2024.02.21

Real-Time Surrounding Transport Information

Server to provide RSTI

SDV본부 플랫폼팀 김인호

CONTROLWORKS

CONTENTS

1. Introduction
2. Server&Client
3. Verification
4. Conclusion

Introduction

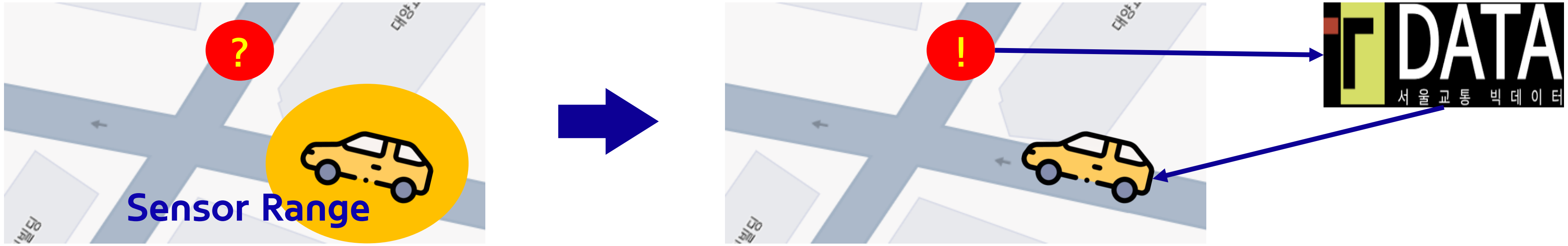
Background

Real-Time Surrounding Transport Information (RSTI)

- Road environment which can be changed around a specific vehicle

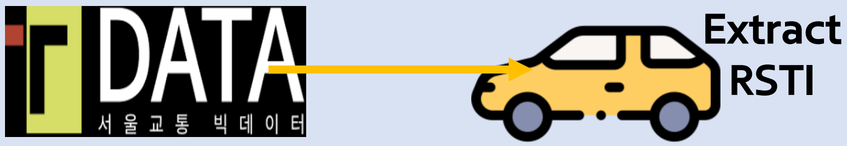
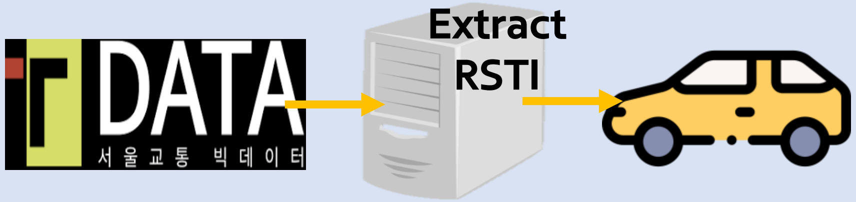
How autonomous-driving can recognize RSTI?

- Sensor
  - Normal method, but it cannot process the case beyond a sensor range.
- OpenAPI Communication
  - OpenAPI provides whole range traffic information.
  - Car can get accurate&quick information which sensor cannot see.



Background

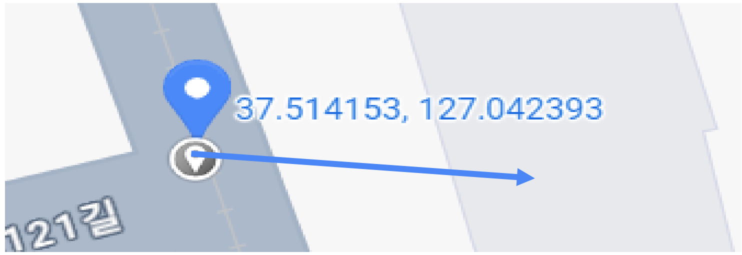
■ How RSTI can be delivered from OpenAPI to Car?

Method	Direct	Server
Structure		
Pros	Reduce transfer delay	According to new region, Only server should be updated (Server process extracting RSTI)
Cons	According to new region, All cars should be updated. (Different region has different ITS)	Increase transfer delay

Goal

■ Objective

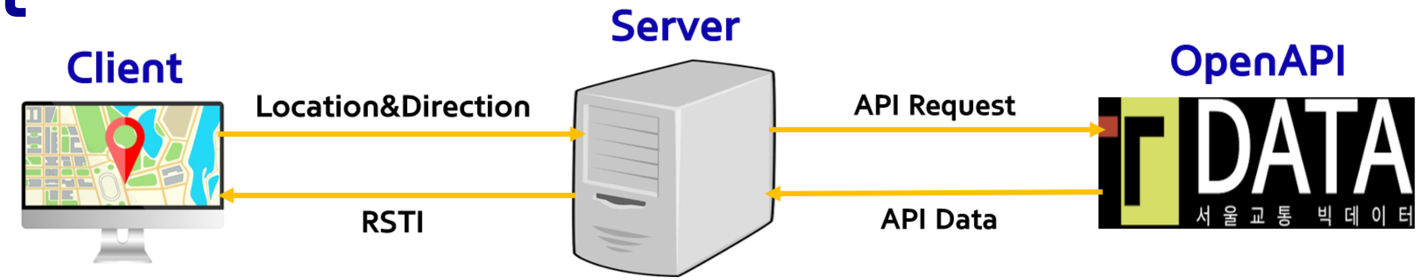
- ▶ Project range : develop prototype before being applied to real vehicles.
- ▶ Client : virtual location&direction of a point on the map.
- ▶ API : Seoul trafficlight/velocity/emergency info + one another region's API.



