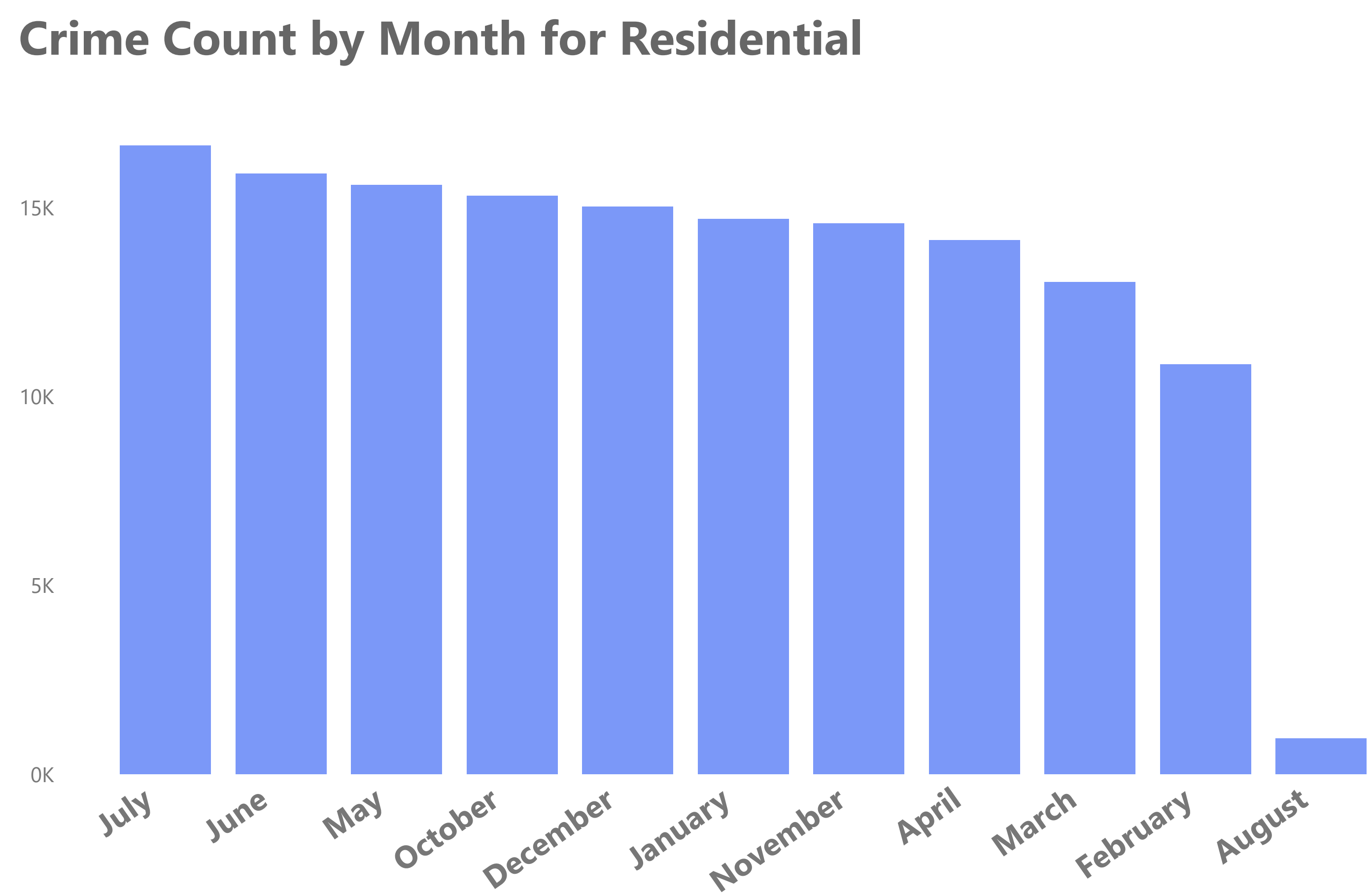

| Month | Crime Count |
|---|---|
| April | 14139 |
| August | 940 |
| December | 15026 |
| February | 11999 |
| January | 16128 |
| July | 16632 |
| June | 15896 |
| March | 13019 |
| May | 15592 |
| November | 14580 |
| October | 15302 |

Applying a filter to the BuildingTypeDim, selecting '**Residential**' buildings only. We then performed a **drill-down** operation on the TimeDim at the Month level, aggregating the CrimeCount to pinpoint the **month** with the highest recorded crime incidents in residential areas.

