HATE CRIME IN NYC

Jinming Chen, Benjamin Freund, Ziyang Guo, Randy Leon, Alejandra Zapata

Agenda

- Research Questions
- Data Profile
- Conceptual Architecture
- Dimensional Modeling
- Visualization
- Project Milestones & Timeline
- Team Responsibilities
- Challenges
- Lessons Learned

Research Questions

- Can we see differences from what is printed in news headlines to actual police hate crime data?

- Are there any insights we may gather from looking at where and when hate crime occurs?

Data Profile: Crime Dataset

Source of Information:

https://data.cityofnewyork.us/Public-Safety/NYPD-Hate-Crimes/bqiq-cu78

Number of Records: 832

Frequency of Updates: Quarterly

Data Type and Structure: Structured Dataset

Number of Columns: 15

Data Profile: News Dataset

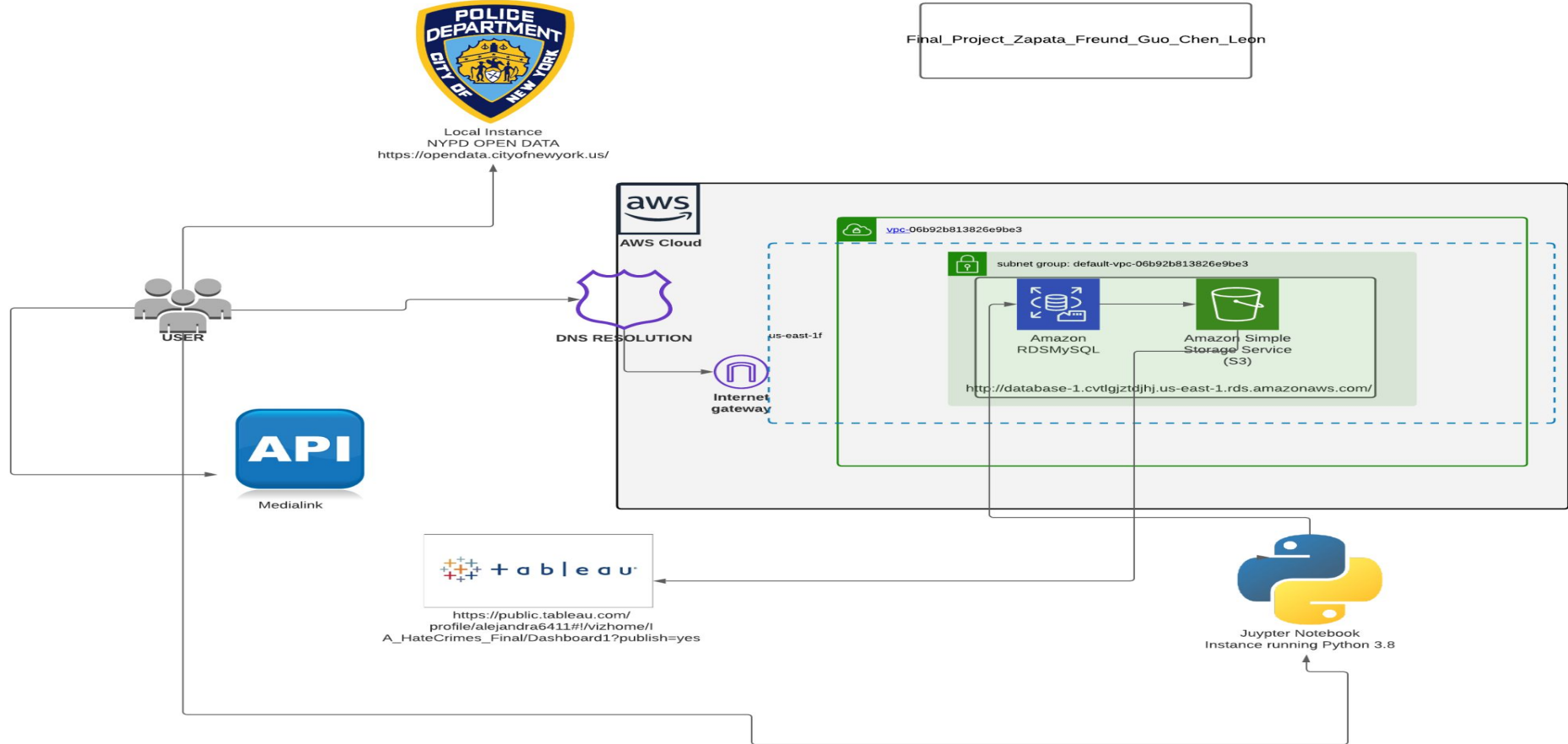Source of Information: mediastack API

Number of Records: 320
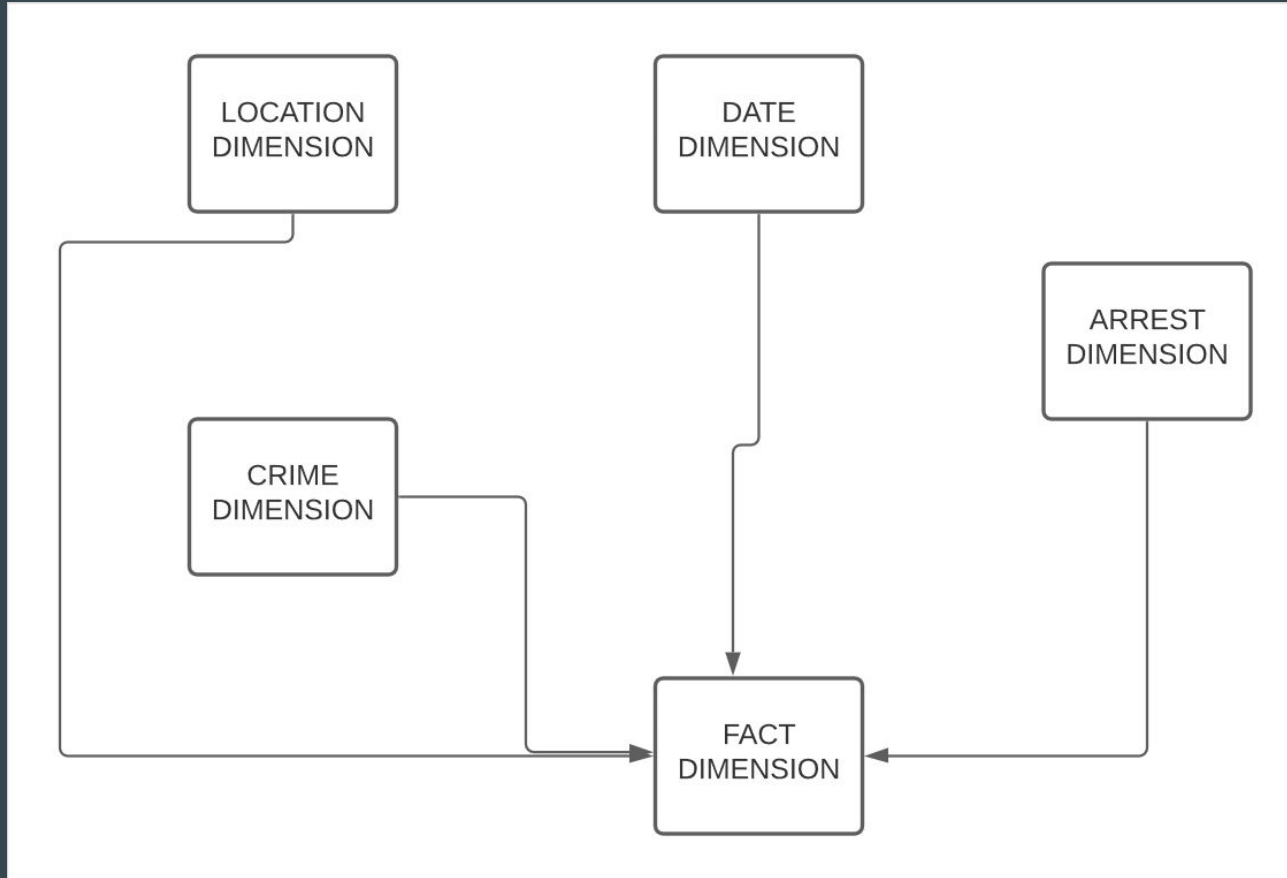
Frequency of Updates: Quarterly