Seoul
trafficlight
emergency
velocity

➕ Another Region

■ Expected Result

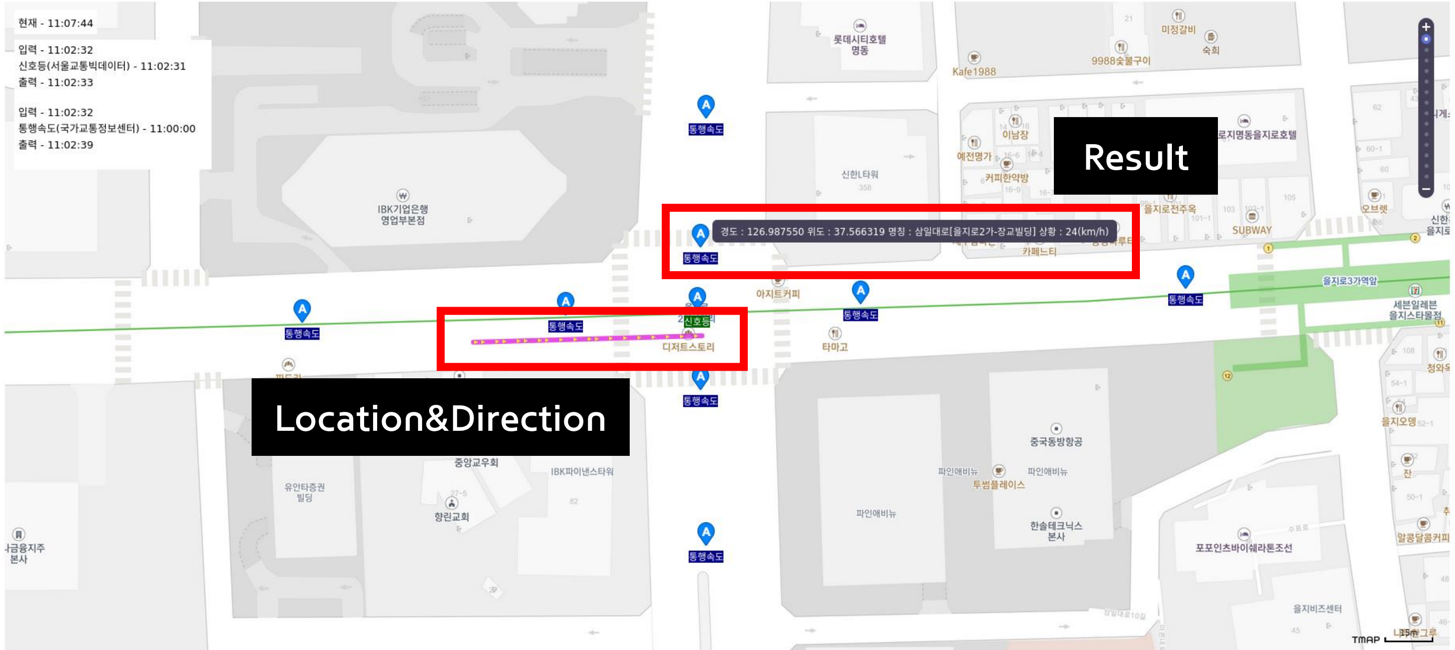


Server&Client

Client

Client

- ▶ Send the location&direction on the map to the server.
- ▶ Receive RSTI from the server.



```
현재 - 11:07:44

입력 - 11:02:32
신호등(서울교통빅데이터) - 11:02:31
출력 - 11:02:33

입력 - 11:02:32
통행속도(국가교통정보센터) - 11:00:00
출력 - 11:02:39
```

경도 : 126.987550 위도 : 37.566319 명칭 : 삼일대로[을지로2가-장교빌딩] 상황 : 24(km/h)

Result

Location&Direction

Server

■ Communicator

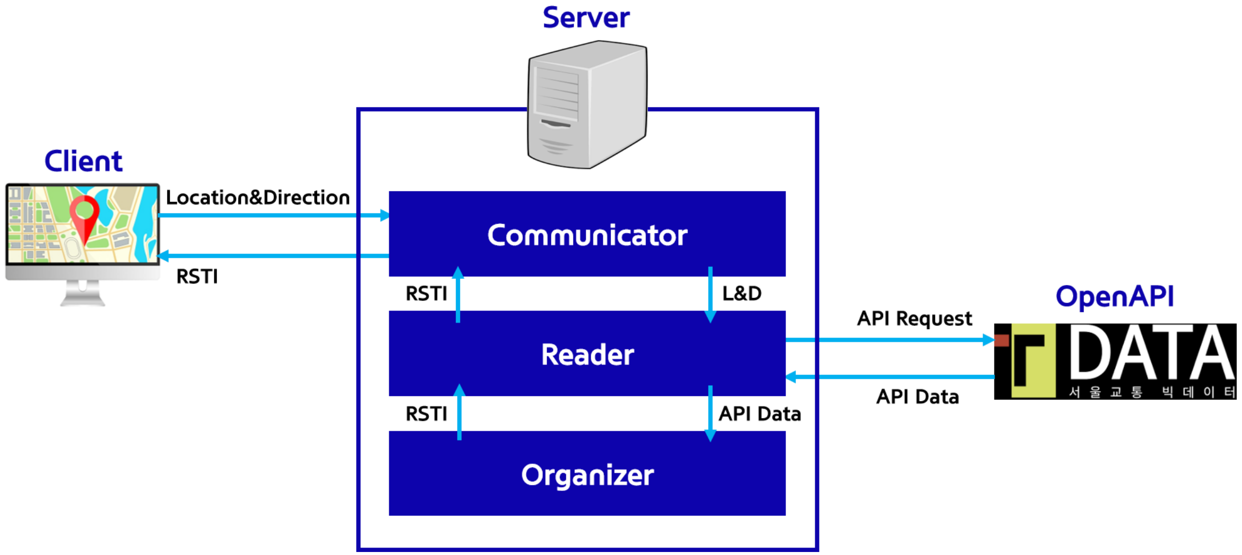
▶ Exchange messages with the client.

■ Reader

▶ Get OpenAPI data according to the location.

■ Organzier

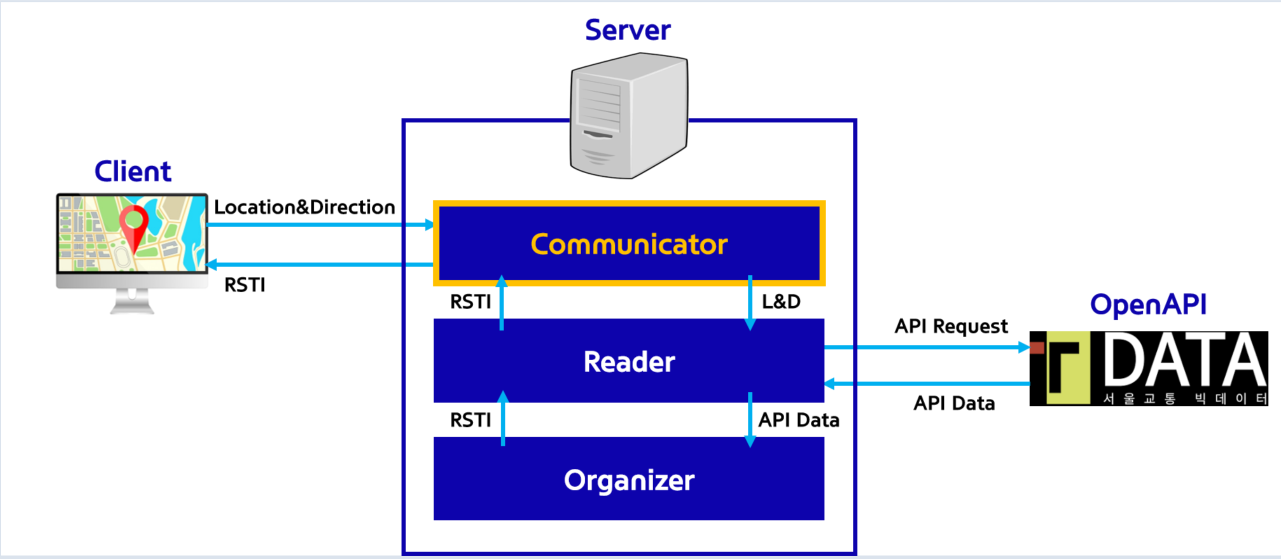
▶ Extract RSTI from APIdata.



