

Project Design Report

[Atlanta PD Uniform Crime Reports]

[CrimeDB/Team Randomness]
[4/30/2015]

Michael Dang
Dean Brannon
Bradley Thomas

Introduction:

Our database model is a representation of the Atlanta Police Department's Uniform Crime Reports. To be more specific, our database revolves around the crime incidents that are reported to the Atlanta PD. This includes but is not limited to the location of the incident, the people involved, the type of offense, and the time/date that it occurred. We chose to use the Atlanta PD's UCR, because we wanted to shed more light on the crime activity in our city. By creating an expansive database, detailing as much information on a specific incident, we want to provide an application that can be used to filter and retrieve the history of specific crime incidents of the city of Atlanta, while only having to provide little background information. We also hope that our database can be utilized to provide safety, by providing a method to pinpoint areas of the city that tend to have higher crime activity, so that people can avoid any unwarranted trouble.

Requirements Analysis:

Data Requirements:

INCIDENT Entity

- 1) Each incident will have a unique ID, idIncident.(Primary Key)
- 2) Incident will have a Date and Time to track when the incident was reported
- 3) If applicable, an incident can have a PersonOrWitness.(PersonOrWitness_ID, is FK to PersonOrWitness Entity)
- 4) If applicable, an incident can have an Officer.(Officer_Badge, is FK to Officer Entity)
- 5) Each incident is classified by at least one UCR.(UCRID, is FK to UCR Entity)
- 6) Each incident is associated with a Location.(Location_Address is FK to Location Entity)

UCR Entity

- 1) Each UCR will have a unique UCRID, this value comes from a standardized UCR Code
- 2) Each UCR will have a UCRDESCRIPTION, which details the UCRID
- 3) Each UCR has a boolean value for isViolent.

OFFICER Entity

- 1) Each officer has a unique Badge#.(Primary Key)
- 2) Each officer has a FNAME and LNAME
- 3) Each officer has a Rank, a number indicating their Rank
- 4) Each officer is assigned to a Beat(s). (BEAT is FK to Beat Entity)

PERSONORWITNESS Entity

- 1) Each PersonOrWitness has a unique ID. (Primary Key)
- 2) A Victim has a FNAME and LNAME

BEAT Entity

- 1) Each Beat has a unique ID.(Primary Key)
- 2) Each Beat has a schedule
- 3) Each Beat has a time

NEIGHBORHOOD Entity

- 1) Each Neighborhood has a unique ID.(Primary Key)
- 2) Each neighborhood has a Name
- 3) Each Neighborhood is assigned a BEAT. (BEAT_ID is FK to Beat Entity)

LOCATION Entity

- 1) Each Location has a unique ADDRESS. (Primary Key)
- 2) Location has type(home, apartment, office, etc)
- 3) Each location is assigned to a Neighborhood. (NEIGHBORHOOD_ID is FK to Neighborhood Entity)

INVENTORY Entity

- 1) Each INVENTORY has a unique ID. (Primary Key)
- 2) Each INVENTORY has an ENAME(equipment name)
- 3) Each INVENTORY has an ECOUNT(equipment count)

ASSIGNEDINV Entity

- 1) AssignedInv has a unique ID. (Primary Key)
- 2) Is assigned to an Officer. (Officer_Badge is FK to Officer Entity)

DEPARTMENT Entity

- 1) Each Department has a unique DepNo. (Primary Key)
- 2) Each Department has a Supervisor which is an Officer. (Supervisor is FK to Officer Entity)

Functional Requirements:

- Functional Requirement 1: Database can return all incidents an officer worked on
- Functional Requirement 2: Database can return all Burglary incidents reported
- Functional Requirement 3: Database can return all incidents reported at an apartment
- Functional Requirement 4: Database can return all incidents reported in the year 2014
- Functional Requirement 5: Database can return all officers assigned to a Location
- Functional Requirement 6: Database can return retrieve all incidents reported before noon
- Functional Requirement 7: Database can return incidents involved a specific victim
- Functional Requirement 8: Database can return all incidents associated with a Person's name
- Functional Requirement 9: Database can return all violent incidents
- Functional Requirement 10: Database can return all non violent incidents