Data Type and Structure: Unstructured Dataset - API
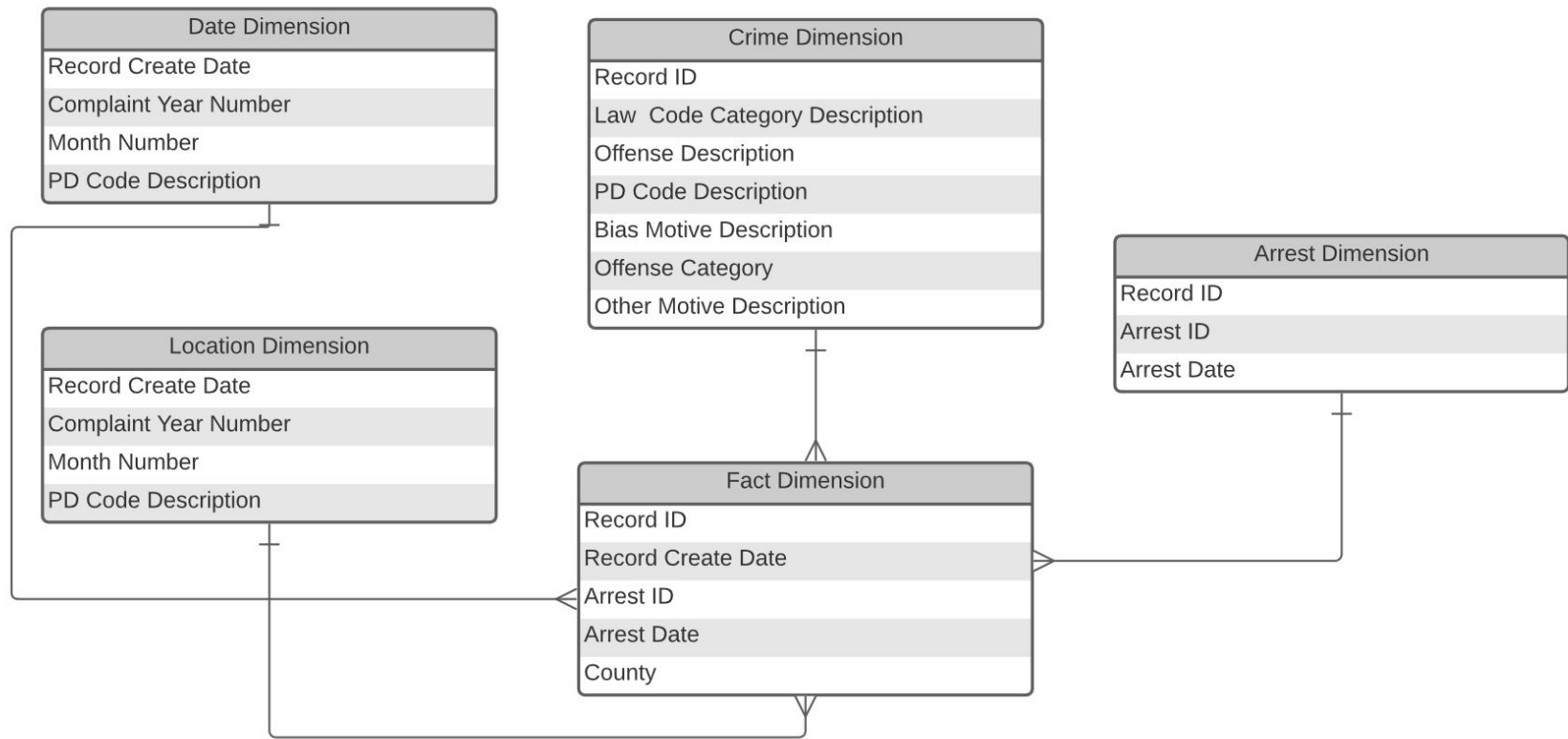
Number of Columns: 4
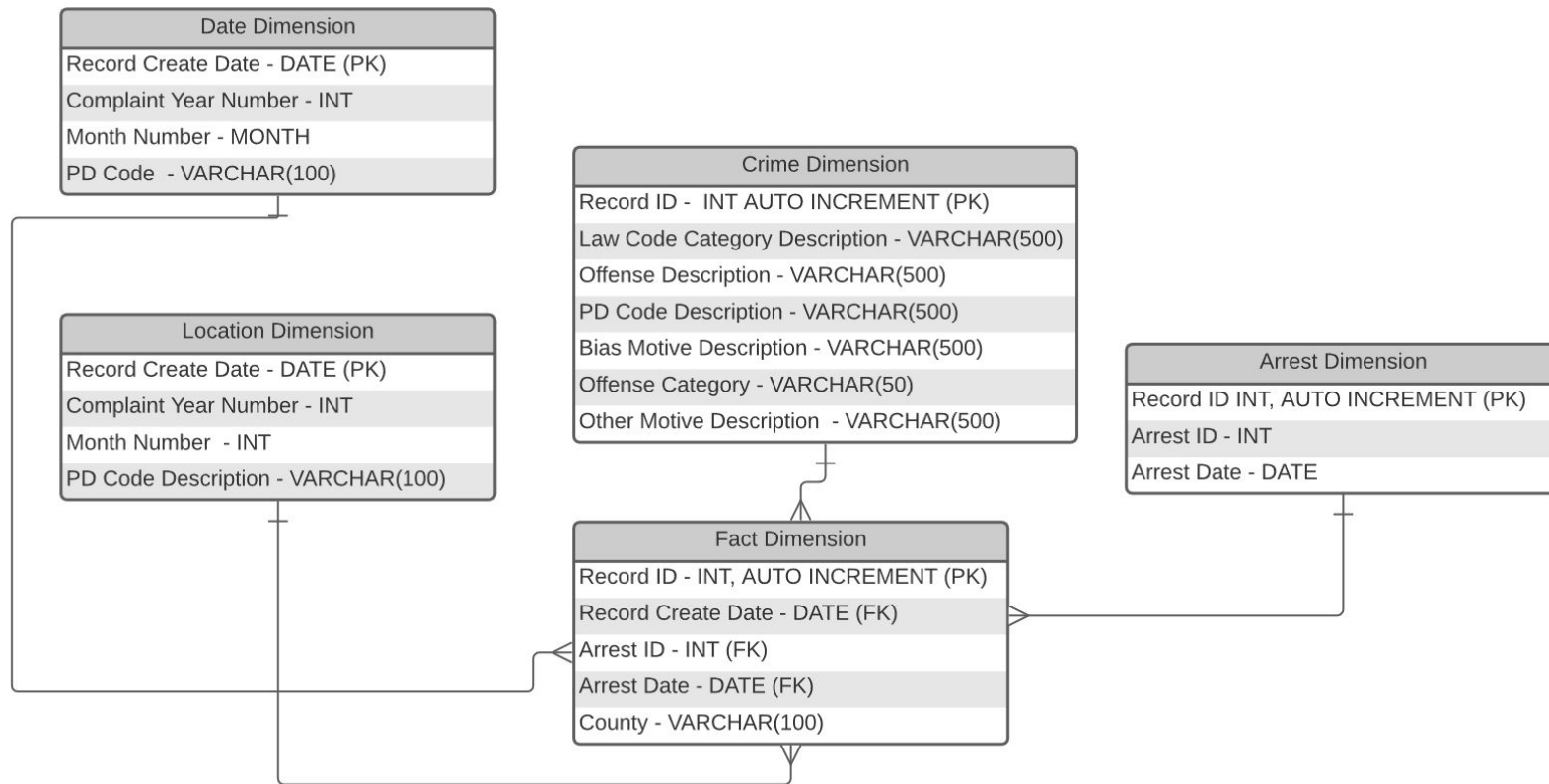
Architectural Diagram