# Q2 Extension: Which month had the highest count of crime in residential area?

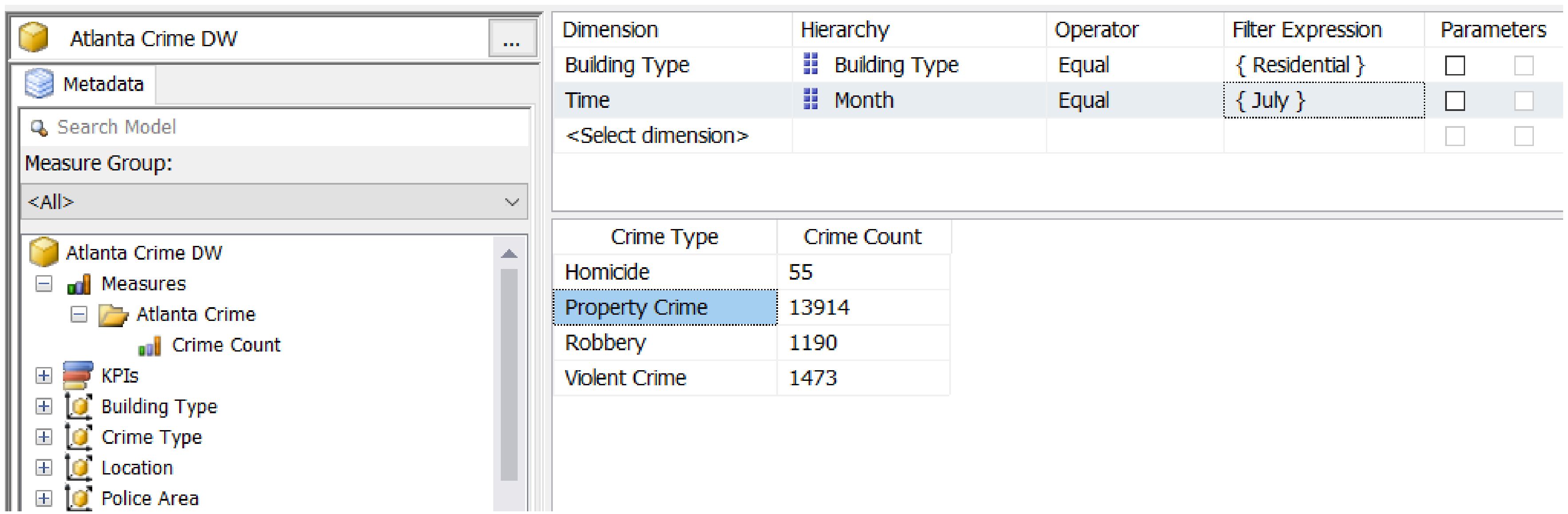
The gragh indicates that **July** is **peak period** for cimes occuring in residential areas.



# Q3: What is the distribute of crime types in residential areas during July?

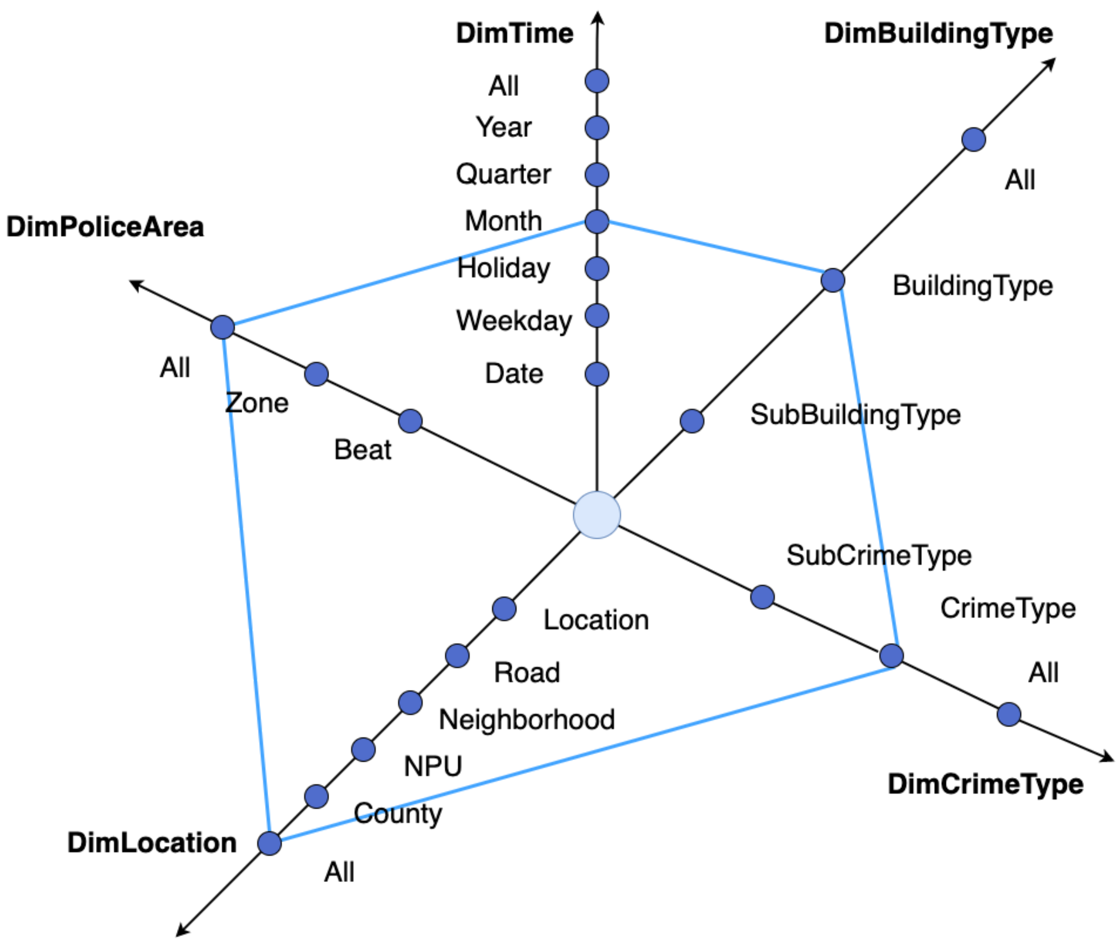
## Use multi-dimensional cube to Analyze Crime Type Distribution in Residential Areas during July

