CIS 3400 Group Project

MTA Train Scheduling

Crystal Leon: crystal.leon@baruchmail.cuny.edu
Tata Saoidoh: tata.dagnogo@baruchmail.cuny.edu
Yongqiang Zhou:yongqiang.zhou@baruchmail.cuny.edu
Jinwoo Rhee: jinwoo.rhee@baruchmail.cuny.edu
Mohammed Chenafi :mohammed.chenafi@baruchmail.cuny.edu

Page1: Business Scenario

Group 9: MTA Business Scenario | Dataset (Click to see the dataset)

Problem: Excessive crowding on certain train lines, particularly during peak hours.

Solution: Enhance services for MTA users by providing accurate train schedules and regulating the number of passengers allowed into the subway.

In response to the significant overcrowding on Manhattan's MTA train lines, especially during peak hours, a comprehensive solution is being developed to enhance the commuting experience. The primary focus is to address the congestion on popular lines such as M, W, Q, F, and especially the 1, 2, 3, N, Q, and R trains. To tackle this, accurate and transparent train schedules will be provided to the public, allowing for better trip planning and time management. Furthermore, the number of passengers allowed entry into the subway will be controlled to prevent excessive crowding.

In addition, the train schedules will be carefully analyzed and additional trains will be scheduled as needed to distribute the passenger load more evenly across the network. This measure will be supplemented by optimizing routes to ensure minimal train traffic delays and to manage peak hour travel more effectively. The synchronization of employee and conductor schedules with these peak times is crucial to maintaining a smooth operation.

The service improvement plan also includes a mix of more local and express trains, catering to the various needs of the commuters, whether they seek faster travel or more frequent stops. By enhancing the reliability of the trains and ensuring that they run on time, the MTA aims to reduce the waiting periods for passengers, thus minimizing the crowds that typically gather during delays. The overarching aim is to maximize service with minimal train traffic delays, creating a more reliable, efficient, and comfortable experience for all MTA train users.

Page 2: E-R diagram

MTA PROJECT DATABASE

ER DIAGRAM

TABLES

Trains
TrainID (PK)
TrainName
TrainType
TrainLine
TrainCapacity
TrainNumberOfCars
TotalCapacity

Station
StationID (PK)
StationName
StationDivision
StationLine
StationBorough

Route
RouteID (PK)
DepartureStationID (FK)
TerminationStationID (FK)
RouteDistance
RouteDuration
AverageRouteDuration_inMins

Train_Schedule
TrainScheduleID (PK)
TrainID (FK)
RouteID(FK)
ConductorID (FK)
DepartureTime
ArrivalTime

Employees
EmployeeID (PK)
FirstName
LastName
Position
EmployeeEmail
EmployeeStreetAddress
EmployeeCityAddress
EmployeeStateAddress
EmployeeZipcode
EmployeePhone

Conductor
ConductorID (PK)
EmployeeID (FK)
CertificationStatus

Employee_Schedule
EmployeeScheduleID (PK)
TrainScheduleID (FK)
EmployeeID (FK)
ShiftStartTime
ShiftEndTime
DaysScheduled

RELATIONSHIP SENTENCES

Each train schedule is associated with a specific train.

one-to-one relationship between Train_Schedule and Train.

Each train operates on a specific route.

one-to-one relationship between Train_Schedule and Route.

Each route connects a departure station to a termination station.

Each employee has one schedule.

one-to-one relationship between Employees and Employee_Schedule.

One Employee may have multiple Train Schedules in the Employee_Schedule" table

Each conductor is associated with only one employee.

one-to-one relationship between Conductor and Employees.