Server

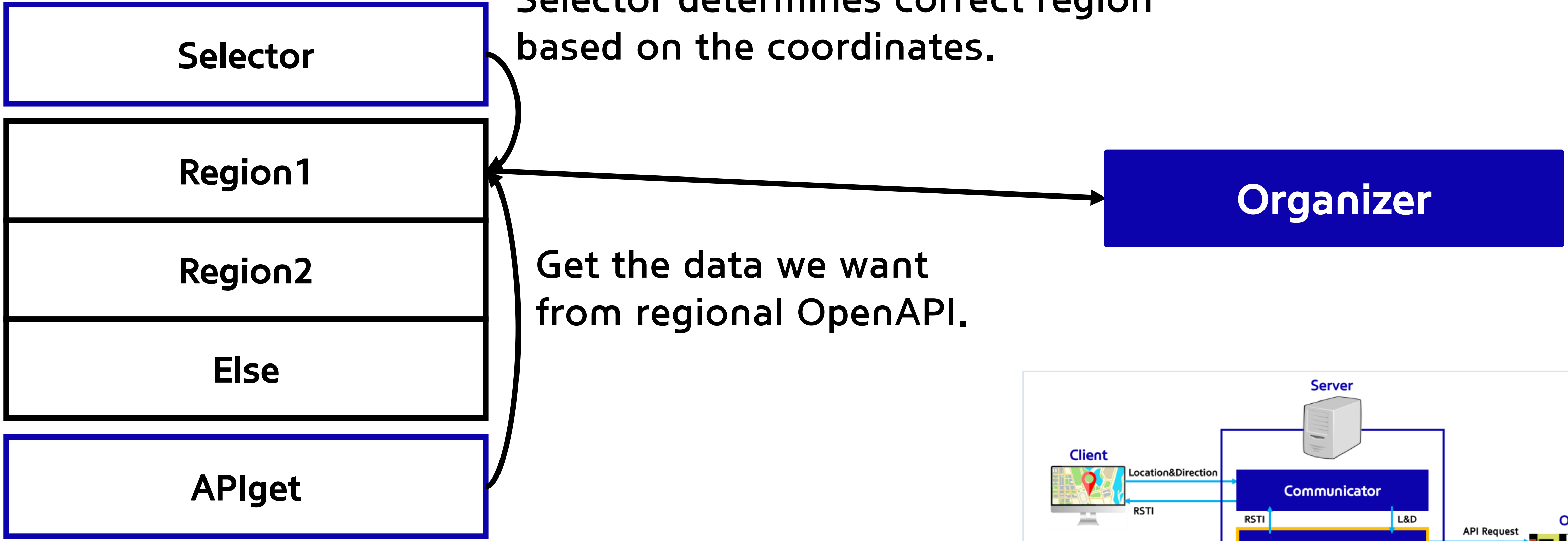
■ Communicator

- ▶ Exchange messages with the client.
- ▶ Extract coordinates and angle from message, and send to Reader.
- ▶ Communicator doesn’t have to be modified with API’s change.



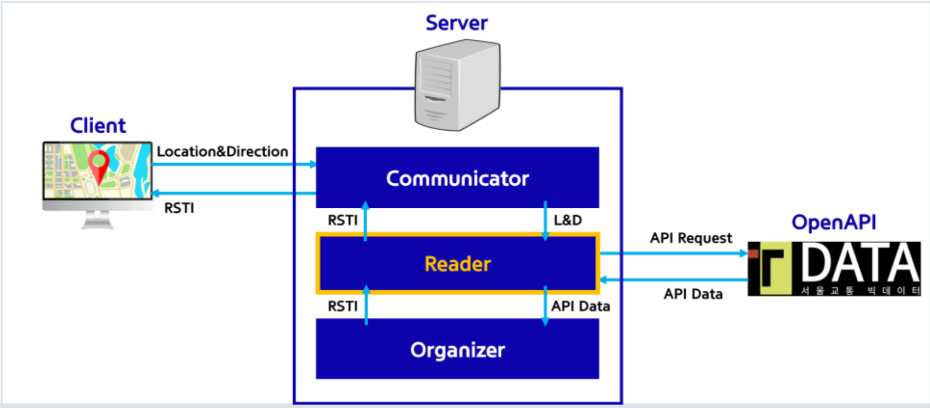
Server

Reader



Selector determines correct region based on the coordinates.

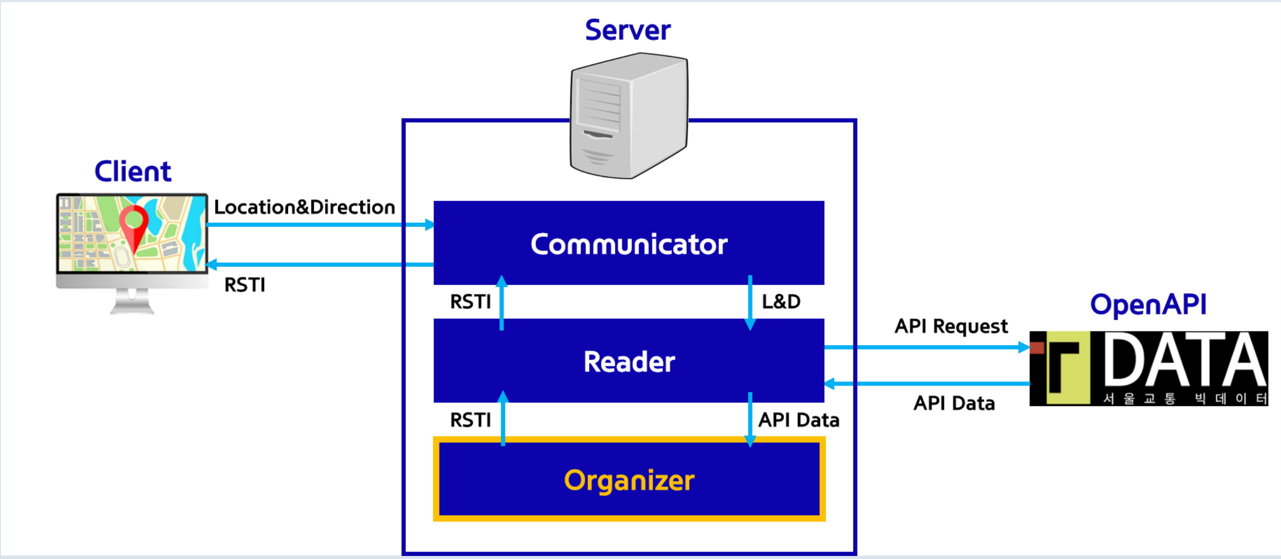
Get the data we want from regional OpenAPI.



Server

Organizer

- Receive API data from Reader.
- Extract RSTI from APIdata according to the location and direction.
- Return the result to the client.