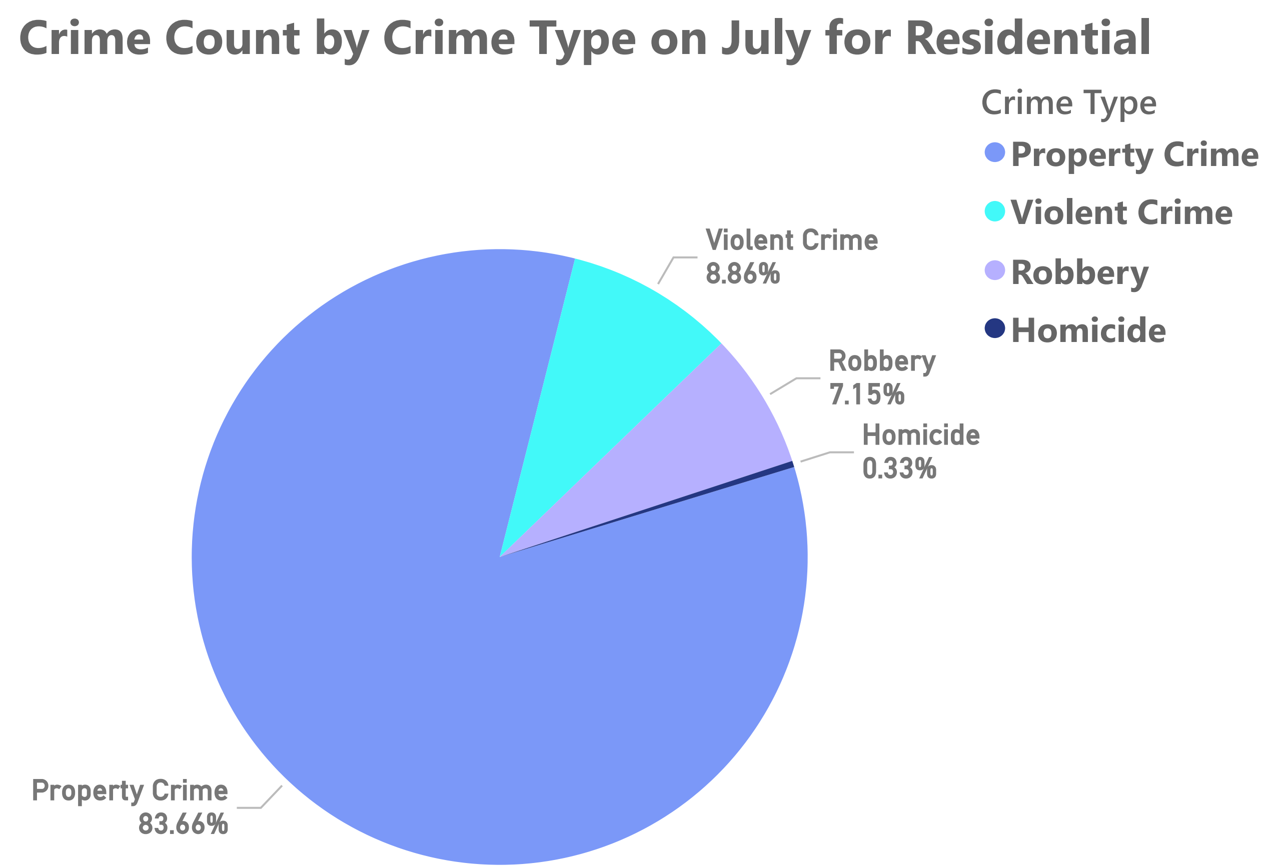



Applying filters to the BuildingTypeDim and TimeDim, selecting '**Residential**' buildings and the month of '**July**'. Next, we performed a **roll-up** operation on the CrimeTypeDim at the CrimeType level, aggregating the CrimeCount to reveal the **distribution of various crime types** in residential areas for the selected month.

# Q3: What is the distribute of crime types in residential areas during July?

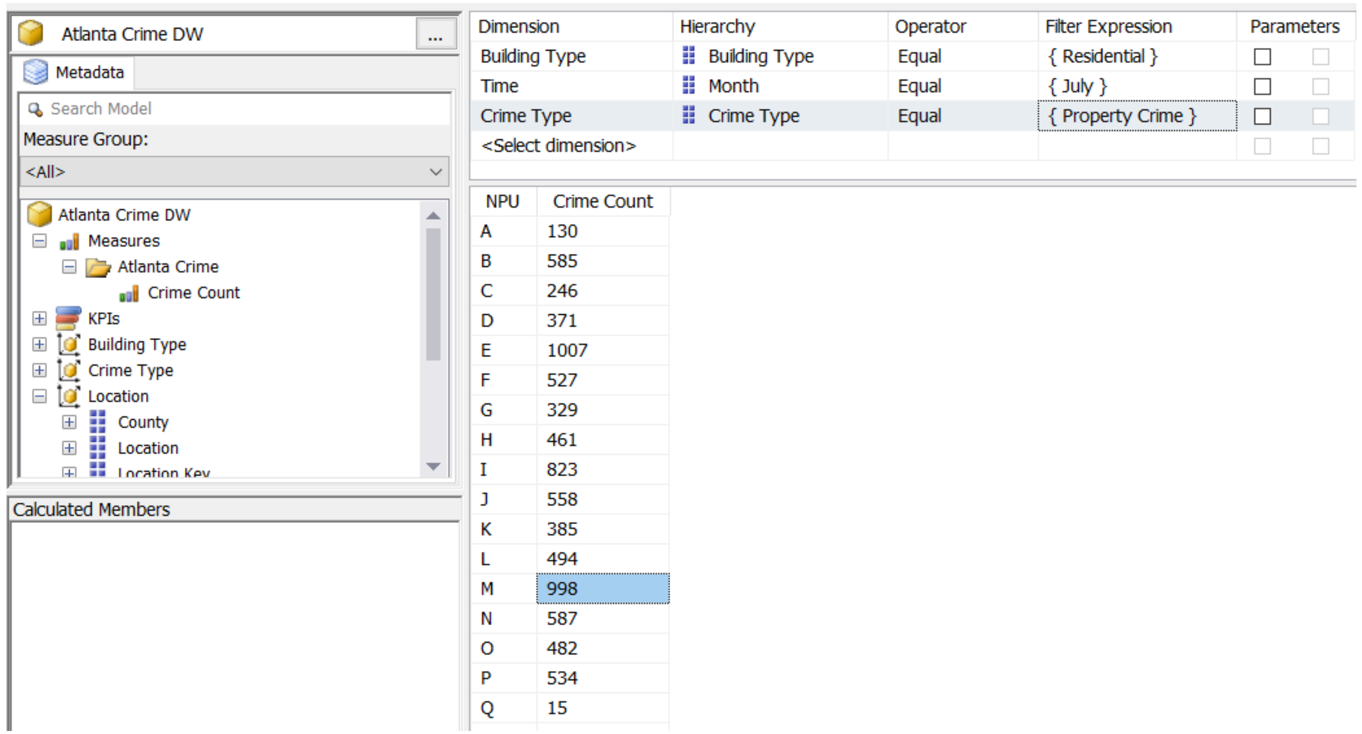
The pie chart displays the proportion of different crime types, where **Property Crime** account for the largest portion at **83.66%** during July.