Conceptual Design:

Entities and Attributes:

Entity 1: Incident

Incident: Crime Incident that is reported to the Atlanta PD

Attributes

- INCIDENT_ID: unique ID for every incident reported
- DATE: the date that the incident was reported
- TIME: the time that the incident was reported
- ADDRESS: the address(house number and street) where the incident was reported
- UCRID: the UCR classification of the incident
- OFFICERBADGE: The Officer's, who worked the incident, badge number
- PersonOrWitness_ID: ID of the PersonOrWitness involved in the incident

Relationships

- has UCR: Uniform Crime Report classification code that details the type of incident that occurred
- has Officer: Officer who is assigned to the incident
- has PersonOrWitness: This is a person involved/witness to the incidents
- has Location: This is the address/location where the incident was reported

Primary Key:

- It is identified by idIncident, which is a uniquely assigned identifier for each incident reported

Entity 2: UCR

UCR: Uniform Crime Report classification of an Incident

Attributes

- idUCR: unique UCR Code, based on type of UCR
- UCRDescription: Detailed description of the UCR type
- IsViolent: boolean value based on if the UCR type is violent or non violent

Entity 3: PersonOrWitness

PersonOrWitness: A Person/Witness who is in some way or form involved with an incident.

Attributes:

- Victim_ID: Unique identifier for PersonOrWitness
- FNAME: First Name of Person
- LNAME: Last Name of Person

Entity 4: Officer

Officer: An Officer of the Atlanta Police Department, assigned to incidents

Attributes:

- Badge#: unique identifier for the Officers
- rank: Officer's stature within the Police Department. Given by int value
- FNAME: First name of Officer
- LNAME: Last name of Officer
- BEAT: This is the Beat ID that they patrol

Relationships:

- works Beat: This is a patrol route that the Officer works

Primary Key:

- Officer is identified by his Badge#

Entity 5: Neighborhood

Neighborhood: A neighborhood within the jurisdiction of the Atlanta PD

Attributes:

- NeighborhoodID: Unique ID per Neighborhood
- Nname: Neighborhood name
- BEATID: ID of the Beat that patrols the neighborhood

Relationships:

- has Beat: This is the Beat that patrols the Neighborhood

Entity 6: Location

Location: A location(physical address) within the jurisdiction of the Atlanta PD

Attributes:

- Address: House# and Street Name of the Location
- type: This is the type of location(house, apartment, office, etc)
- LATITUDE: latitude
- LONGITUDE: longitude
- NeighborhoodID: This is the Neighborhood's ID that reside within a location

Relationships:

- has Neighborhood: Neighborhood where the location resides

Primary Key:

- Location is identified by the Address

Entity 7: Beat

Beat: Patrol route of an Officer

Attributes:

- BeatID: unique identifier for the Beat
- Schedule: The different shifts of the beat
- Time: Date/time of the Beat

Primary Key:

- It is identified by the BeatID, which is a unique identifier assigned to each beat(patrol route)

Entity 8: AssignedInv

AssignedInv: This is the equipment of an Officer

Attributes:

- InventoryID: Inventory item
- OFFICERBADGE: Officer

Relationships:

- has Inventory: The inventory to be assigned
- has Officer: The officer being assigned inventory

Primary Key:

- AssignedInv is identified by both the InventoryID and the OFFICERBADGE

Entity 9: Inventory

Inventory: This is the equipment officers can have

Attributes:

- InventoryID: Unique identifier for the Inventory item
- ENAME: Equipment name
- ECOUNT: Quantity of the equipment

Primary Key:

