

Railway Train Tracker

CSCI 370 Project

By: Andrew Gair

Introduction:

The application presented is a Railway Management software intended to be used by workers at the given railway network. The project is a snapshot of a real-time system that would track trains, cargo, personnel, and stations along the rail network. It is assumed that in the real-time hypothetical version of the software, all updates would be provided every minute, this program would then execute queries based on the input received using the data provided every minute.

Technical notes:

For the purposes of this project, the current time is assumed to be 13:02.

At midnight each night, the program will update planned routes with the day's schedule. If a train or cargo item is listed at a station at 00:00, that means the item is either starting the day at that location, or has reached its final destination of the day.

Application Requirements:

The application functionality includes:

- The ability to see all stops a given train makes throughout the day and at what time those stops are planned to occur at, and to also see where a train is at the current time.

This is done through the "stops" command. The user is prompted to enter a train ID, the program will list details relevant to that train.

- The ability to see which trains travel to a given station throughout the day, and at what time those stops occur at.

This is done through the "traffic" command. The user is prompted to enter a station ID, the program will list details relevant to that station.

- The ability to see cargo that is currently on the rail network, and to track where that cargo has been, where it is going, and where it currently is.

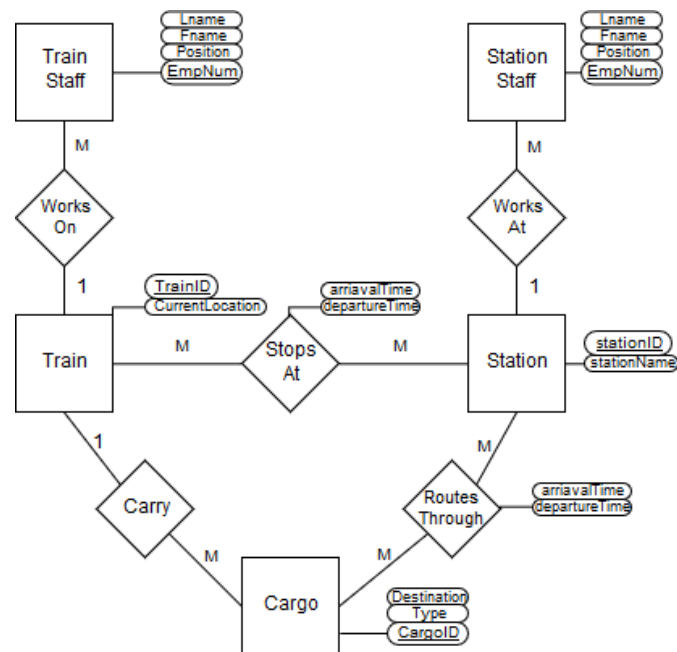
This is done through the "routes" command. The user is prompted to enter a cargo ID, the program will list details relevant to that cargo.

- The ability to see staff currently working on the rail network.

This is done through the "staff" command. The user is prompted to enter either "train" or "station", depending on which staff the user wants to view.

Database Design:

ER-Diagram:



Relational Schema:

```
CREATE TABLE trains(trainID CHAR(6), currentLocation VARCHAR(20), PRIMARY KEY(trainID));
```

```
CREATE TABLE trainStaff(empNum CHAR(6), Lname VARCHAR(15), Fname VARCHAR(15), position VARCHAR(10), trainID CHAR(6), PRIMARY KEY(empNum));
```

```
CREATE TABLE stationStaff(empNum CHAR(6), Lname VARCHAR(15), Fname VARCHAR(15), position VARCHAR(10), stationID CHAR(6), stationName VARCHAR(20), PRIMARY KEY(empNum));
```

```
CREATE TABLE stations(stationID CHAR(6), stationName VARCHAR(20), PRIMARY KEY(stationID));
```

```
CREATE TABLE stopsAt(stationID CHAR(6), trainID CHAR(6), arrivalTime char(5), departureTime char(5), PRIMARY KEY(stationID, trainID));
```

```
CREATE TABLE cargo(cargoID CHAR(6), type VARCHAR(15), destination VARCHAR(20), trainID CHAR(6), PRIMARY KEY(cargoID));
```

```
CREATE TABLE routesThrough(stationID CHAR(6), cargoID CHAR(6), arrivalTime CHAR(5), departureTime CHAR(5), PRIMARY KEY(stationID, cargoID));
```