# Q4: Which NPU had the highest count of property crime in residential areas during July?

## Use multi-dimensional cube to facilitatie the roll-up analysis



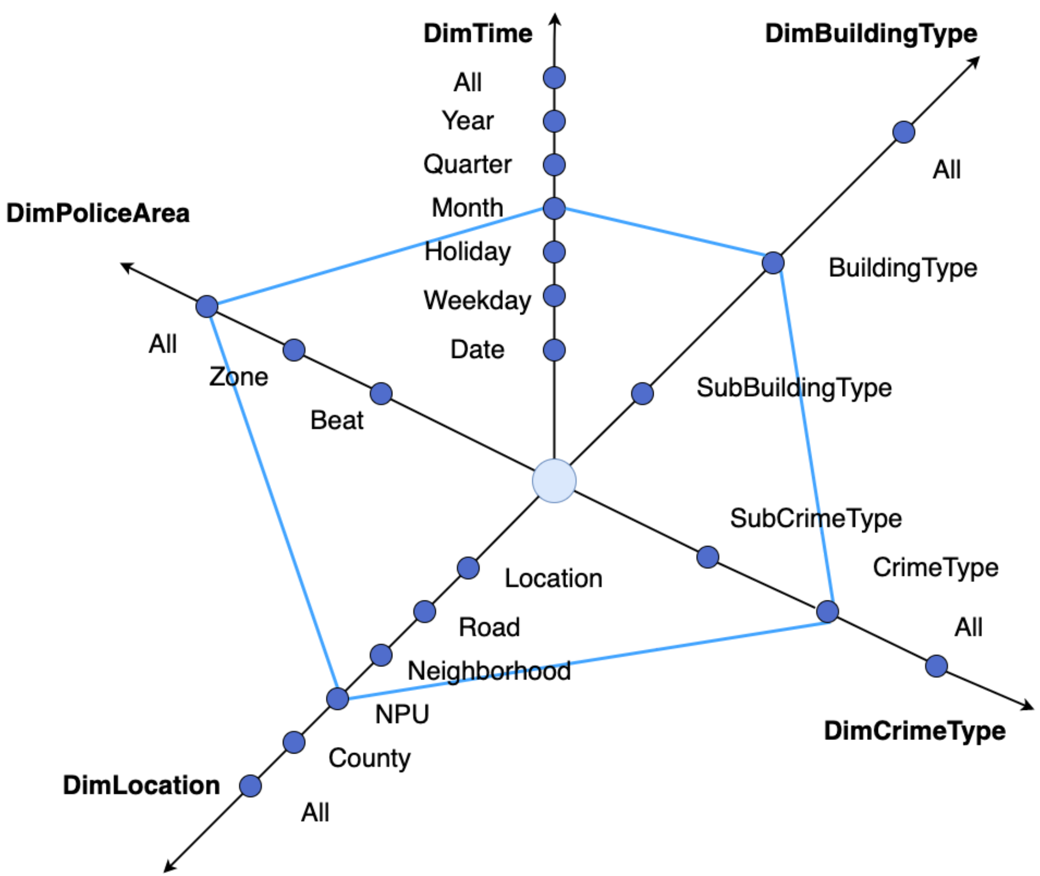

| Dimension | Hierarchy | Operator | Filter Expression | Parameters |
|---|---|---|---|---|
| Building Type | Building Type | Equal | { Residential } | |
| Time | Month | Equal | { July } | |
| Crime Type | Crime Type | Equal | { Property Crime } | |
| <Select dimension> | | | | |