- The primary key for inventory is the InventoryID

Entity 10: Department

Department: This is a department of the Atlanta PD that houses Officers

Attributes:

- DeptNo: This is a unique identifier of the Department
- Supervisor: This is an Officer's badgenumber who is in charge at the Department's
- DNAME: Department Name
- Address: Address(house num and street name)

Relationship:

- hasOfficer: This is the supervisor of the department

Primary Key:

- DeptNo. Is the unique identifier for the Department

Relationships

Relationship 1: Incident has PersonOrWitness

Relation: shows the PersonOrWitness associated with an Incident

Cardinalities:

- Incident to *PersonOrWitness*, 0 to N, Incident does not have to have a Person or witness but can have many
- *PersonOrWitness* to Incident, 1 to N, Person must be associated to at least one incident to be in the database, but can be associated to many Incidents

Relationship 2: Incident has Location

Relation: shows the relation between an Incident and the location that it was reported at

Cardinalities:

- Incident to Location, 1 to 1, Every incident has to have a location of occurrence
- Location to Incident, 0 to N, A location might not have had any incidents occur

Relationship 3: Incident has UCR

Relation: shows the relation between an Incident and the UCR classification of the incident

Cardinalities:

- Incident to UCR Classification, 1 to N, because an incident can be classified under multiple categories based on the events that took place
- UCR Classification to Incident, 0 to N, because there may not have been an incident reported for a classification(possibly something very rare and severe), but many of the same classification of incidents can occur

Relationship 4: Incident handled by Officer

Relation: shows the relation between an Incident and the Officer that handles the case

Cardinalities:

- Incident to Officer, 0 to N, an incident does not have to be assigned to an officer right away, because the Atlanta PD only has so much manpower, but some incidents require multiple officers
- Officer to Incident, 1 to N, every officer has to be working on at least one incident, but due to the lack of manpower they can be assigned to multiple incidents

Relationship 5: Location in Neighborhood

Relation: shows the relation between a Location and the neighborhood in which the location resides in

Cardinalities:

- Location to Neighborhood, 1 to 1, A location can only be in one neighborhood
- Neighborhood to Location, 1 to N, Neighborhood has many locations

Relationship 6: Beat has Neighborhoods

Relation: shows the relation between neighborhoods and the beat that patrols the neighborhoods

- Neighborhood to Beat, 1 to 1, A neighborhood is covered by one beat
- Beat to Neighborhood, 1 to N, A Beat covers many neighborhoods

Relationship 8: Officer has AssignedInv

Relation: shows the relation between an Officer and his AssignedInv

- Officer to AssignedInv, 0 to N, Officer might have a lot of inventory, or none
- AssignedInv to Officer, 1 to 1, AssignedInv must belong to an Officer to be created in the database

Relationship 9: AssignedInv has Inventory

Relation: shows the relation between a AssignedInv and Inventory

- AssignedInv to Inventory, 1 to 1, AssignedInv has one Inventory item to be assigned
- Inventory to AssignedInv, 0 to N, Inventory might not be assigned, but can be assigned to multiple people based on the type of inventory

Relationship 10: Department has Officer

Relation: shows the relation between a Department and the Officer in charge of the Department

Cardinalities:

- Department to Officer, 1 to 1, Department has 1 Officer in charge
PD can not afford to patrol the same location with multiple units
- Officer to Department, 0 to 1, An officer is either in charge of one department or not in charge at all

Relational Model:

The design of our EER diagram left not much work in the way of converting to a relational model. Originally there were no composite keys in the final EER diagram so 1NF is achieved. None of our relations contain any composite keys, and since are in 1NF are therefore also in 2NF. All non-key attributes are fully dependent on the primary keys, and therefore the database is in 3NF.

INVENTORY	<u>INV_ID</u>	ENAME	ECOUNT
-----------	---------------	-------	--------

The inventory relation models the inventory entity. The inventory is checked out by an officer, and these transactions are related in a table named assigned inventory.