Final_Project_Zapata_Freund_Guo_Chen_Leon

Local Instance
NYPD OPEN DATA
https://opendata.cityofnewyork.us/

aws
AWS Cloud

vpc-06b92b813826e9be3

subnet group: default-vpc-06b92b813826e9be3

us-east-1f

DNS RESOLUTION

Internet gateway

Amazon RDSMySQL

Amazon Simple Storage Service (S3)

http://database-1.cvtlgjztdjhj.us-east-1.rds.amazonaws.com/

USER

API
Medialink

tableau
https://public.tableau.com/
profile/alejandra6411#!/vizhome/I
A_HateCrimes_Final/Dashboard1?publish=yes

Juypter Notebook
Instance running Python 3.8

Conceptual Architecture

Conceptual Architecture ii.

Date Dimension
- Record Create Date
- Complaint Year Number
- Month Number
- PD Code Description

Crime Dimension
- Record ID
- Law Code Category Description
- Offense Description
- PD Code Description
- Bias Motive Description
- Offense Category
- Other Motive Description

Arrest Dimension
- Record ID
- Arrest ID
- Arrest Date

Location Dimension
- Record Create Date
- Complaint Year Number
- Month Number
- PD Code Description

Fact Dimension
- Record ID
- Record Create Date
- Arrest ID
- Arrest Date
- County

Conceptual Architecture iii.