Conversion to Relational Model:

Train (TrainID (**PK**), TrainName, TrainType, TrainLine, TrainCapacity,TrainNumberOfCars)

Station (StationID **(PK**),StationName, StationDivision, StationBorough, StationLine)

Route (RouteID (**PK**), Description, DepartureStationID, TerminationStationID, RouteDistance, RouteDuration)

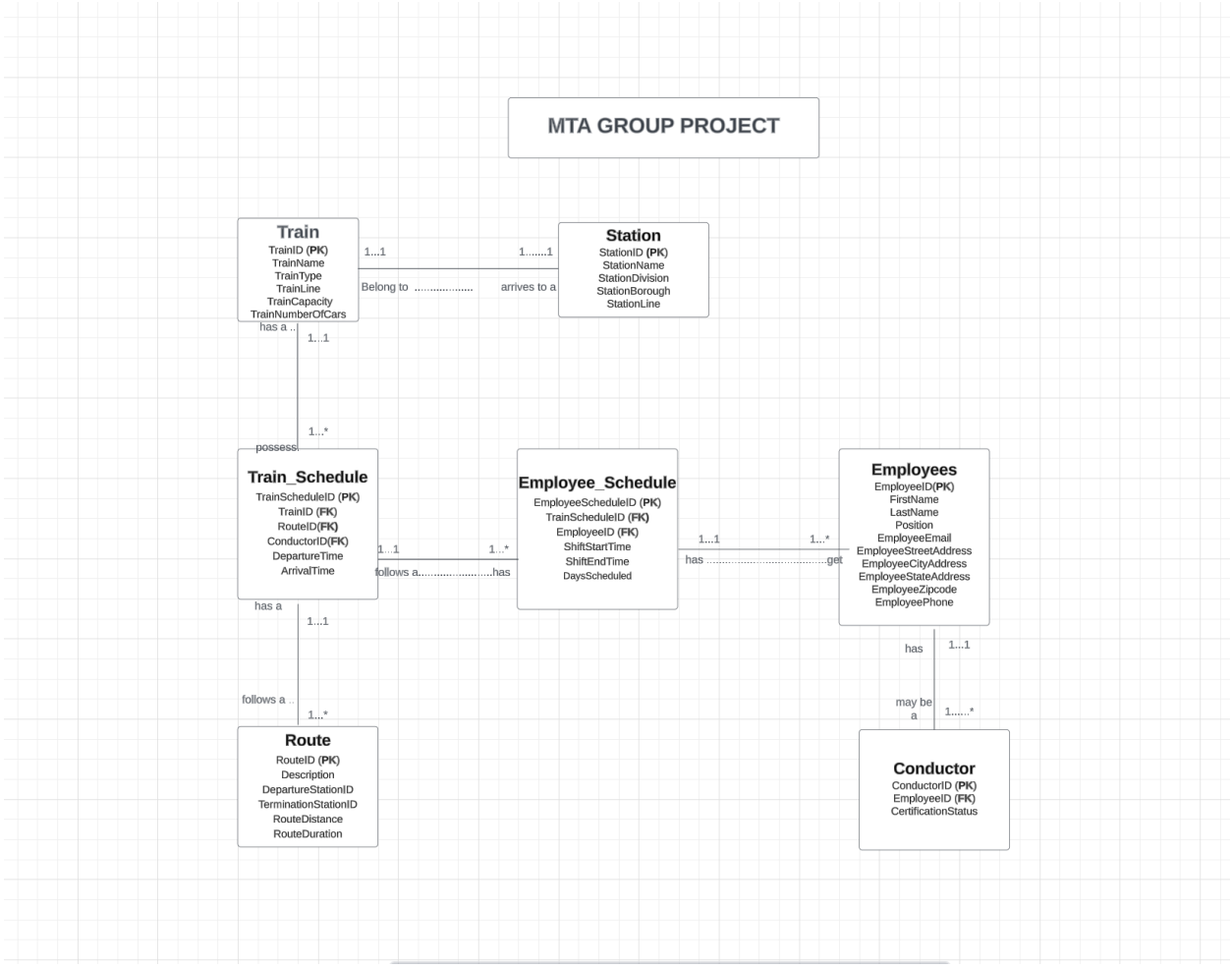
Train_Schedule (TrainScheduleID (**PK**), TrainID (**FK**),RouteID(**FK)**,ConductorID(**FK**), DepartureTime, ArrivalTime)