ASSGNDINV	BADGENO	INVNUM
-----------	---------	--------

This relation stores the relationship between an officer and the inventory. Badgeno is a foreign key referring to the officer's badge (primary key) and invnum refers to the inventory entity primary key inv id. This relation models the relationship CHECKS OUT in the EER diagram between officer and inventory.

DEPARTMENT	<u>DEPT_NO</u>	SUPERVISOR	DNAME	ADDRESS
------------	----------------	------------	-------	---------

Department supervisor is a foreign key referring to an officer's badge number.

UCR	<u>UCRID</u>	UCRDESCRIPTN	IS VIOLENT
-----	--------------	--------------	------------

This relation is modeled directly from the UCR entity in the EER diagram.

OFFICER	<u>BADGE#</u>	FNAME	LNAME	RANK	BEAT
---------	---------------	-------	-------	------	------

This relation is modeled directly from the OFFICER entity in the EER diagram.
Beat is a foreign key that refers to beat id in the Beat entity.

VICTIM/WITNESS	<u>VICT_ID</u>	FNAME	LNAME
-----------------------	----------------	-------	-------

This relation is modeled directly from the VICTIM entity in the EER diagram.

INCIDENT	<u>INCIDENT_ID</u>	DATE	TIME	OFFICERBADGE	VICTIMID	BEATID	UCRID	ADDRESS
-----------------	--------------------	------	------	--------------	----------	--------	-------	---------

This relation is modeled directly from the INCIDENT entity in the EER diagram.
Officerbadge is a foreign key to the OFFICER entity primary key, victim id is a foreign key referring to

BEAT	<u>BEAT_ID</u>	OFFICERBADGE
-------------	----------------	--------------

This relation is modeled directly from the BEAT entity in the EER diagram.

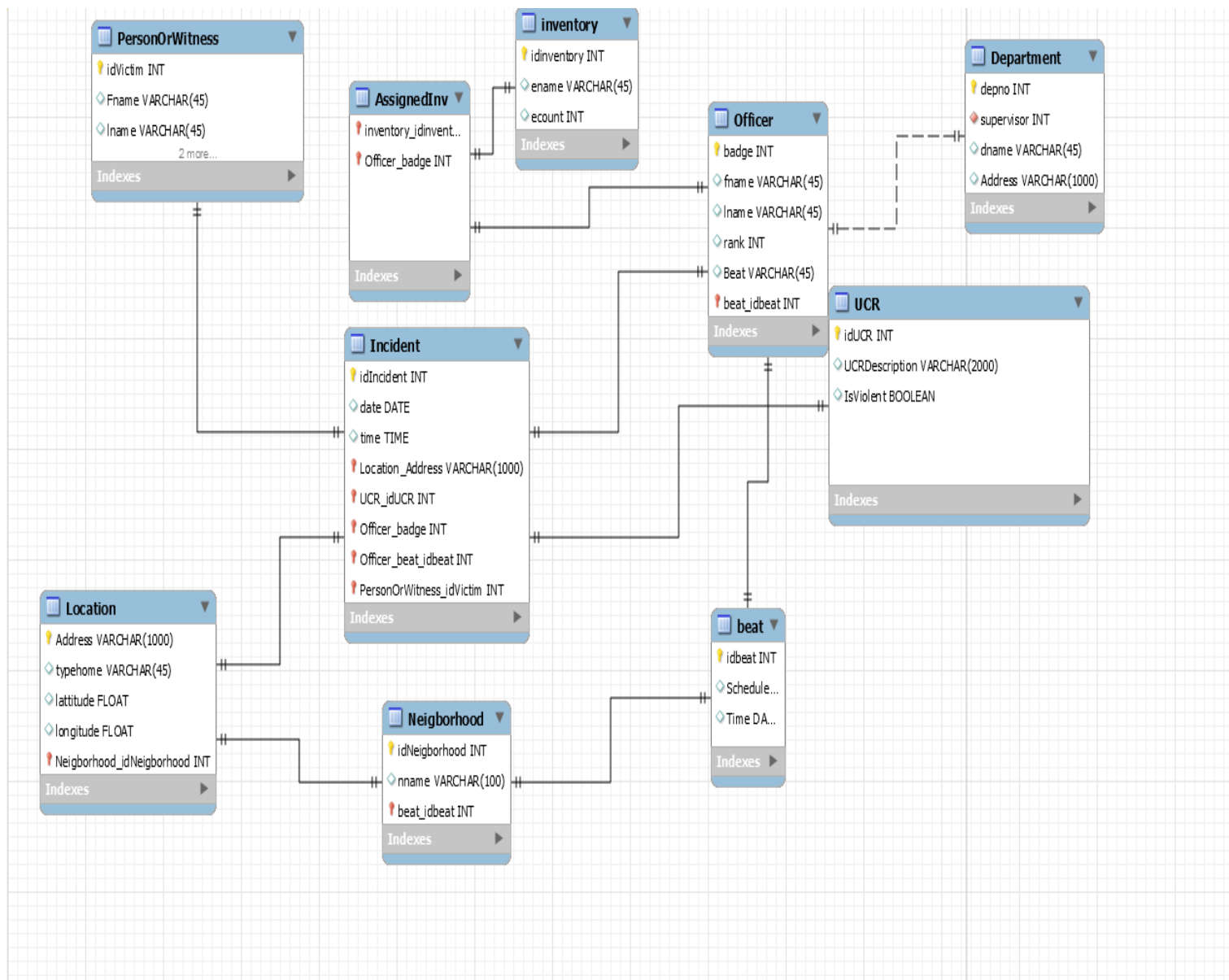
LOCATION	<u>ADDRESS</u>	TYPE	LATITUDE	LONGITUDE	NEIGHBRHOODID
-----------------	----------------	------	----------	-----------	---------------