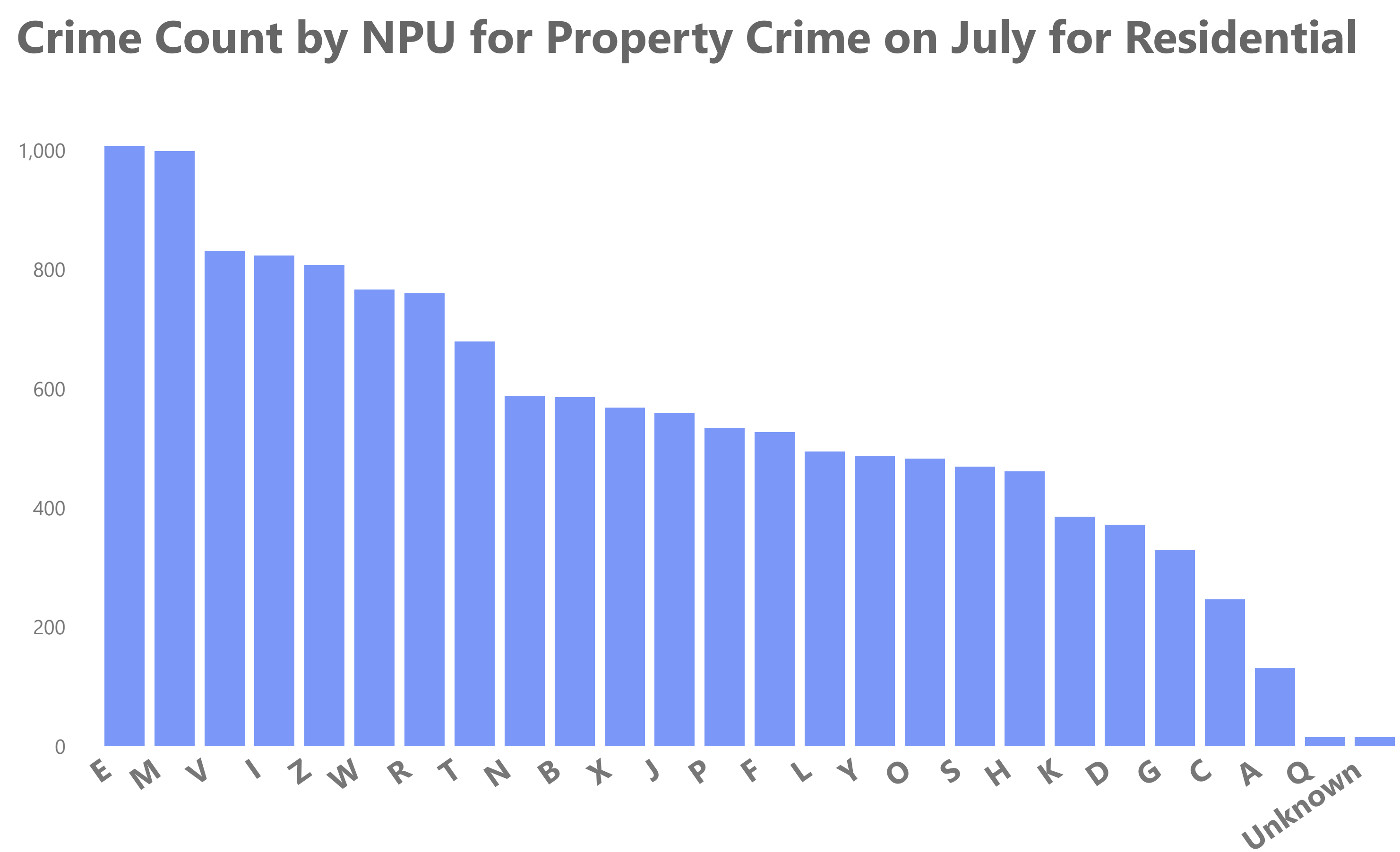
| NPU | Crime Count |
|---|---|
| A | 130 |
| B | 585 |
| C | 246 |
| D | 371 |
| E | 1007 |
| F | 527 |
| G | 329 |
| H | 461 |
| I | 823 |
| J | 558 |
| K | 385 |
| L | 494 |
| M | 998 |
| N | 587 |
| O | 482 |
| P | 534 |
| Q | 15 |

Applying filters to the BuildingTypeDim for '**Residential**' buildings, TimeDim for the month of '**July**', and CrimeTypeDim for '**Property Crime**'. We then performed a **roll-up** operation on the LocationDim at the NPU level, aggregating the CrimeCount to pinpoint the **NPU** with the highest number of recorded property crimes in residential areas for the specified month.

# Q4: Which NPU had the highest count of property crime in residential areas during July?

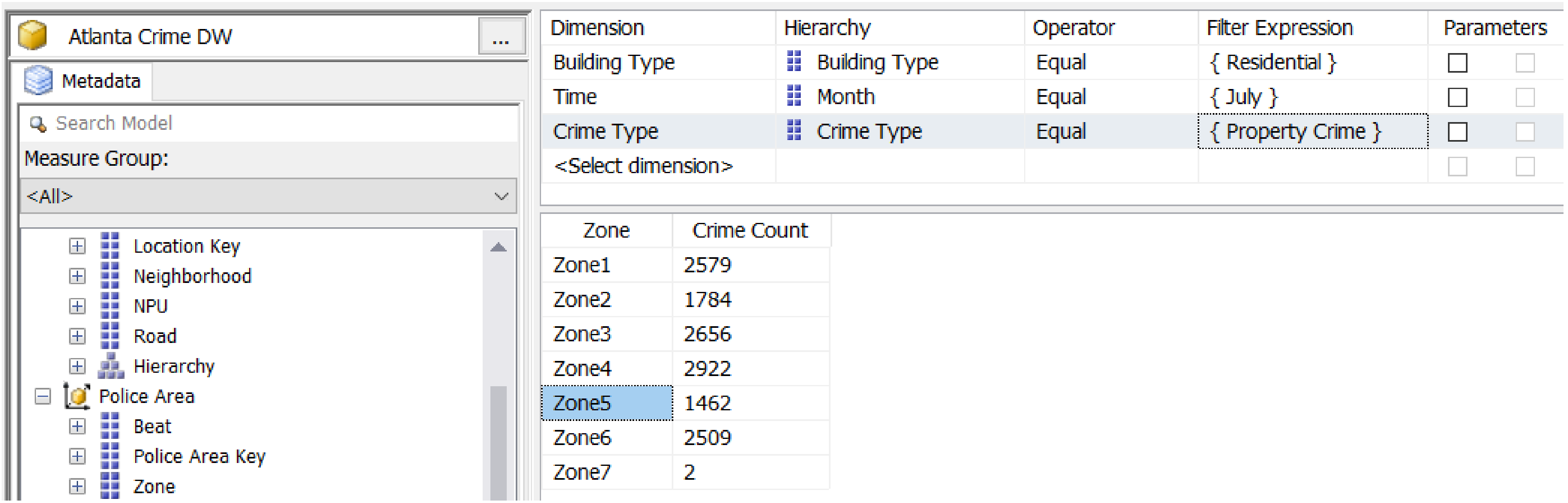
**Top 2** areas with the highest crime counts are the resdential area of **NPU E** and **NPU M** during July. Special attention shold be paid to the security situation in these two areas.