Sample Data: shown table by table

-Train Staff

```
INSERT INTO trainStaff(empNum, Lname, Fname, position, trainID) VALUES ('101537', 'Johnson', 'Fred', 'engineer', '502667');
INSERT INTO trainStaff(empNum, Lname, Fname, position, trainID) VALUES ('101923', 'Andrews', 'Andy', 'attendant', '502667');
INSERT INTO trainStaff(empNum, Lname, Fname, position, trainID) VALUES ('101886', 'Bob', 'Bella', 'engineer', '502703');
INSERT INTO trainStaff(empNum, Lname, Fname, position, trainID) VALUES ('101124', 'Jacobs', 'Jack', 'attendant', '502703');
```

-Station Staff

```
INSERT INTO stationStaff(empNum, Lname, Fname, position, stationID, stationName) VALUES ('102231', 'Holden', 'James', 'clerk', '200667', 'Medina Station');
INSERT INTO stationStaff(empNum, Lname, Fname, position, stationID, stationName) VALUES ('102367', 'Smith', 'John', 'clerk', '200723', 'DS9 Station');
INSERT INTO stationStaff(empNum, Lname, Fname, position, stationID, stationName) VALUES ('102514', 'Doe', 'Jane', 'clerk', '200327', 'Phalanx Station');
INSERT INTO stationStaff(empNum, Lname, Fname, position, stationID, stationName) VALUES ('102314', 'Sue', 'Sally', 'manager', '200765', 'VDNkh Station');
```

-Trains

```
INSERT INTO trains(trainID, currentLocation) VALUES ('502667', 'Medina Station');
INSERT INTO trains(trainID, currentLocation) VALUES ('502703', '0:11 VDNkh Station');
```

-Stations

```
INSERT INTO stations(stationID, stationName) VALUES ('200667', 'Medina Station');
INSERT INTO stations(stationID, stationName) VALUES ('200723', 'DS9 Station');
INSERT INTO stations(stationID, stationName) VALUES ('200327', 'Phalanx Station');
INSERT INTO stations(stationID, stationName) VALUES ('200765', 'VDNkh Station');
```

-Stops At

```
INSERT INTO stopsAt(stationID, trainID, arrivalTime, departureTime) VALUES ('200765', '502667', '00:00', '08:30');
INSERT INTO stopsAt(stationID, trainID, arrivalTime, departureTime) VALUES ('200723', '502667', '10:23', '12:00');
INSERT INTO stopsAt(stationID, trainID, arrivalTime, departureTime) VALUES ('200667', '502667', '13:33', '00:00');
INSERT INTO stopsAt(stationID, trainID, arrivalTime, departureTime) VALUES ('200327', '502703', '00:00', '10:00');
INSERT INTO stopsAt(stationID, trainID, arrivalTime, departureTime) VALUES ('200723', '502703', '18:16', '00:00');
```

-Cargo

```
INSERT INTO cargo(cargoID, type, destination, trainID) VALUES ('606502', 'natural gas', '200667', '502667');
INSERT INTO cargo(cargoID, type, destination, trainID) VALUES ('606503', 'coal', '200667', '502667');
INSERT INTO cargo(cargoID, type, destination, trainID) VALUES ('606121', 'coal', '200723', '0');
INSERT INTO cargo(cargoID, type, destination, trainID) VALUES ('606998', 'Passengers', '200723', '502703');
```

-Routes Through

```
INSERT INTO routesThrough(stationID, cargoID, arrivalTime, departureTime) VALUES ('200765', '606502', '00:00', '08:30');
```

```
INSERT INTO routesThrough(stationID, cargoID, arrivalTime, departureTime) VALUES ('200723',
'606502', '10:23', '12:00');
INSERT INTO routesThrough(stationID, cargoID, arrivalTime, departureTime) VALUES ('200667',
'606502', '13:33', '00:00');
INSERT INTO routesThrough(stationID, cargoID, arrivalTime, departureTime) VALUES ('200765',
'606503', '00:00', '08:30');
INSERT INTO routesThrough(stationID, cargoID, arrivalTime, departureTime) VALUES ('200723',
'606503', '10:23', '12:00');
INSERT INTO routesThrough(stationID, cargoID, arrivalTime, departureTime) VALUES ('200667',
'606503', '13:33', '00:00');
INSERT INTO routesThrough(stationID, cargoID, arrivalTime, departureTime) VALUES ('200723',
'606121', '10:23', '00:00');
INSERT INTO routesThrough(stationID, cargoID, arrivalTime, departureTime) VALUES ('200327',
'606998', '00:00', '10:00');
INSERT INTO routesThrough(stationID, cargoID, arrivalTime, departureTime) VALUES ('200723',
'606998', '18:16', '00:00');
```

Sample Runs:

Run 1) select train staff.