This relation is modeled directly from the LOCATION entity in the EER diagram.

NEIGHBRHD	NEIGHBORHOODID	NNAME	BEATID
-----------	----------------	-------	--------

This relation is modeled directly from the NEIGHBORHOOD entity in the EER diagram.
Beat id is a foreign key referring to the beat entity primary key.

Data Dictionary:



Teamwork:

Introduction [Michael Dang]

Data Requirements [Michael Dang, Dean Brannon, Bradley Thomas]

Functional Requirements [Michael Dang, Dean Brannon]

Conceptual Design [Michael Dang, Dean Brannon, Bradley Thomas]

ER Design [Michael Dang, Dean Brannon, Bradley Thomas]

Relational Model [Dean Brannon]

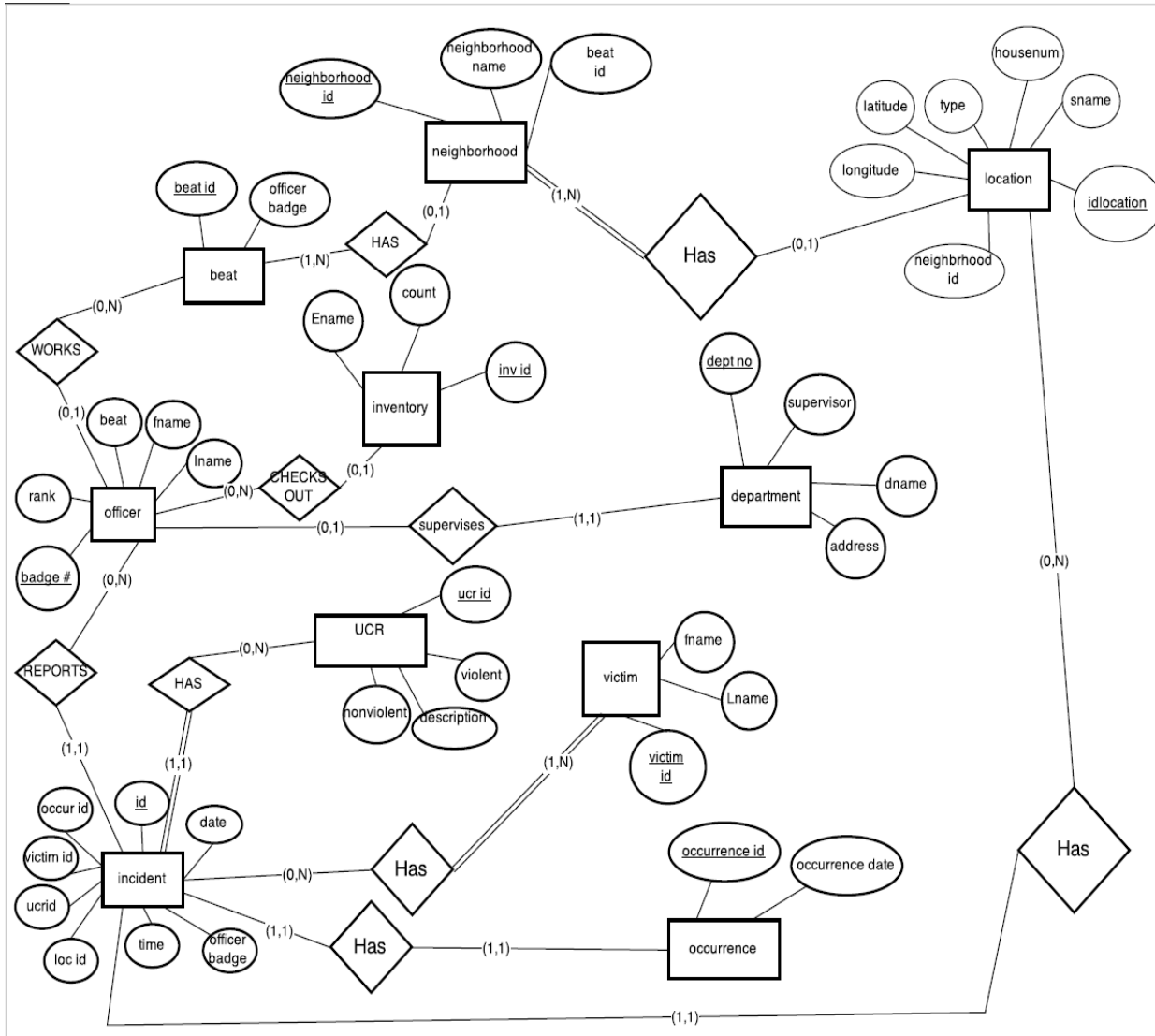
Data Dictionary [Bradley Thomas]

Database Implementation [Bradley Thomas]