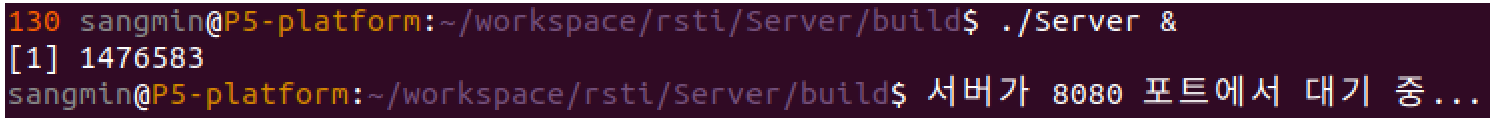


Verification

Verification

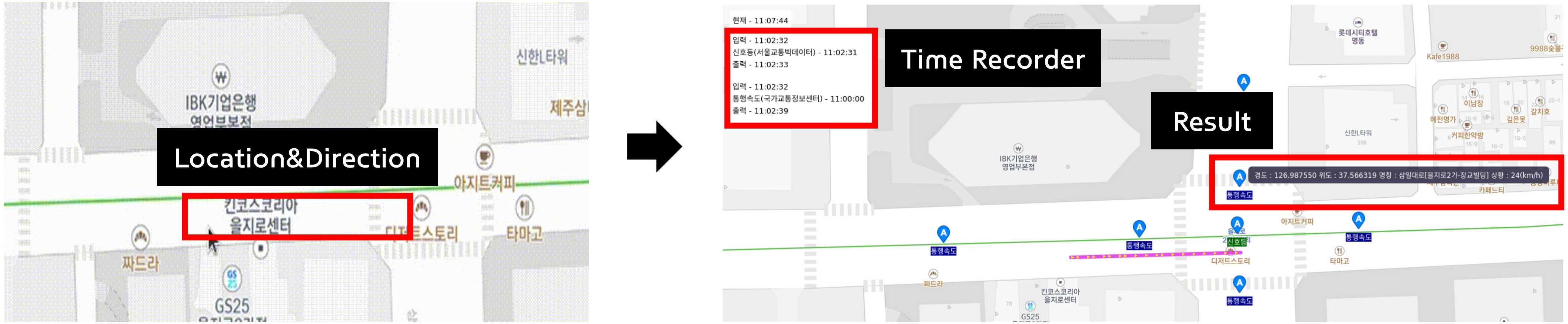
■ Implementation

- ▶ Run server on the team computer.



```
130 sangmin@P5-platform:~/workspace/rsti/Server/build$ ./Server &
[1] 1476583
sangmin@P5-platform:~/workspace/rsti/Server/build$ 서버가 8080 포트에서 대기 중...
```

- ▶ Run client and draw arrow indicating location and direction on the map
- ▶ Result are marked by marker on the map.
- ▶ The top left box shows the client’s in/out time and the RSTI’s generation time.

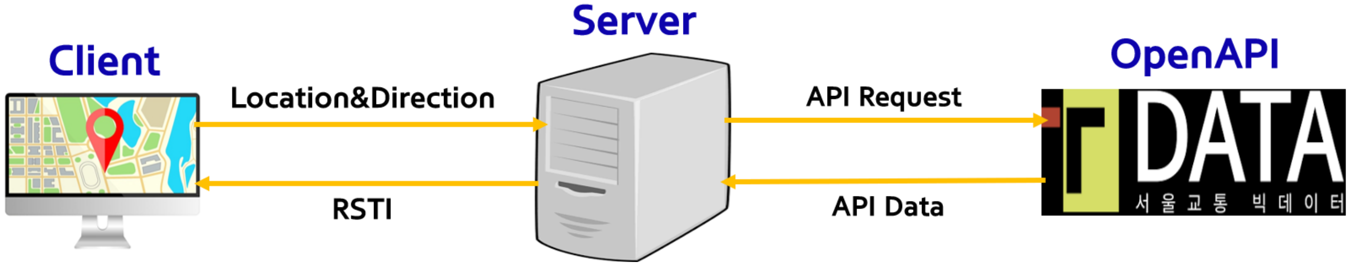


```
현재 - 11:07:44

입력 - 11:02:32
신호등(서울교통빅데이터) - 11:02:31
출력 - 11:02:33

입력 - 11:02:32
통행속도(국가교통정보센터) - 11:00:00
출력 - 11:02:39
```

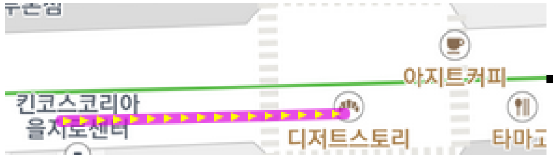
```
경도 : 126.987550 위도 : 37.566319 명칭 : 삼일대로[을지로2가-장교빌딩] 상황 : 24(km/h)
```

Verification

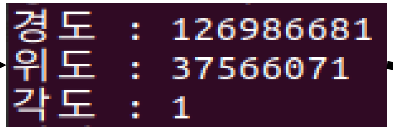
■ Structure



Arrow on the Seoul

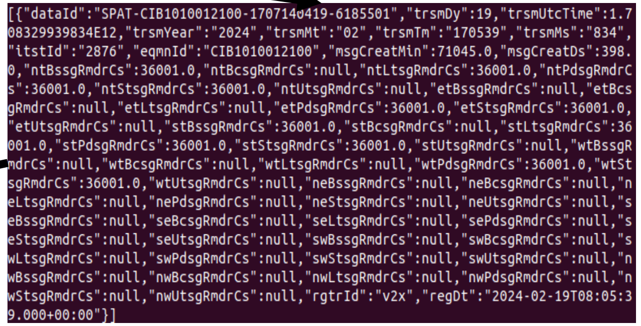


Coordinates/Angle



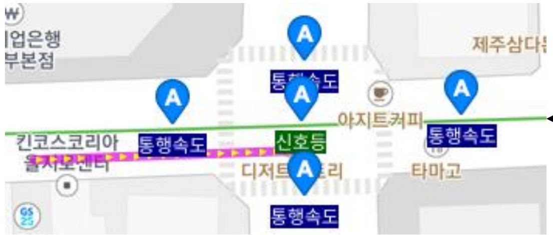
```
경도 : 126986681
위도 : 37566071
각도 : 1
```