Will select staff from the options menu.

```
Please enter a cmd:  select
please specify what to select from:
'staff' to see station or train staff
'stops' to see stops a train makes throughout the day
'routes' to see cargo as it travels
'traffic' to see which trains stop at a station
'back' to leave select menu

Please enter a cmd:  staff
```

This will open a sub-menu asking for either train or station staff. The user can also go back to the select menu if required.

```
please enter one of the following to select which staff will be shown:
'train'
'station'
'back'

Please enter a cmd:  train
```

This command then displays the results of the query. And brings user back to select-menu.

```
Please enter a cmd:  train

+-----+-----+-----+-----+
| EmpNum | Lname | Fname | Position | trainID |
+-----+-----+-----+-----+
| 101537 | Johnson | Fred | engineer | 502667 |
| 101923 | Andrews | Andy | attendant | 502667 |
| 101886 | Bob | Bella | engineer | 502703 |
| 101124 | Jacobs | Jack | attendant | 502703 |
+-----+-----+-----+-----+

please specify what to select from:
'staff' to see station or train staff
'stops' to see stops a train makes throughout the day
'routes' to see cargo as it travels
'traffic' to see which trains stop at a station
'back' to leave select menu

Please enter a cmd:  
```

Run 2) select the stops a train makes throughout the day

The user enters “stops” to bring up the selection menu for trains.

```
please specify what to select from:
'staff' to see station or train staff
'stops' to see stops a train makes throughout the day
'routes' to see cargo as it travels
'traffic' to see which trains stop at a station
'back' to leave select menu

Please enter a cmd:  stops
```

The user then enters a train ID from the table shown. An incorrect train ID will result in an error and the user will be returned to the select menu.

```
List of all trains currently on the rails:

+-----+-----+
| trainID | currentLocation |
+-----+-----+
| 502667 | 00:21 Medina Station |
| 502703 | 05:14 DS9 Station |
+-----+-----+

please enter a valid trainID to look for, or enter 'back' to go back to select menu:

Please enter a cmd:  502667
```

The results of the query are shown, and the user is returned to the select menu (not shown).

```
The train with ID 502667 stops at the following stations:

+-----+-----+-----+-----+
| StationID | StationName | arrivalTime | departureTime |
+-----+-----+-----+-----+
| 200765 | VDNkh Station | 00:00 | 08:30 |
| 200723 | DS9 Station | 10:23 | 12:00 |
| 200667 | Medina Station | 13:33 | 00:00 |
+-----+-----+-----+-----+
```

Run 3) select a given piece of cargo and see its route through the train network.

The user enters “routes” into the selection menu

```
please specify what to select from:
'staff' to see station or train staff
'stops' to see stops a train makes throughout the day
'routes' to see cargo as it travels
'traffic' to see which trains stop at a station
'back' to leave select menu

Please enter a cmd:  routes
```

The user enters a cargo ID from the table shown, incorrect ID's will result in an error and return the user to the select menu.

```
+-----+-----+-----+-----+
| cargoID |      Type      | destination | trainID |
+-----+-----+-----+-----+
| 606502 | natural gas | 200667 | 502667 |
| 606503 | coal | 200667 | 502667 |
| 606121 | coal | 200723 | 0 |
| 606998 | Passengers | 200723 | 502703 |
+-----+-----+-----+-----+
please enter a valid cargoID to look for, or enter 'back' to go back to select menu:

Please enter a cmd:  606502
```

The results of the query are shown, and the user is returned to the select menu (not shown).

```
The cargo with ID 606502 has the following stops scheduled:
+-----+-----+-----+-----+
|      stationName      | arrivalTime | departureTime |
+-----+-----+-----+-----+
|      VDNkh Station    | 00:00 | 08:30 |
|      DS9 Station      | 10:23 | 12:00 |
|      Medina Station    | 13:33 | 00:00 |
+-----+-----+-----+-----+
```

Run 4) see what traffic passes through a given station throughout the day.

The user selects “traffic” from the select menu

```
please specify what to select from:
'staff' to see station or train staff
'stops' to see stops a train makes throughout the day
'routes' to see cargo as it travels
'traffic' to see which trains stop at a station
'back' to leave select menu

Please enter a cmd:  traffic
```

The user is prompted to enter a station ID, an incorrect ID would return the user to the select menu and display an error message.

```
List of all stations on this rail network:
+-----+-----+
| StationID | StationName |
+-----+-----+
| 200667 | Medina Station |
| 200723 | DS9 Station |
| 200327 | Phalanx Station |
| 200765 | VDNkh Station |
+-----+-----+
please enter a valid stationID to look for, or enter 'back' to go back to select menu:

Please enter a cmd: 200667
```

The results of the query are shown

```
The station with ID 200667 has the following trains stop at it:
+-----+-----+-----+
| trainID | arrivalTime | departureTime |
+-----+-----+-----+
| 502667 | 13:33 | 00:00 |
+-----+-----+-----+
```