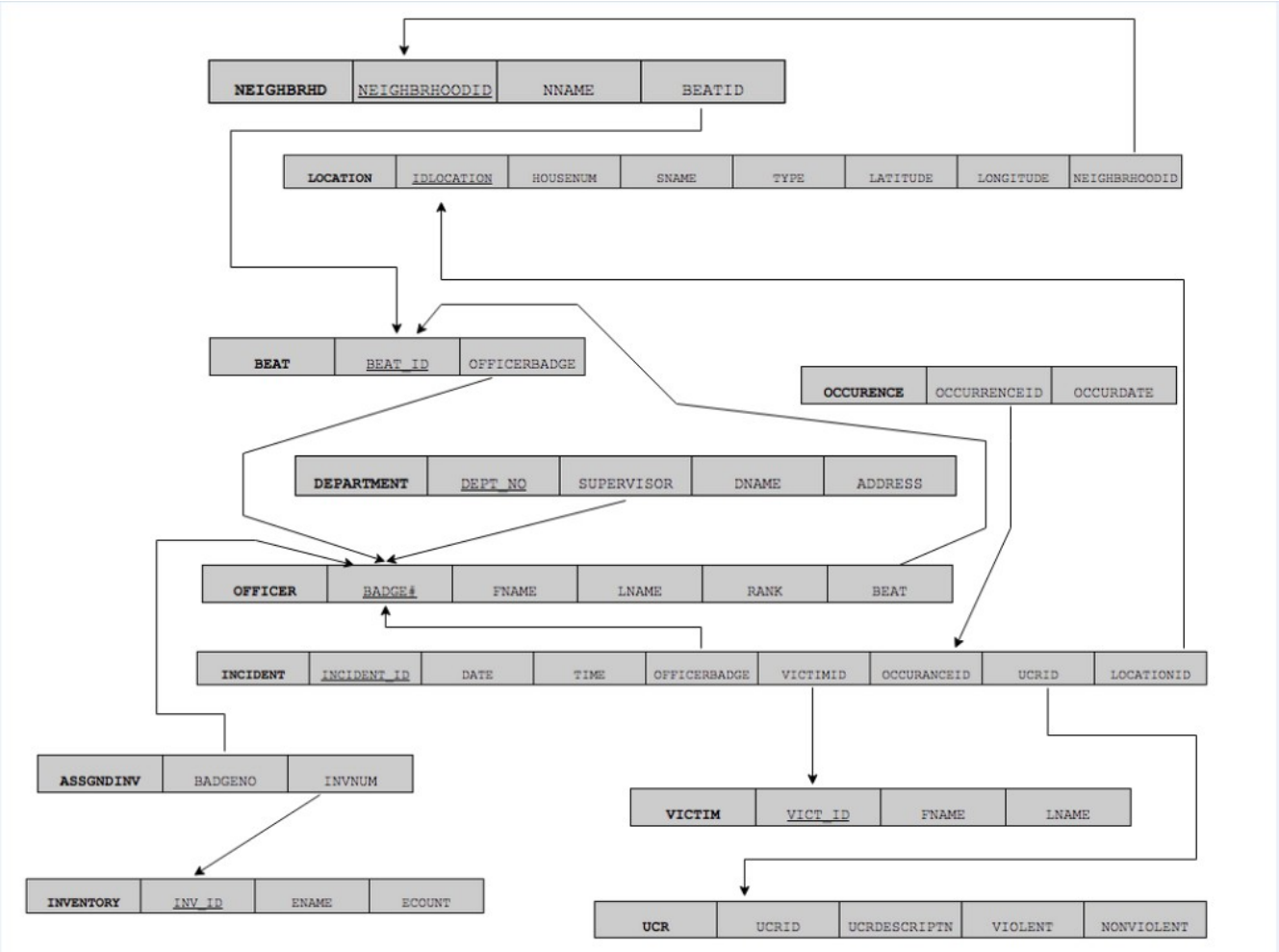
Appendix [Michael Dang, Dean Brannon]

Appendix:

EER Diagram



Relational Model



Change Log:

A lot of entities were removed or condensed into another entity for provide a better design.

Removed

- UCR Sub-classification – moved imp fields into UCRDESCRIPTION
- Offense and Non-Offense – Changed to a boolean field in UCR
- Shift – Redundant
- Person - Redundant

Condensed

- X and Y Coordinates – moved this to latitude and longitude in Location entity
- Occurrence Time and Date – moved these to a Time and Date field in Incident entity
- Suspect and Victim – Combined the two entities into a single entity