Date Dimension
- Record Create Date - DATE (PK)
- Complaint Year Number - INT
- Month Number - MONTH
- PD Code - VARCHAR(100)

Location Dimension
- Record Create Date - DATE (PK)
- Complaint Year Number - INT
- Month Number - INT
- PD Code Description - VARCHAR(100)

Crime Dimension
- Record ID - INT AUTO INCREMENT (PK)
- Law Code Category Description - VARCHAR(500)
- Offense Description - VARCHAR(500)
- PD Code Description - VARCHAR(500)
- Bias Motive Description - VARCHAR(500)
- Offense Category - VARCHAR(50)
- Other Motive Description - VARCHAR(500)

Arrest Dimension
- Record ID INT, AUTO INCREMENT (PK)
- Arrest ID - INT
- Arrest Date - DATE

Fact Dimension
- Record ID - INT, AUTO INCREMENT (PK)
- Record Create Date - DATE (FK)
- Arrest ID - INT (FK)
- Arrest Date - DATE (FK)
- County - VARCHAR(100)

Development of the warehouse solution

First we created a github repository for our project. Link

Dimensional model

Python codes

Dimensional model

```sql
1   use master_dw;
2
3   drop table if exists `Arrest_Dim`;
4   create table `Arrest_Dim`(
5     `Record ID`  int primary key,
6     `Arrest ID`  varchar(50),
7     `Arrest_Date`  varchar(50)
8   );
9
10  drop table if exists `Crime_Dim`;
11  create table `Crime_Dim`(
12    `Record ID`  int primary key,
13    `Law Code Category Description`  varchar(30),
14    `Offense Description`  varchar(30),
15    `PD Code Description`  varchar(80),
16    `Bias Motive Description`  varchar(30),
17    `Offense Category`  varchar(60),
18    `Other Motive Description`  varchar(30)
19  );
20
21  drop table if exists `Date_Dim`;
22  create table `Date_Dim`(
23    `Record Create Date`  varchar(60),
24    `Complaint Year Number`  int,
25    `Month Number`  int,
26    `Arrest Date`  varchar(50),
27    key `fk_01`(`Arrest Date`),
28    key `fk_02`(`Record Create Date`)
29  );
```

```sql
31  drop table if exists `Location_Dim`;
32  create table `Location_Dim`(
33    `Patrol Borough Name` varchar(50),
34    `County` varchar(25)
35  );
36
37  drop table if exists `fact_Dim`;
38  create table `fact_Dim`(
39    `Record ID` int auto_increment,
40    `Record Create Date` varchar(60),
41    `Arrest ID` varchar(30),
42    `Arrest Date` varchar(50),
43    `County` varchar(25),
44    primary key(`Record ID`),
45    unique key(`Record Create Date`, `Arrest Date`,`County`),
46    constraint `fk_Arrest_Dim_fact` foreign key (`Record ID`) references `Arrest_Dim` (`Record ID`),
47    constraint `fk_Crime_Dim_fact` foreign key (`Record ID`) references `Crime_Dim` (`Record ID`),
48    constraint `fk_Date_Dim_fact_01` foreign key (`Record Create Date`) references `Date_Dim` (`Record Create Date`),
49    constraint `fk_Date_Dim_fact_02` foreign key (`Arrest Date`) references `Date_Dim` (`Arrest Date`)
50  )
51
```

Dimensional model

```
1   use master_dw;
2   insert into `Arrest_Dim`(
3     `Record ID`,
4     `Arrest ID`,
5     `Arrest_Date`)(
6   select `Record ID`,`Arrest Id`, `Arrest Date`
7   from master.NYPD
8   );
9
10  insert into `Crime_Dim`(
11    `Record ID`,
12    `Law Code Category Description`,
13    `Offense Description`,
14    `PD Code Description`,
15    `Bias Motive Description`,
16    `Offense Category`,
17    `Other Motive Description`
18    )(
19  select
20    `Record ID`,
21    `Law Code Category Description`,
22    `Offense Description`,
23    `PD Code Description`,
24    `Bias Motive Description`,
25    `Offense Category`,
26    `Other Motive Description`
27  from master.NYPD
28  );
```

```
43  insert into `Location_Dim`(
44    `Patrol Borough Name`,
45    `County`
46    )(
47  select `Patrol Borough Name`,
48    `County`
49  from master.NYPD
50  );
51
52  insert into `fact_Dim`(
53    `Record ID`,
54    `Record Create Date`,
55    `Arrest ID`,
56    `Arrest Date`,
57    `County`
58    )(
59  select
60    `Record ID`,
61    `Record Create Date`,
62    `Arrest ID`,
63    `Arrest Date`,
64    `County`
65  from master.NYPD
66  );
```

```
67  select * from fact_Dim;
```

