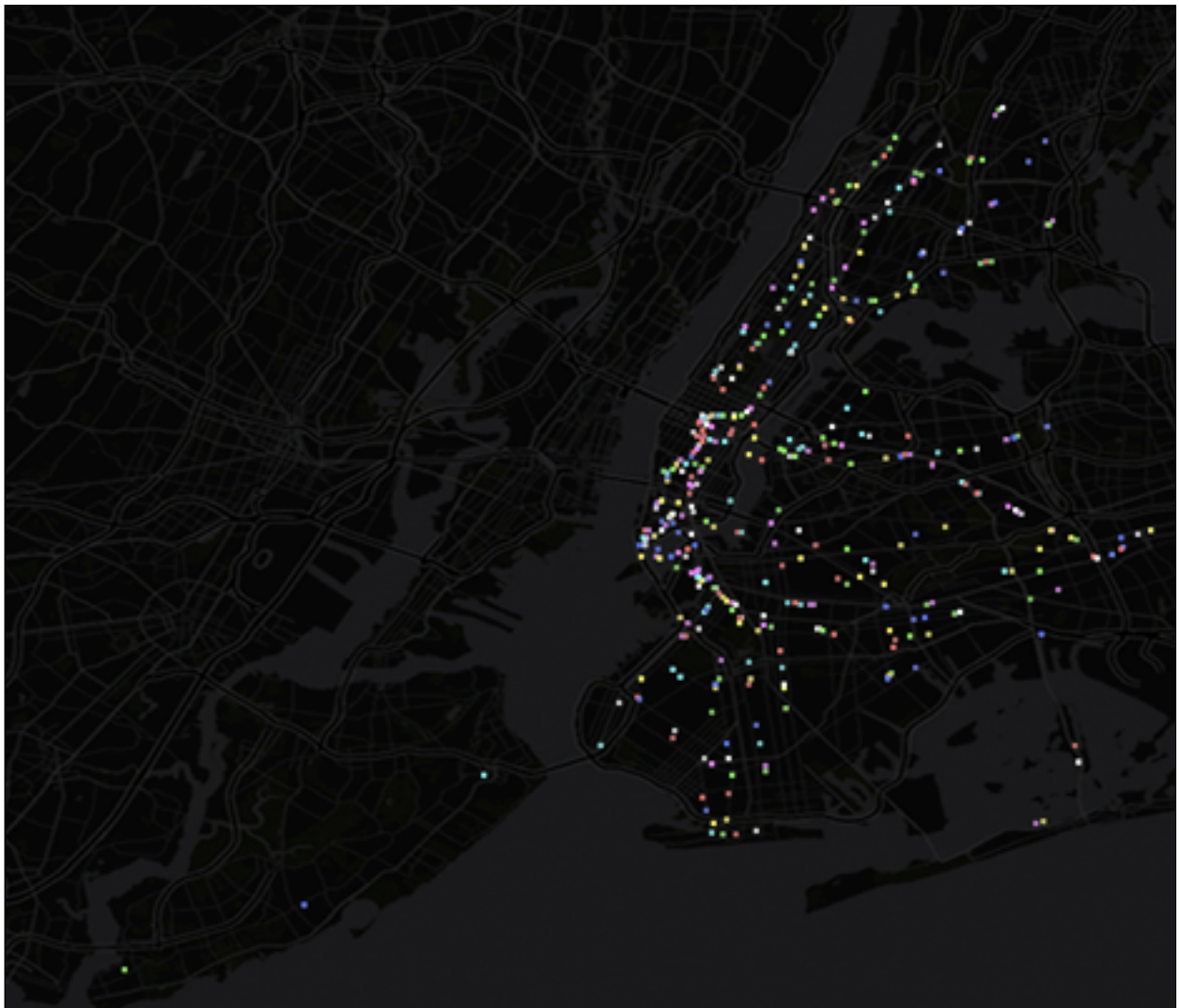

NYC Transit Interactive Visualization

New York University Shanghai

CSCI-SHU 370-001 Object Oriented Programming

Jack B. Du (Jiadong Du) - May 15, 2015



Introduction

This project is an NYC transit interactive visualization built in Java. MTA New York City Transit is the largest public transportation agency in North America and one of the largest in the world. The subway has a daily ridership of 5.6 million and an annual ridership in 2014 of roughly 1.751 billion. Our fleet of more than 6,300 subway cars traveled approximately 361.1 million miles in 2014, along 660 miles of track, 24 hours a day, seven days a week, although not all routes operate around the clock. There are 468 subway stations, including 109 made accessible to customers with disabilities, via elevators and ramps. NYC Transit Subway serves Brooklyn, the Bronx, Manhattan, and Queens, making 7,883 weekday trips. MTA Staten Island Railway (SIR) serves Staten Island and has four ADA-accessible stations. ([MTA](#))



The purpose of this project is to provide to the audience a visually pleasing way to view the NYC transit, both for study and for arts. The application simulates the positions of all the public vehicles.

Code Architecture

App.java

App is a class that extends JPanel, where all the visualization is being painted. This class stores most of the data that needs to be displayed or transformed and also handles most of the events that are required for the interaction.

In `public void paint(Graphics g)` method, it loads an external image and displays it at the bottom layer of the graphics. Then as long as the routes are not hidden, it will go through the ArrayList of the shapes and paint all the lines. Since there are so many lines, which take up a lot of CPU to paint so that it slows down the whole application, I disabled it for default so that the animations are much smoother and it looks more pleasing. After painting the lines, it paints the vehicles. Basically, it goes through all the trajectories for all the vehicles and finds the corresponding position for the current simulated time. It chooses a random color from a predefined color array and assign it to each vehicles so that different vehicles are distinctive. Also, the background image is very dark so that the vehicles, which are bright dots, would really stand out from the background and look like little lightbulbs at night so that they are beautiful.

`public void parseRoutes()` method parses all the routes from the file and store the route names in an array so that the user can loop through each route in single route mode. `public void prevRoute()` navigates to the previous route while `public void nextRoute()` navigates to the next route. There are also methods that handles “play/pause” and “show/hide” and `public void setDay()` that switch between Sunday, Saturday and Workday. `public void setSpeed()` sets the speed, which actually changes the time that the thread sleeps. `public void setTime()` sets the time of a day from 00:00:00 to 23:59:59 while the `public void update()` updates the simulated time.

Finally, the `public static void main(String[] args)` is where the most high level stuff is going on. This whole application starts from this method. It first parses all the trips that are in the data file and converts it to an ArrayList of Trajectories so that it could be later looped though and painted. It creates the JFrame, where the Panel and all other components are placed and it also creates all the buttons and scroll bars. It handles all the events that are related to those buttons and it controls the main flow of the visualization, or in other words, those animations.

Features

Simulation

Conclusion

The NYC transit interactive visualization not just visualizes the NYC transit data for academic or research use, but also a piece of interactive art itself. Interactive Media Arts is still relatively a new major and new field, but since I am majoring in both Computer Science and Interactive Media Arts, I will try, in my future career, to integrate those two subjects seamlessly. I like technology because technology is changing our lives and technology helps me to think reasonably and scientifically. I like arts because arts help me to relax and arts help me to achieve immortality. Thus, I believe that the integration of technology and arts, or in other words, the combination of Computer Science and Interactive Media Arts will really help me to make the world a big difference. Just like this project, it not only shows my ability to make use of my own technical skills but also conveys my own way to express artistic concepts through technical project.