# Q5: Which zone had the highest count of property crime in residential areas during July?

## Use multi-dimensional cube to facilitatie the roll-up analysis



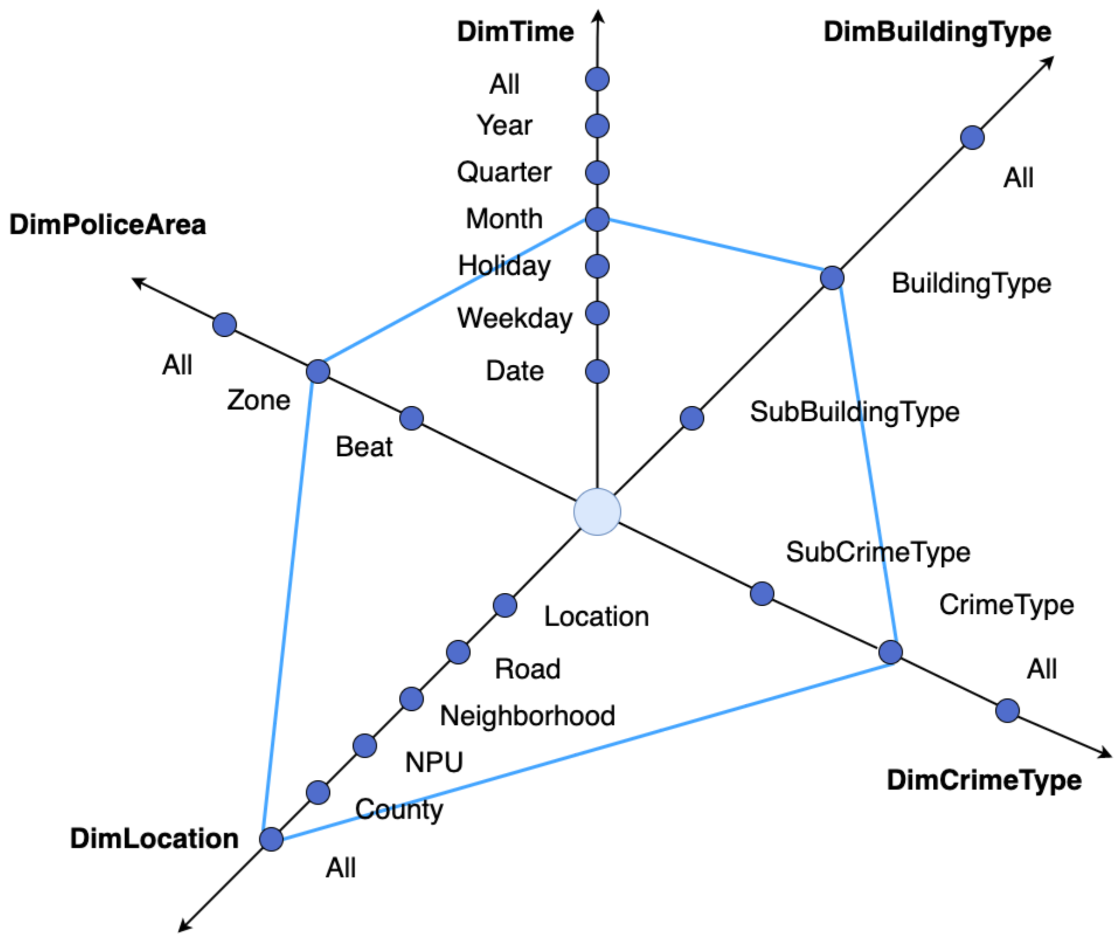

| Dimension | Hierarchy | Operator | Filter Expression | Parameters |
|---|---|---|---|---|
| Building Type | Building Type | Equal | { Residential } | |
| Time | Month | Equal | { July } | |
| Crime Type | Crime Type | Equal | { Property Crime } | |
| <Select dimension> | | | | |