Employees(EmployeeID(**PK**),FirstName,LastName,Position,EmployeeEmail,EmployeeStreetAddress,EmployeeCityAddress, EmployeeStateAddress, EmployeeZipcode, EmployeePhone)

Conductor (ConductorID (**PK**),EmployeeID (**FK**), CertificationStatus)

Employee_Schedule (EmployeeScheduleID (**PK**), TrainScheduleID (**FK)**, EmployeeID (**FK**), ShiftStartTime, ShiftEndTime, DaysScheduled)

UML NOTATION



Report:

MTA train stations are some of the most popular in the entire country, used by everyone, everywhere, at all times for their needs. We, for years by now, have been using a spreadsheet to keep track of a plethora of things, including but not limited to: trains, stations, routes, schedules(Both train and employee), and workers(Employees and Conductors). Recently, we found out that the spreadsheet is becoming outdated and messy to keep track of all information, so we decided to switch to a database.

Our stations face the problem of too many people and too many schedules and times to keep track off, which racks up the lines and crowds, which creates delays, which causes rush hours during lines 1, 2, 3, Q, N, and R. So what we did was examine the stations(Division, Line, Borough, Departure, and Termination) without forgetting about the routes(Its distance and duration). We'll also check the trains like their schedule, of when they arrive and stop, so that we can always be prepared and know in advance when and how a train will be in place.

We also need to keep track of our workers, which are divided into Employees and Conductors. For the employees we look at their contact information(First/Last name, E-Mail, Address and more) while the conductors we only look at their ID and whether they're certified or not. We also need to look at the Employee Schedule(Star/Endshift and Days Scheduled). With this, we'd be able to keep track of every detail necessary to keep trains running while minimizing the delays.

Normalization:

Sample data:

TrainID	TrainName	TrainType	TrainLine	TrainCapacity	TrainNumberOfCars
C54	A	R46 (B Division)	IND Eighth Avenue Line	1150	10
C55	A	R46 (B Division)	IND Eighth Avenue Line	1150	10
C56	A	R143 (B Division)	IND Eighth Avenue Line	1080	10
C57	C	R143 (B Division)	IND Eighth Avenue Line	1080	9
C58	C	R143 (B Division)	IND Eighth Avenue Line	1080	9
C59	C	R143 (B Division)	IND Eighth Avenue Line	1080	9
C60	E	R143 (B Division)	IND Eighth Avenue Line	1080	11
C61	E	R143 (B Division)	IND Eighth Avenue Line	1080	11
C62	E	R143 (B Division)	IND Eighth Avenue Line	1080	11
C63	B	R143 (B Division)	IND Sixth Avenue Line	1080	11
C64	B	R143 (B Division)	IND Sixth Avenue Line	1080	11
C65	B	R143 (B Division)	IND Sixth Avenue Line	1080	11

Trian Relation

Train (TrainID (**PK**), TrainName, TrainType, TrainLine, TrainCapacity,TrainNumberOfCars)

Key: TrainID

FD1: TrainID -> TrainName, TrainType, TrainLine, TrainCapacity,TrainNumberOfCars

1NF: Meets the definition of a relation

2NF: No partial Key dependencies

3NF: No Transitive dependencies

Station Relation:

Sample data:

StationID	StationName	StationDivision	StationLine	StationBorough
101	Marcy Av	BMT	Jamaica	Bk
232	2 Av	IND	6th Av - Culver	M
364	Zerega Av	IRT	Pelham	Bx
454	74 St-Broadway	IRT	Flushing	Q
222	Roosevelt Island	IND	63rd St	M
215	Tremont Av	IND	Concourse	Bx
309	103 St	IRT	Broadway - 7Av	M
84	85 St-Forest Pkwy	BMT	Jamaica	Q
354	Sterling St	IRT	Nostrand	Bk
468	Times Sq-42 St	IRT	Lexington - Shuttle	M
123	Grand St	BMT	Canarsie	Bk
337	Nevins St	IRT	Eastern Pky	Bk
322	14 St	IRT	Broadway - 7Av	M

Station (StationID (**PK**), StationName, StationDivision, StationBorough, StationLine)