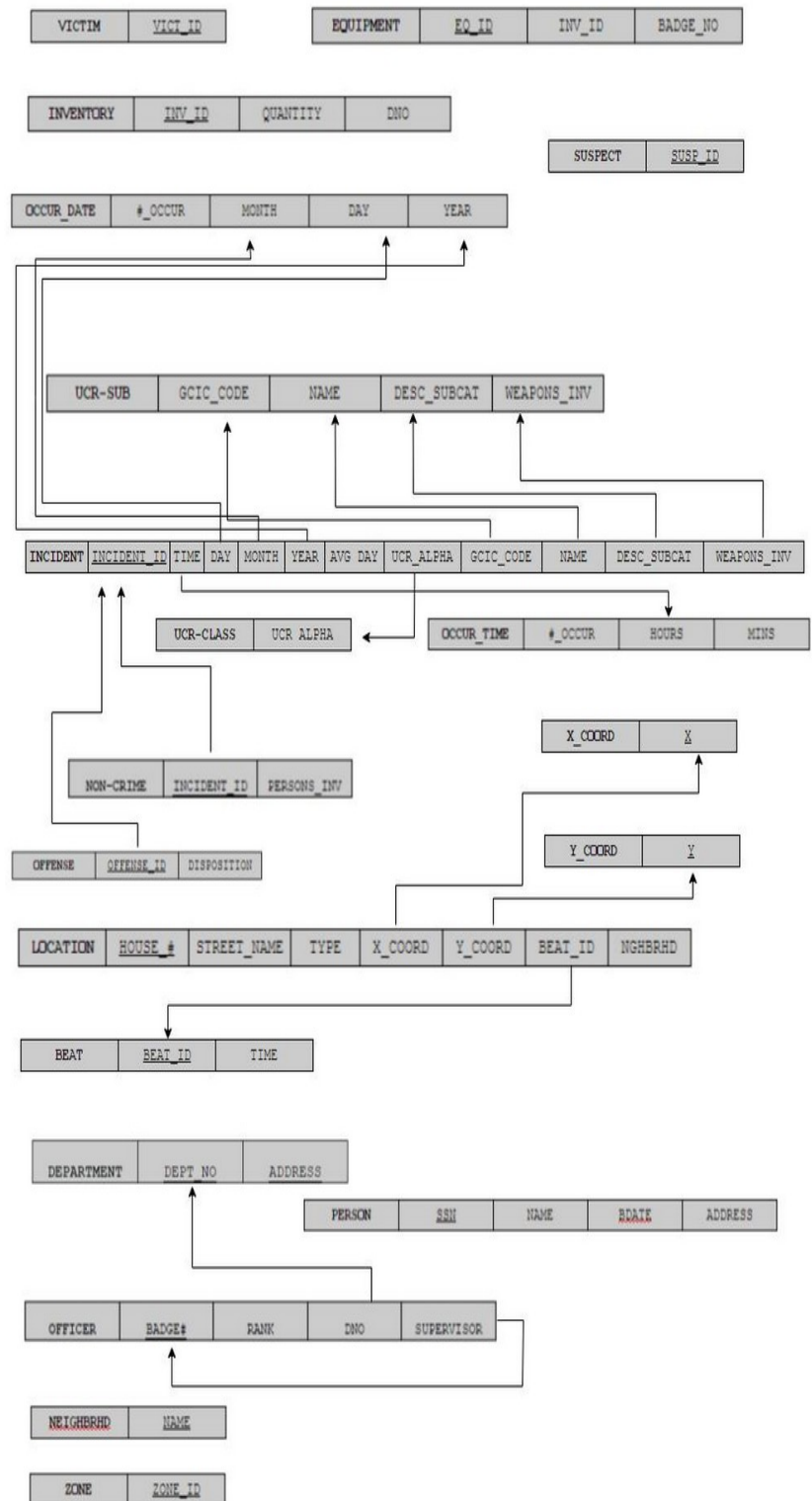
Added

- Department
- Inventory
- AssignedInv

Original Relational Model:

MAPPING THE OVERLY-
COMPLEX ER DIAGRAM
TO RELATIONAL MODEL:
BIG HEADACHE

- REDUNDANCIES
- CONFUSION



Original EER Diagram

OUR ORIGINAL
MODEL:

- LARGE
- MANY RELATIONSHIPS