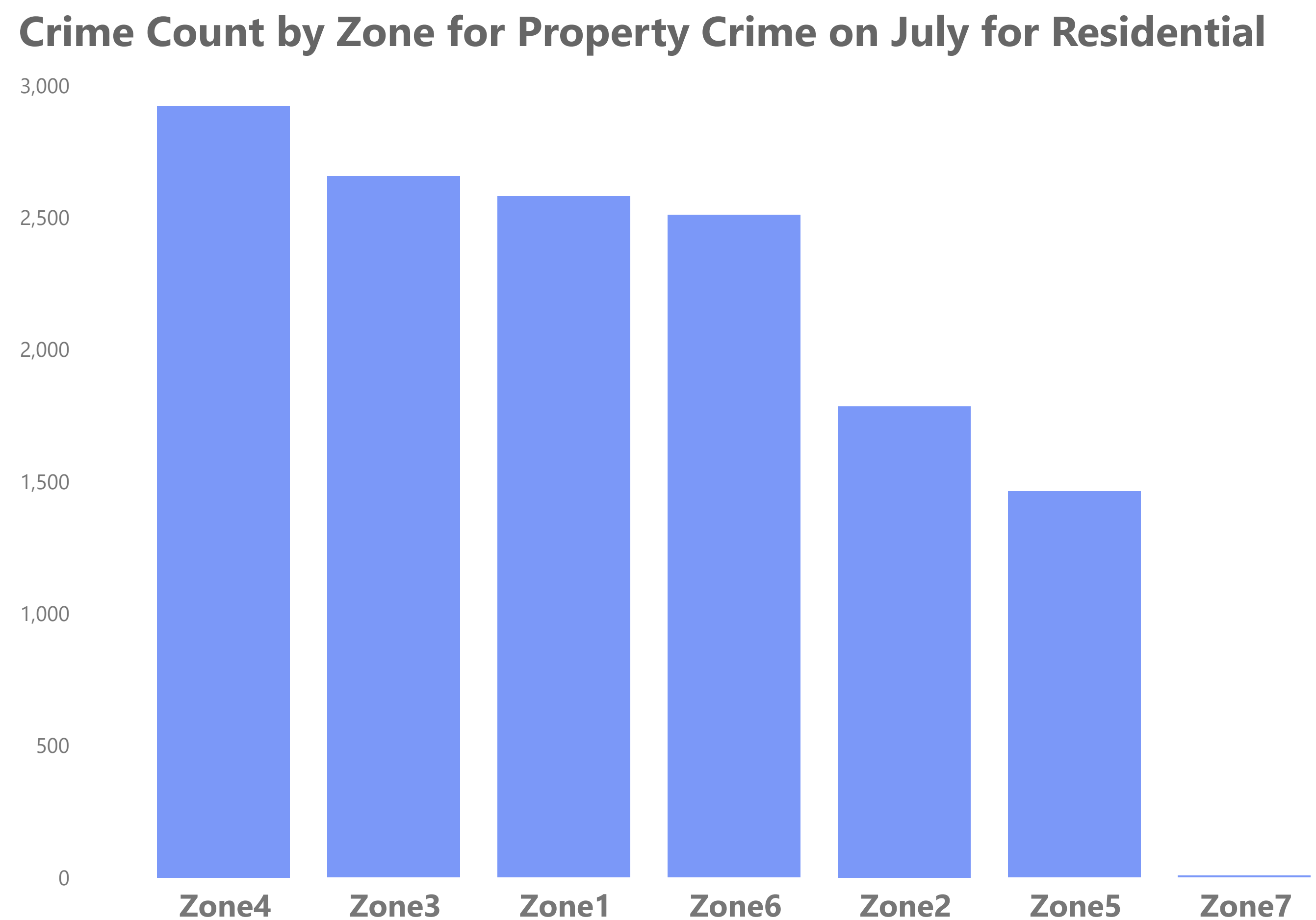
| Zone | Crime Count |
|---|---|
| Zone1 | 2579 |
| Zone2 | 1784 |
| Zone3 | 2656 |
| Zone4 | 2922 |
| Zone5 | 1462 |
| Zone6 | 2509 |
| Zone7 | 2 |

Applying multiple filters to Crime data, selecting only '**Residential**' buildings in the BuildingTypeDim, filtering by **July** in the TimeDim, and selecting '**Property Crime**' in the CrimeTypeDim. We then performed a **roll-up** operation on the Police Area Dim, selecting the **Zone** dimension to visualize which specific police zones had the highest recorded incidents of property crime in residential areas during July.

## Q5: Which zone had the highest count of property crime in residential areas during July?

The residential area of Zone5 have the **lowest crime counts** during the month of July each year, second of Zone7 (which is exclude due to its small and remote area).