Raw API data

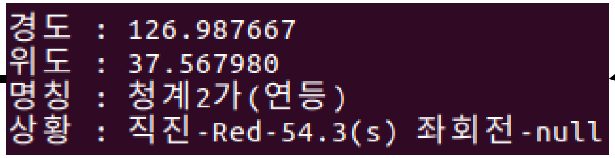


```
[{"dataId":"SPAT-CIB1010012100-1707140419-6185501","trsmDy":19,"trsmUtcTime":1.7
08329939834E12,"trsmYear":"2024","trsmMt":"02","trsmTm":"170539","trsmMs":"834",
"itstId":"2876","eqmnId":"CIB1010012100","msgCreatMin":71045.0,"msgCreatDs":398.
0,"ntBssgRmdrCs":36001.0,"ntBcsgRmdrCs":null,"ntLtsgRmdrCs":36001.0,"ntPdsgRmdrC
s":36001.0,"ntStsgRmdrCs":36001.0,"ntUtsgRmdrCs":null,"etBssgRmdrCs":null,"etBcs
gRmdrCs":null,"etLtsgRmdrCs":null,"etPdsgRmdrCs":36001.0,"etStsgRmdrCs":36001.0,
"etUtsgRmdrCs":null,"stBssgRmdrCs":36001.0,"stBcsgRmdrCs":null,"stLtsgRmdrCs":36
001.0,"stPdsgRmdrCs":36001.0,"stStsgRmdrCs":36001.0,"stUtsgRmdrCs":null,"wtBssgR
mdrCs":null,"wtBcsgRmdrCs":null,"wtLtsgRmdrCs":null,"wtPdsgRmdrCs":36001.0,"wtSt
sgRmdrCs":36001.0,"wtUtsgRmdrCs":null,"neBssgRmdrCs":null,"neBcsgRmdrCs":null,"n
eLtsgRmdrCs":null,"nePdsgRmdrCs":null,"neStsgRmdrCs":null,"neUtsgRmdrCs":null,"s
eBssgRmdrCs":null,"seBcsgRmdrCs":null,"seLtsgRmdrCs":null,"sePdsgRmdrCs":null,"s
eStsgRmdrCs":null,"seUtsgRmdrCs":null,"swBssgRmdrCs":null,"swBcsgRmdrCs":null,"s
wLtsgRmdrCs":null,"swPdsgRmdrCs":null,"swStsgRmdrCs":null,"swUtsgRmdrCs":null,"n
wBssgRmdrCs":null,"nwBcsgRmdrCs":null,"nwLtsgRmdrCs":null,"nwPdsgRmdrCs":null,"n
wStsgRmdrCs":null,"nwUtsgRmdrCs":null,"rgtrId":"v2x","regDt":"2024-02-19T08:05:3
9.000+00:00"}]
```

Result



RSTI



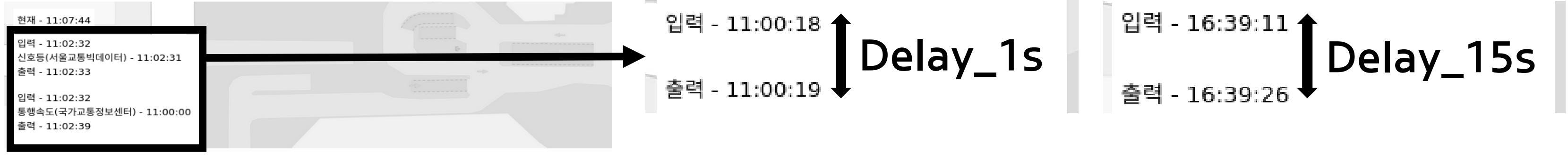
```
경도 : 126.987667
위도 : 37.567980
명칭 : 청계2가(연등)
상황 : 직진-Red-54.3(s) 좌회전-null
```

Verification

■ Delay

▶ Delay Test

- It showed random delay from 1s to 15s.
- National Transport Info Center’s OpenAPI related to velocity and emregency is unstable.
- Since three types of data were returned at once, trafficlight data was also returned late.



현재 - 11:07:44

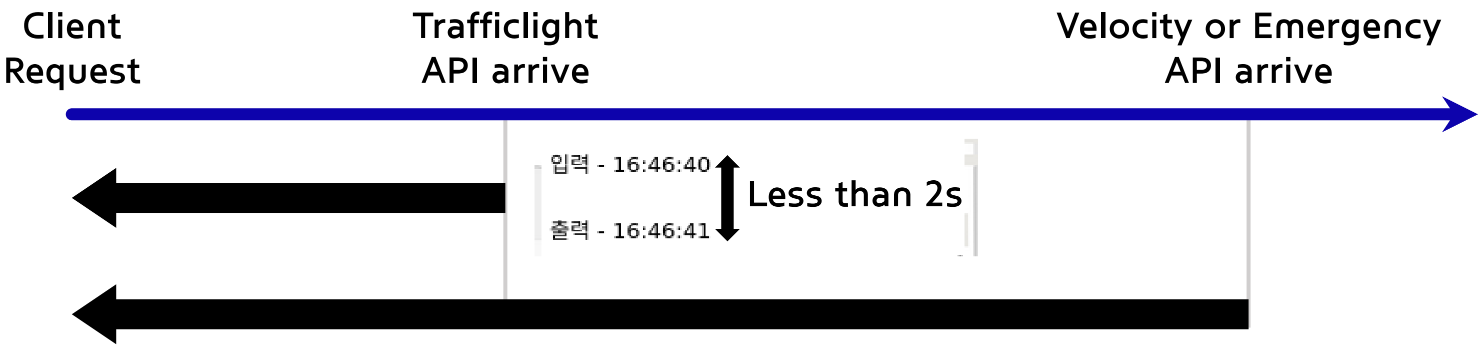
입력 - 11:02:32
신호등(서울교통빅데이터) - 11:02:31
출력 - 11:02:33