Result Grid | Filter Rows: | Edit: | Export/I

Record ID	Record Create Date	Arrest ID	Arrest Date	County
1	2/5/2019	B19606200	2/8/2019	BRONX
2	3/9/2019	B19610772	3/9/2019	BRONX
3	3/8/2019	B19610788	3/10/2019	BRONX
4	3/10/2019	B19610788	3/12/2019	BRONX
5	5/9/2019	B19620156	5/10/2019	BRONX
6	5/10/2019	B19620157	5/11/2019	BRONX
7	5/11/2019	B19620158	5/12/2019	BRONX
8	6/1/2019	B19625012	6/14/2019	BRONX
9	7/9/2019	B19630132	7/21/2019	BRONX
10	8/5/2019	B19632323	8/5/2019	BRONX
11	10/22/2019	B19642917	10/22/2019	BRONX
12	10/23/2019	B19642921	10/24/2019	BRONX
13	12/28/2019	B19650852	12/28/2019	KINGS
14	2/6/2020	B20604576	2/6/2020	BRONX
15	3/30/2020	B20610471	3/28/2020	BRONX
16	3/27/2020	B20610472	3/25/2020	BRONX
17	3/28/2020	B20610474	3/26/2020	BRONX
18	4/12/2020	B20612516	4/29/2020	BRONX

Demo of Tableau Dashboard

https://public.tableau.com/profile/alejandra6411#!/vizhome/IA_HateCrimes_Final/Dashboard1?publish=yes

Project Milestones / Timeline

Jinming Chen, Benjamin Freund, Ziyang Guo, Randy Leon, Alejandra Zapata

	Feb 21 - Feb 27	Feb 28 - Mar 6	Mar 7 - Mar 13	Mar 14 - Mar 20	Mar 21 - Mar 27	Mar 28 - Apr 3	Apr 4 - Apr 10	Apr 11 - Apr 17	Apr 18 - Apr 24	Apr 25 - May 1	May 2 - May 8	May 9 - May 15
Bus Matrices	█	█										
Basic Use Cases	█	█										
Dimensional Model with ETL Instructions		█	█	█	█							
Conceptual Model		█	█	█								
Logical Model		█	█	█								
Physical Model		█	█	█								
Data Dictionaries			█	█								
Architecture Diagram with Network Information				█	█	█						
Waterfall Plan	█	█	█	█	█	█	█	█	█	█	█	█
ETL/ELT Code							█	█	█	█	█	
Sample Queries							█	█	█	█	█	
Visualization										█	█	█
Presentation										█	█	█

Team Responsibilities

Jinming Chen - Data dictionary, ETL code

Benjamin Freund - Bus matrices, basic use cases, waterfall project plan

Ziyang Guo - Dimensional model matrix, ETL code

Randy Leon - Conceptual, logical, and physical models, architecture diagram

Alejandra Zapata - Visualization

Assumptions

1. Not all crime is reported.

2. The body of a news article is more descriptive than the headline.

3. NYPD officers are more interested in stopping violent crime than non-violent crime.

4. Our data warehouse would primarily be used by NY government and police officials, not by the general public.

5. A more extensive data warehouse can be built given more time and resources.

6. Knowing the information presented in our data warehouse could help NY take definitive action against violent hate crimes.

Challenges

1. Still so many tools and knowledge we have not practice in the class.

2. So many interesting research cannot be done due to the dataset/money barrier.

3. Different skill levels in each group member.

4. Inability to meet in person to sit down and work through problems.

5. Scheduling around time zones, work schedules, religious observances, etc.

Lessons Learned

1. It's important to have a developing understanding of AWS architecture when trying to build an accessible data warehouse.

2. Planning, delegating, and cooperating are crucial aspects of any team.

3. Flexibility is key.

4. Everything you create should be designed with the user in mind, not with an end goal that you decided was important.

5. Data warehouses can be complex, so make sure you schedule enough time to fully develop one.

6. "Teamwork makes the dream work!"

Thanks for listening!

Any questions?