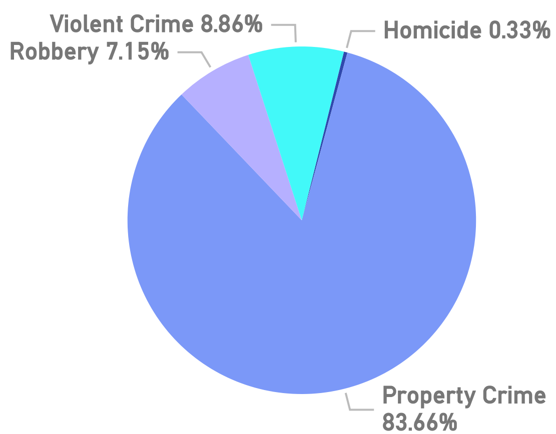



# Analysis of Atlanta Crime - Interactive

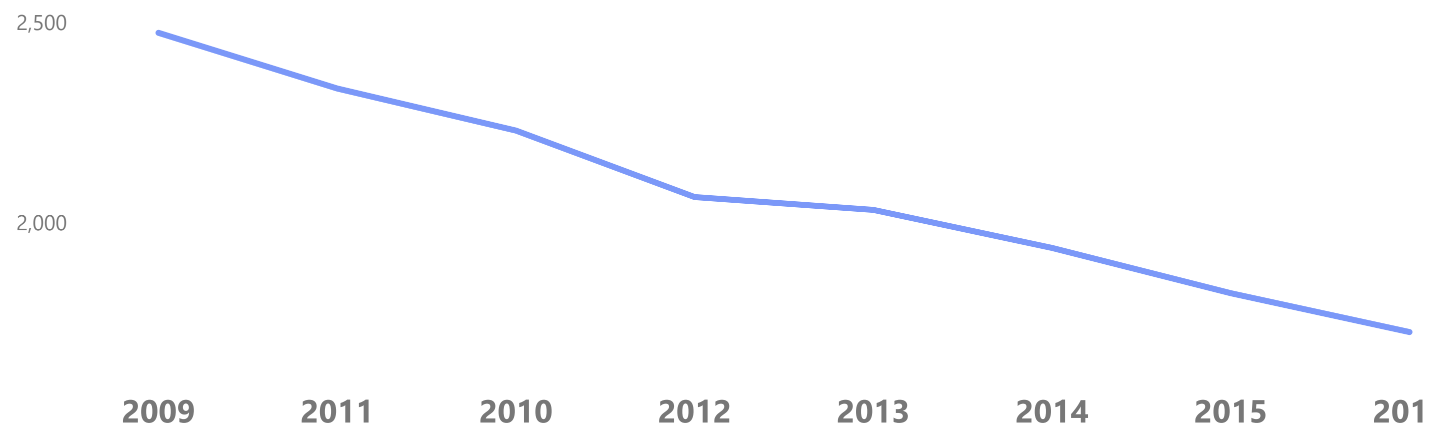
**16.63K**

Crime Count

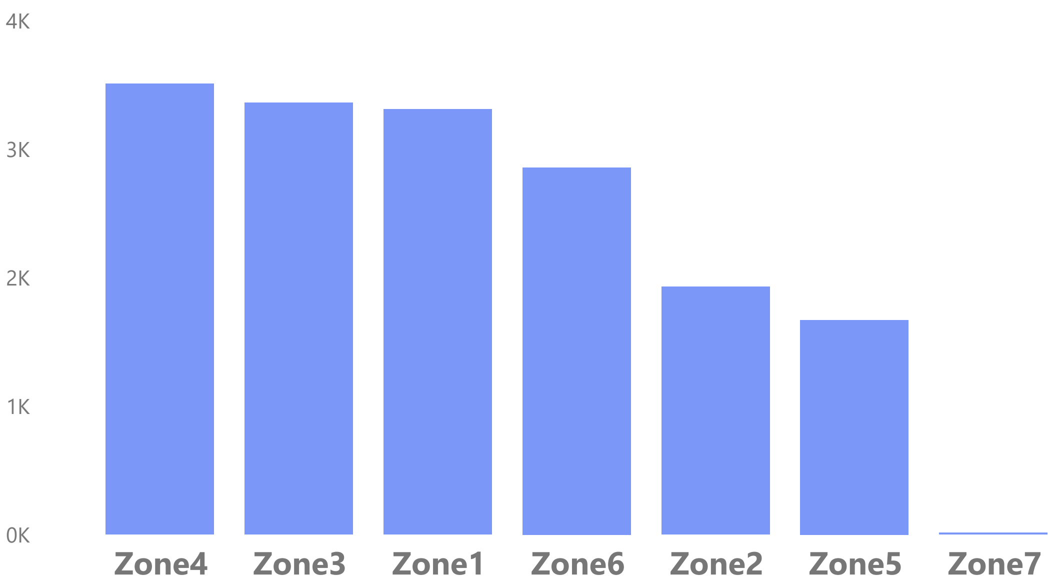
## Crime Count by Crime Type



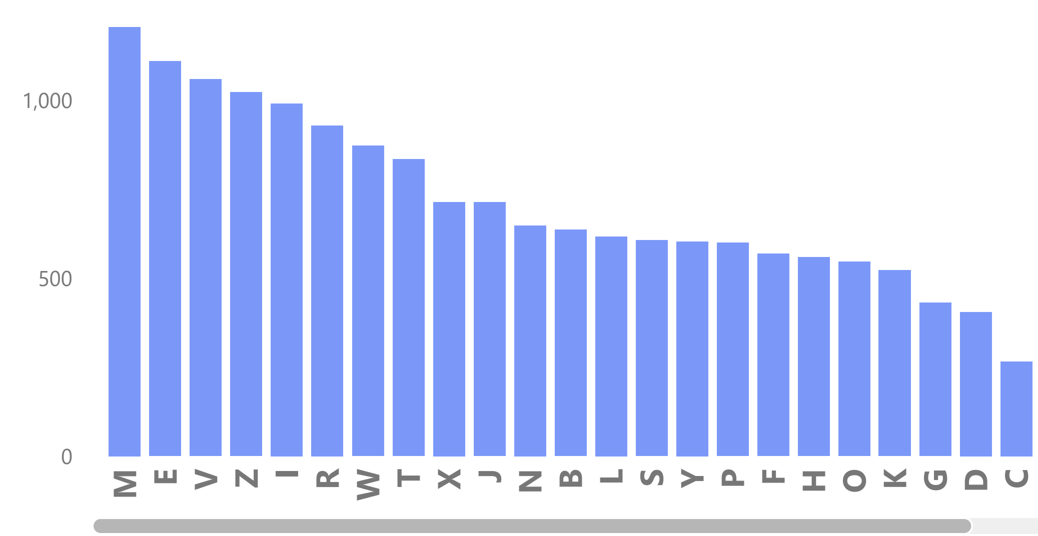

## Crime Count by Year




## Crime Count by Zone




## Crime Count by NPU




**Building Type**

- [ ] Commercial/Service
- [ ] Entertainment
- [ ] Medical
- [ ] Office
- [ ] Other
- [x] Residential
- [ ] Tourism
- [ ] Transportation
- [ ] Urban/Rural

**Month**

- [ ] April
- [ ] August
- [ ] December
- [ ] February
- [ ] January
- [x] July
- [ ] June
- [ ] March
- [ ] May
- [ ] November
- [ ] October

**Crime Type**

- [ ] Homicide
- [ ] Property Crime
- [ ] Robbery
- [ ] Violent Crime

# *Reference*


- *City of Atlanta. (n.d.). NPU & Neighborhood Relationship. Retrieved from https://www.atlantaga.gov/government/departments/city-planning/office-of-zoning-development/neighborhood-planning-unit-(npu)*
- *City of Atlanta Police Department. (n.d.). Atlanta Police Department Zone and Beat Maps. Retrieved from https://www.atlantapd.org/i-want-to/crime-data-downloads/zone-and-beat-maps*
- *Federal Bureau of Investigation (n.d.). Uniform Crime Reporting (UCR) Program. U.S. Department of Justice. Retrieved from https://www.fbi.gov/services/cjis/ucr*
- *Little, R.J.A., & Rubin, D.B. (2002). Statistical Analysis with Missing Data. John Wiley & Sons.*
- *Kimball, R., & Ross, M. (2013). The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling (3rd ed.). Wiley.*
- *OpenAI. (2021). ChatGPT: Conversational AI language model [Computer software]. Retrieved from https://www.openai.com/chatgpt*