입력 - 11:02:32
통행속도(국가교통정보센터) - 11:00:00
출력 - 11:02:39

입력 - 11:00:18
출력 - 11:00:19
Delay_1s

입력 - 16:39:11
출력 - 16:39:26
Delay_15s

▶ Solve Problem

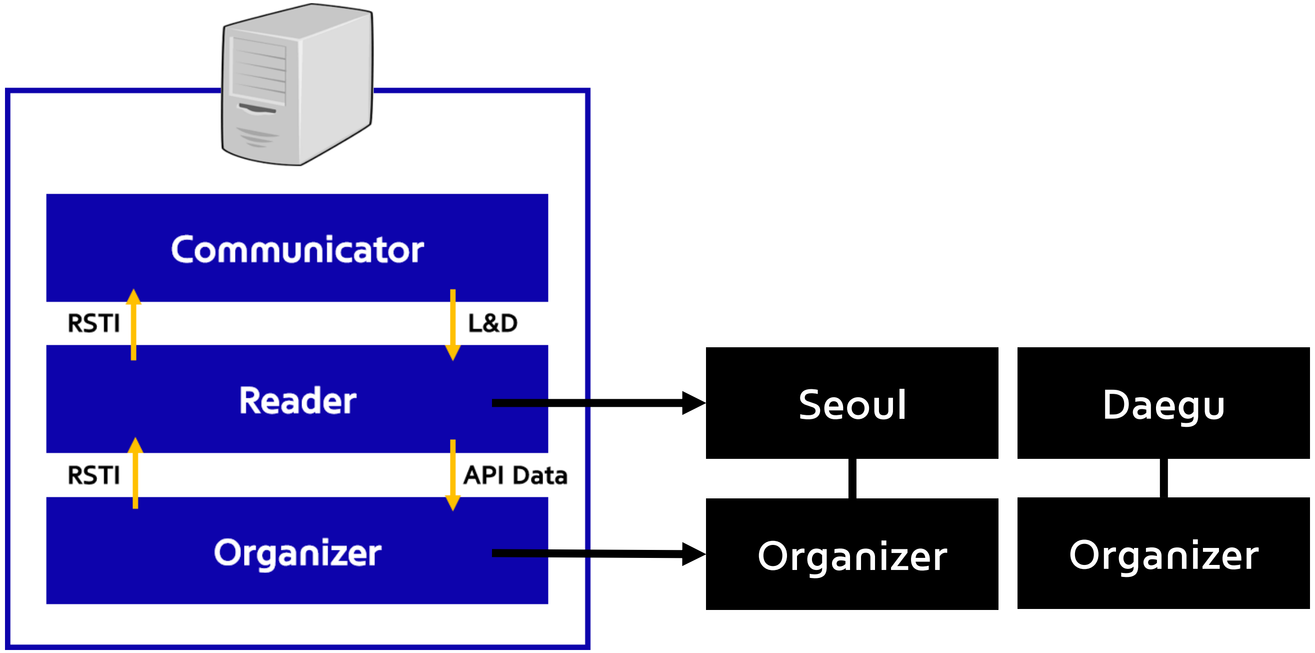


입력 - 16:46:40
출력 - 16:46:41
Less than 2s

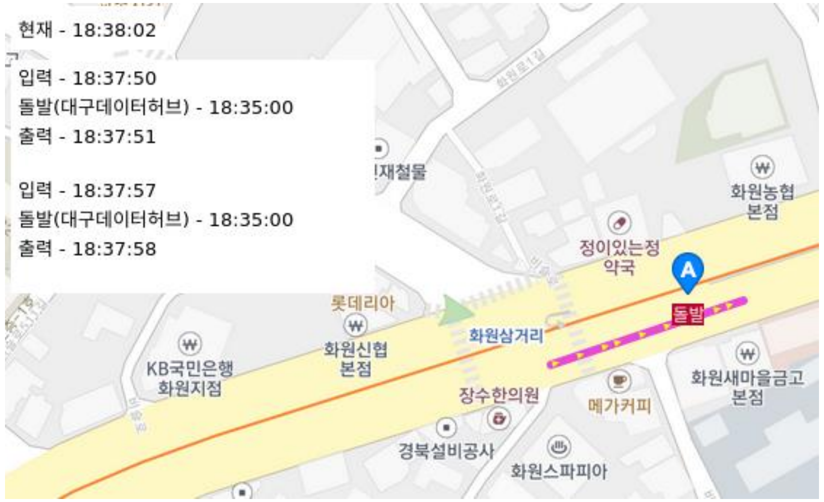
Verification

■ Expansion

- ▶ To verify region-expansion, add emergency API of Daegu area
- ▶ Simple expansion is possible by adding the code to the reader that brings up the Daegu API and creating an organizer module connected to it.



Test



현재 - 18:38:02

입력 - 18:37:50
돌발(대구데이터허브) - 18:35:00
출력 - 18:37:51

입력 - 18:37:57
돌발(대구데이터허브) - 18:35:00
출력 - 18:37:58

Verification

API Request Limit

OpenAPI	Seoul Transport Bigdata Platform	National Transport Information Center	Daegu Data Hub
Basic Account	1000 per day	3000 per day	10000 per day
Advanced Account	100000 per day	3000 per day	1000000 per day
Unlimited Account	Unlimit	Unlimit	Unlimit
How to Upgrade	Actual operation on the car is required		

Conclusion

Conclusion

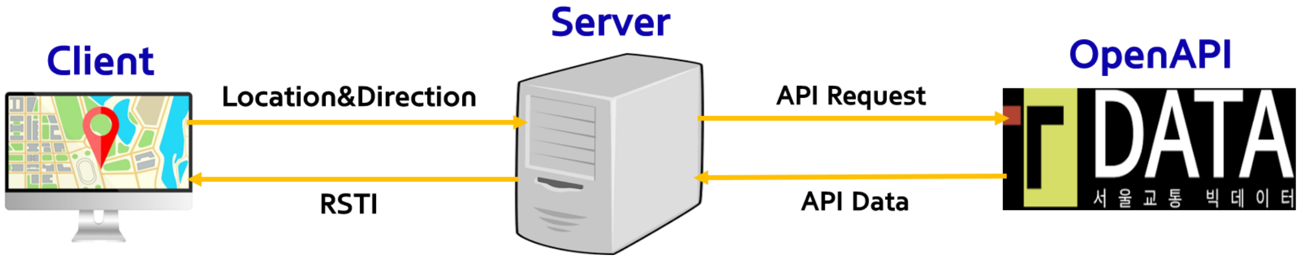
■ Conclusion

▶ Server to provide RSTI from OpenAPI to Client.

- Validation completed for targeted data flows.
- Region expansion is possible through simple code modification of reader and organzier.
- Delay and request limit show that OpenAPI is main cause of influence on the program.

▶ Documentation of how can solve the various situations from OpenAPI.

- The way to reduce api delay by adding spatial database.
- The way to increase request limit by applying for the account upgrade.

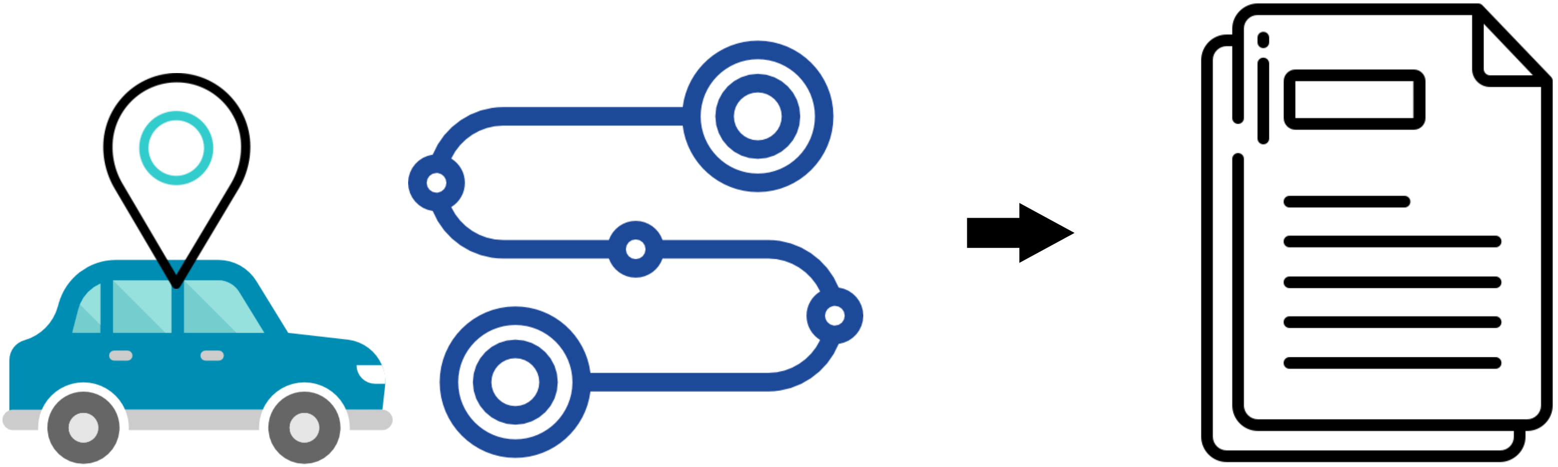


Conclusion

■ Utilization

▶ Apply the program to actual autonomous driving beyond prototype.