Key: StationID

FD1: StationID -> StationName, StationDivision, StationBorough, StationLine

1NF: Meets the definition of a relation

2NF: No partial Key dependencies

3NF: No Transitive dependencies

Route Relation:

Sample data:

RouteID	Description	DepartureStationID	TerminationStationID	RouteDistance_inMiles	RouteDuration_inMins
1	1 Train: Broadway–Seventh Avenue Local	293	330	13.5	30 - 40
2	2 Train: Seventh Avenue Express	416	359	18.6	55 - 60
3	3 Train: Seventh Avenue Express	436	352	23.2	70 - 75
4	4 Train: Lexington Avenue Express	378	345	23.9	65 - 70
5	5 Train: Lexington Avenue Express	417	359	20.7	65 - 70
6	6 Train: Lexington Avenue Local/Pelham Local	360	411	13.9	40 - 45
7	7 Train: Flushing Local/Express	447	471	11.4	30 - 35
8	A Train: Eighth Avenue Express	143	209	31	75 - 80
9	B Train: Sixth Avenue Express	151	55	15	45 - 50
10	C Train: Eighth Avenue Local	148	188	19	50 - 55
11	D Train: Sixth Avenue Express	210	58	31.5	55 - 60

Route (RouteID (**PK**), Description, DepartureStationID, TerminationStationID, RouteDistance, RouteDuration)

Key: RouteID

FD1: RouteID -> Description, DepartureStationID, TerminationStationID, RouteDistance, RouteDuration

1NF: Meets the definition of a relation

2NF: No partial Key dependencies

3NF: No Transitive dependencies

Train_Schedule Relation:

Sample data:

TrainScheduleID	TrainID	Description	ConductorID	ArrivalTime	DepartureTime
MNG	C54	1 Train: Broadway–Seventh Avenue Local	1001	5:00	5:10
AFT	C55	1 Train: Broadway–Seventh Avenue Local	1002	12:05	12:10
NHT	C56	1 Train: Broadway–Seventh Avenue Local	1003	9:00	9:10
MNG	C57	2 Train: Seventh Avenue Express	1004	5:00	5:10
AFT	C58	2 Train: Seventh Avenue Express	1005	12:05	12:10
NHT	C59	2 Train: Seventh Avenue Express	1006	9:00	9:10
MNG	C60	3 Train: Seventh Avenue Express	1007	5:00	5:10
AFT	C61	3 Train: Seventh Avenue Express	1008	12:05	12:10
NHT	C62	3 Train: Seventh Avenue Express	1009	9:00	9:10

Train_Schedule (TrainScheduleID, TrainID, Description, ConductorID, DepartureTime, ArrivalTime)

Key: TrainScheduleID, TrainID

FD1: TrainScheduleID, TrainID -> Description, ConductorID, DepartureTime, ArrivalTime

FD2: TrainID -> TrainScheduleID

FD3: ConductorID -> Description

1NF: Meets the definition of a relation

2NF: Partial Key dependencies exists: TrainScheduleID, TrainID

3NF: Transitive dependencies exists: ConductorID -> Description

Solution: Split Train_Schedule relation into two new relations named TrainData and ConductorData:
TrainData: (TrainScheduleID, TrainID)
Key: TrainID
FD1: TrainID -> TrainScheduleID
ConductorData: (ConductorID, Description)
Key: ConductorID
FD1: ConductorID -> Description

Train_Schedule: (TrainScheduleID, TrainID, DepartureTime, ArrivalTime)
Key: TrainScheduleID, TrainID
FD1: TrainScheduleID, TrainID -> DepartureTime, ArrivalTime

Employees(EmployeeID(**PK**),FirstName,LastName,Position,EmployeeEmail,EmployeeStreetAddress,EmployeeCityAddress, EmployeeStateAddress, EmployeeZipcode, EmployeePhone)

Sample data:

EmployeeID	FirstName	LastName	Position	EmployeeEmail	EmployeeStreetAddress	EmployeeCityAddress	EmployeeStateAddress	EmployeeStreetZipcode
3382	Rhett	York	Conductor	RhettYork@MTA.com	9904 Fairview Ave.	Buffalo	NY	14221
3383	Jimmy	Patel	Conductor	JimmyPatel@MTA.com	8793 Roberts Rd.	Staten Island	NY	10312
3384	Saniya	Padilla	Conductor	SaniyaPadilla@MTA.com	845 White Drive	Astoria	NY	11106
3385	Kenny	Gilmore	Conductor	KennyGilmore@MTA.com	414 E. Canal Ave.	Jamaica	NY	11435
3386	Bridget	Bryant	Conductor	BridgetBryant@MTA.com	571 Cleveland St.	North Tonawanda	NY	14120
3387	Trystan	Chavez	Conductor	TrystanChavez@MTA.com	179 Brickell Ave.	Jamaica	NY	11432
3388	Garv	Hollowav	Conductor	GarvHollowav@MTA.com	414 3rd Rd.	Brooklvn	NY	11211

Key: EmployeeID
FD1: EmployeeID -> FirstName,LastName,Position,EmployeeEmail,EmployeeStreetAddress,EmployeeCityAddress, EmployeeStateAddress, EmployeeZipcode, EmployeePhone
FD2: EmployeeZipcode -> EmployeeCityAddress, EmployeeStateAddress
1NF: Meets the definition of a relation
2NF: No partial Key dependencies
3NF: Transitive dependencies exists: ConductorID -> EmployeeZipcode -> EmployeeCityAddress, EmployeeStateAddress

Solution: Split Employees relation into two new relations named EmployeesData and ZipCodes:
EmployeesData:(EmployeeID,FirstName,LastName,Position,EmployeeEmail, EmployeeStreetAddress, EmployeePhone)
Key: EmployeeID
FD1: EmployeeID -> FirstName,LastName,Position,EmployeeEmail, EmployeeStreetAddress, EmployeePhone

ZipCodes: (EmployeeZipcode, EmployeeCityAddress, EmployeeStateAddress)
Key: EmployeeZipcode
FD1: EmployeeZipcode -> EmployeeCityAddress, EmployeeStateAddress

Conductor (ConductorID (**PK**),EmployeeID (**FK**), CertificationStatus)
Sample data:

ConductorID	EmployeeID	CertificationStatus
1001	3382	certified
1002	3383	certified
1003	3384	certified
1004	3385	certified
1005	3386	certified
1006	3387	certified
1007	3388	certified
1008	3389	certified
1009	3390	certified
1010	3391	certified
1011	3392	certified
1012	3393	certified
1013	3394	certified

Key: ConductorID
FD1: ConductorID -> EmployeeID, CertificationStatus
1NF: Meets the definition of a relation
2NF: No partial Key dependencies
3NF: No Transitive dependencies

Employee_Schedule (EmployeeScheduleID (**PK**), TrainScheduleID (**FK)**, EmployeeID (**FK**), ShiftStartTime, ShiftEndTime, DaysScheduled)

Sample data:

EmployeeScheduleID	TrainScheduleID	EmployeeID	ShiftStartTime	ShiftEndTime	DaysScheduled
101	MNG	3382	4:40AM	12:40PM	MON-THUR
102	AFT	3383	11:40AM	7:40PM	MON-THUR
103	NHT	3384	7:00PM	3:00AM	MON-THUR
104	MNG	3385	4:40AM	12:40PM	MON-THUR
105	AFT	3386	11:40AM	7:40PM	MON-THUR
106	NHT	3387	7:00PM	3:00AM	MON-THUR
107	MNG	3388	4:40AM	12:40PM	MON-THUR
108	AFT	3389	11:40AM	7:40PM	MON-THUR
109	NHT	3390	7:00PM	3:00AM	MON-THUR
110	MNG	3391	4:40AM	12:40PM	MON-THUR
111	AFT	3392	11:40AM	7:40PM	MON-THUR

Key: EmployeeScheduleID
FD1: EmployeeScheduleID -> TrainScheduleID, EmployeeID, ShiftStartTime, ShiftEndTime, DaysScheduled
FD2: EmployeeID -> TrainScheduleID
1NF: Meets the definition of a relation
2NF: No partial Key dependencies
3NF: Transitive dependencies exists: EmployeeID -> TrainScheduleID