- Connect the GPS and direction indicator of vehicle to client, instead of the virtual value.
- In connection with the route of the autonomous vehicle, the velocity and emergency data of the expected locations can be updated in advance.
- Other information for the operation on the real car.



Q&A

Your partner for the Future Mobility

THANK YOU

CONTROLWORKS

Appendix

■ API List

▶Traffic-light API

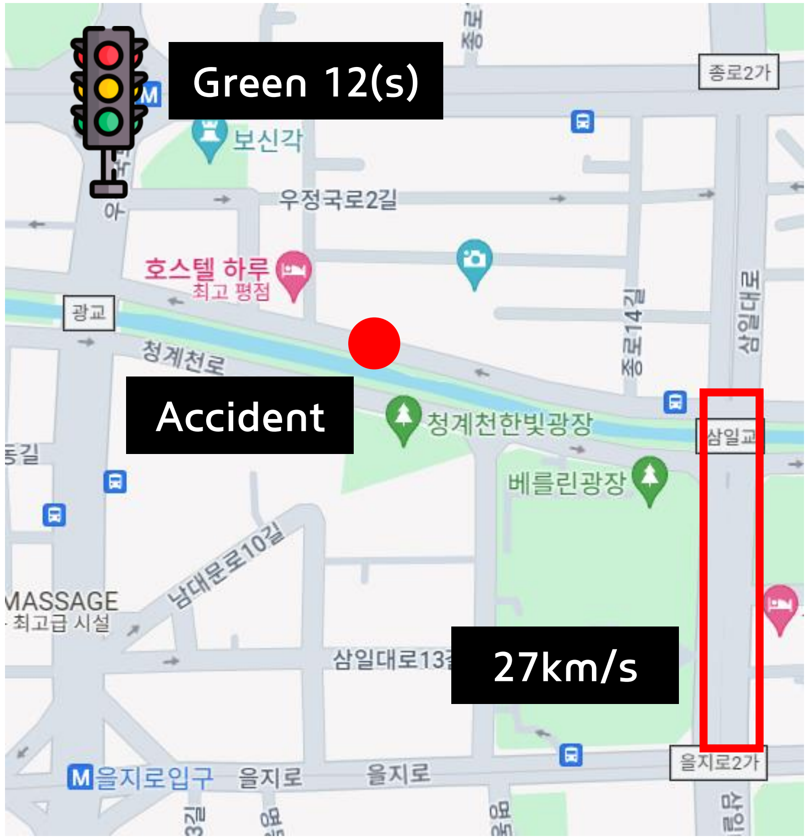
- ● Signal/Remaining-Time

▶Velocity API

- ● Traffic Flow Speed

▶Emergency API

- ● Accident/Construction



Appendix

Seoul_Organzier - Trafficlight

▶ Seoul Intersection ID/Coordinates API

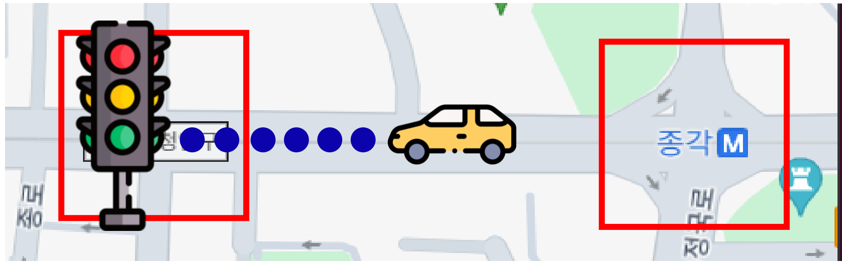
- Find the front intersection of the client.
  - Make 40m*40m square on the intersections
  - Draw 200 dots in front of the client using unit vector
  - Determine whether the dot is in the square

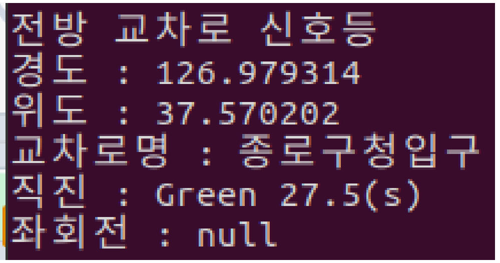
▶ Intersection Light Signal API

- Get the front intersection’s light signal

▶ Intersection Light Remaining-Time API

- Get the front intersection’s light remaining-time



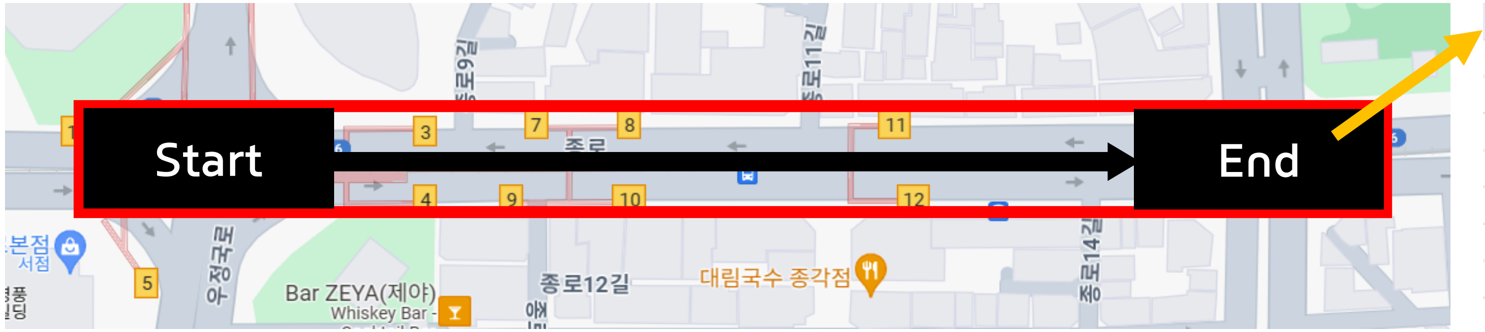


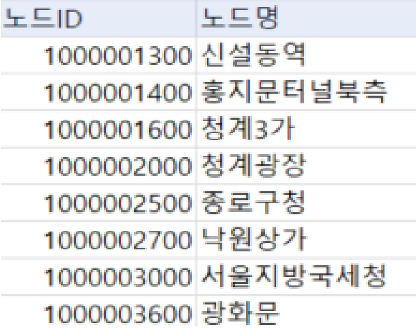
```
전방 교차로 신호등
경도 : 126.979314
위도 : 37.570202
교차로명 : 종로구청입구
직진 : Green 27.5(s)
좌회전 : null
```

Appendix

■ Seoul_Organzier - Velocity

1. Road Traffic Flow Speed API

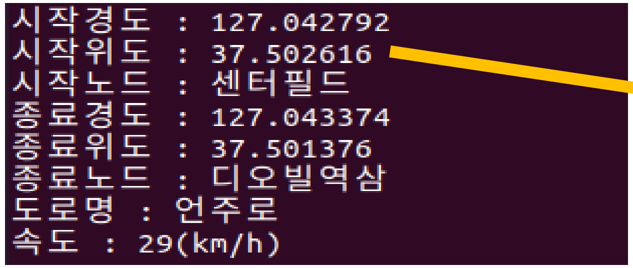




노드ID	노드명
1000001300	신설동역
1000001400	홍지문터널북측
1000001600	청계3가
1000002000	청계광장
1000002500	종로구청
1000002700	낙원상가
1000003000	서울지방국세청
1000003600	광화문

Cannot find exact location with NodeID

2. Seoul NodeID to Coordinates API



```
시작경도 : 127.042792
시작위도 : 37.502616
시작노드 : 센터필드
종료경도 : 127.043374
종료위도 : 37.501376
종료노드 : 디오빌역삼
도로명 : 언주로
속도 : 29(km/h)
```

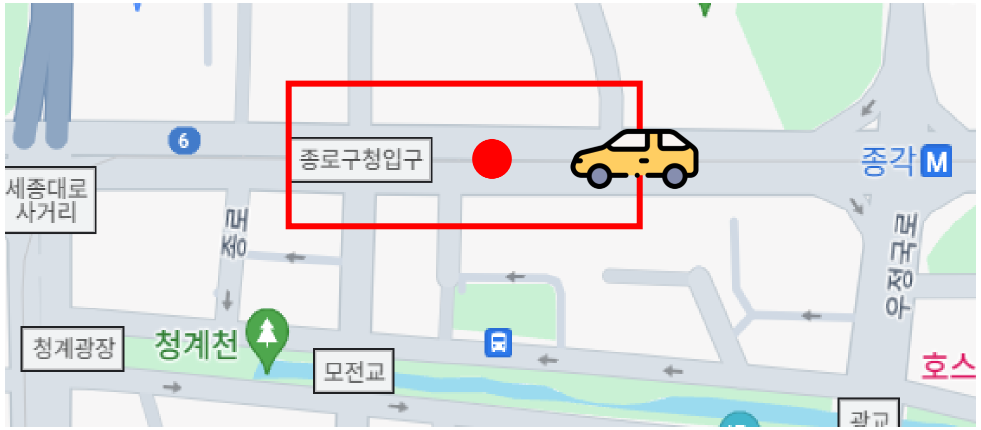
NodeID → Coordinates

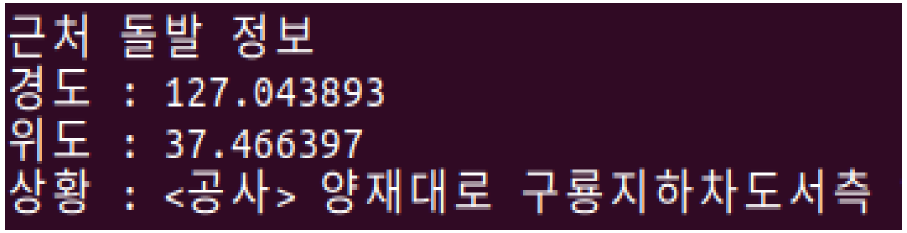
Appendix

■ Seoul_Organzier - Emergency

▶ Transport Emergency API

- Get the emergency data of the front range.



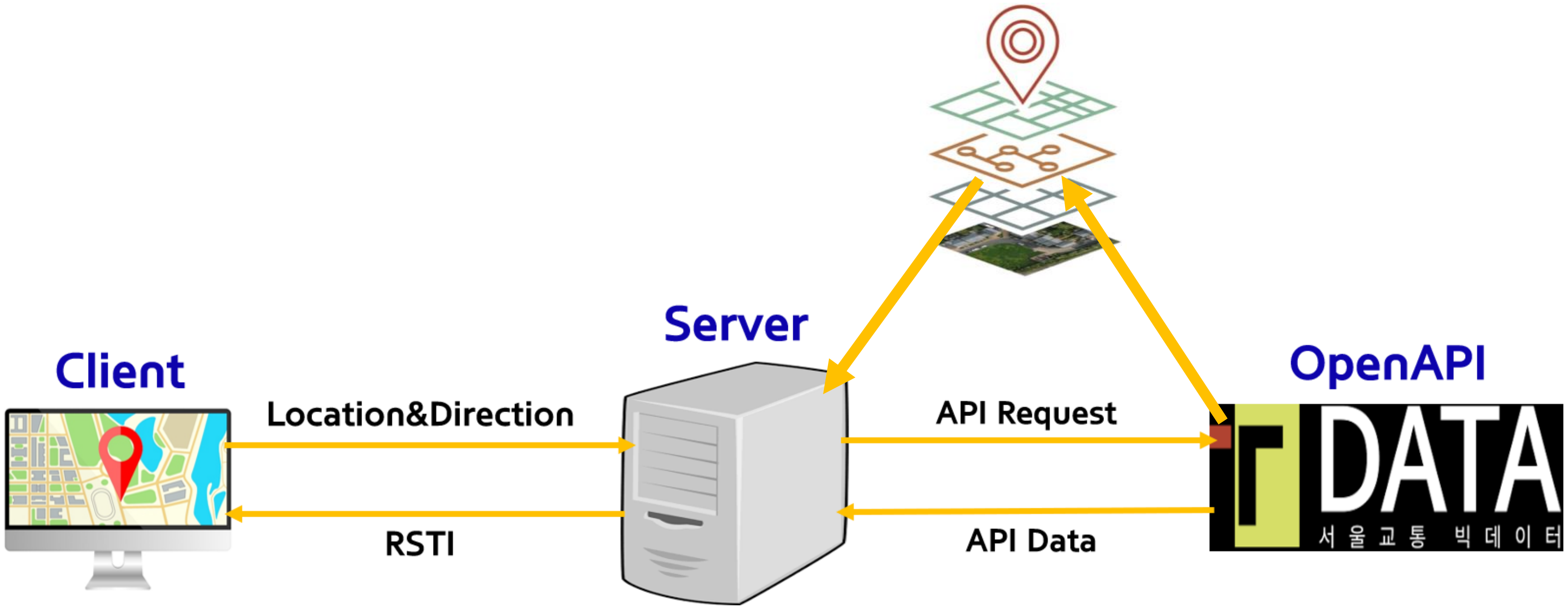


```
근처 돌발 정보
경도 : 127.043893
위도 : 37.466397
상황 : <공사> 양재대로 구룡지하차도서측
```

Appendix

■ Reduce API delay by Spatial Database

- ▶ Save OpenAPI data in advance, and server can fetch it quickly.
- ▶ Provide exact layout of the land.



Server
Client
Location&Direction
RSTI
OpenAPI
API Request
API Data
DATA
서울교통 빅데이터