Solution: Split Employee_Schedule relation into two new relations named EmployeeSchedule and TrainSchedule:
EmployeeSchedule: (EmployeeScheduleID, ShiftStartTime, ShiftEndTime, DaysScheduled)
Key: EmployeeScheduleID -> ShiftStartTime, ShiftEndTime, DaysScheduled

TrainSchedule: (EmployeeID, TrainScheduleID)
Key: EmployeeID
FD1: EmployeeID -> TrainScheduleID

Queries

As illustrated in the Route Sheet table, the maximum route distance is 31.5 miles, with the minimum being 0.7 miles. Given that there is minimal variation between route distances, identifying the longest average route duration is crucial for establishing a stable and balanced train schedule.

To determine the five longest average route durations in minutes by station, I used the following code:

```
SELECT TOP 5 *
FROM Route
ORDER BY AverageRouteDuration_inMins DESC;
```

Average Route Duration in Mins

2023년 12월 8일 금요일
10:19:11 오후

RouteID	DepartureStationID	TerminationStationID	RouteDistance_inMiles	AverageRouteDuration_inMins
A Train: Eighth Avenue Express	143	209	31	77.5
3 Train: Seventh Avenue Express	436	352	23.2	73.5
4 Train: Lexington Avenue Express	378	345	23.9	68.5
5 Train: Lexington Avenue Express	417	359	20.7	68.5
Q Train: Broadway Express/Brighton Express	310	58	31.5	65

5

Page 1 of 1

The results of the above query revealed that, in relation to the route distances in miles, Trains 3, 4, and 5 face challenges in providing stable schedules to passengers. For example, a 31-mile route takes 77.5 minutes, and a 23.2-mile route takes 73.5 minutes.

To investigate the reasons behind these durations, I selected data from the "Trains" sheet to analyze the total capacity of each train route among A, 3, 4, 5, and Q with the following code:

```
SELECT *
FROM Trains
WHERE TrainName IN ('A', '3', '4', '5', 'Q');
```

Query2 2023년 12월 8일 금요일 10:16:41 오후

TrainID	TrainName	TrainType	TrainLine	TrainCapacity	TrainNumberOfCars	Total Capacity
C54	A	R46 (B Division)	IND Eighth Avenue Line	1150	10	11500
C55	A	R46 (B Division)	IND Eighth Avenue Line	1150	10	11500
C56	A	R143 (B Division)	IND Eighth Avenue Line	1080	10	10800
C93	Q	R46 (B Division)	BMT Broadway Line	1150	11	12650
C94	Q	R46 (B Division)	BMT Broadway Line	1150	11	12650
C95	Q	R143 (B Division)	BMT Broadway Line	1080	11	11880
C105	3	R62 (A Division)	IRT Broadway–Seventh Avenue Line	1200	11	13200
C106	3	R142 (A Division)	IRT Broadway–Seventh Avenue Line	1080	11	11880
C107	3	R142 (A Division)	IRT Broadway–Seventh Avenue Line	1080	11	11880
C108	4	R142 (A Division)	IRT Lexington Avenue Line	1080	11	11880
C109	4	R188 (A Division)	IRT Lexington Avenue Line	1080	11	11880
C110	4	R188 (A Division)	IRT Lexington Avenue Line	1080	11	11880
C111	5	R188 (A Division)	IRT Lexington Avenue Line	1080	11	11880
C112	5	R142 (A Division)	IRT Lexington Avenue Line	1080	11	11880
C113	5	R142 (A Division)	IRT Lexington Avenue Line	1080	11	11880

15

Page 1 of 1

From this data, it was discovered that among the five train routes—A, Q, 3, 4, and 5—there is no significant capacity difference between routes A and Q when compared to 3, 4, and 5. Therefore, it is vital to either increase the capacity of trains for lines 3, 4, and 5 or to add more trains to these lines to stabilize schedule regularity.