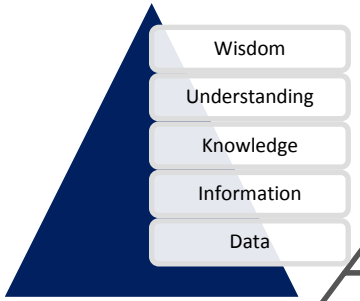
SAVI Design Camp – Theme: Smart Cities

CVST Data - APIs

Ali Tizghadam
Department of Electrical & Computer Engineering
University of Toronto

CVST Hierarchy of Needs

Connected Vehicles & Smart Transportation



Applications

- Transportation applications
- Recommendations (Routing, City planning)
- What-If Analysis, Impact analysis

Intelligent Transportation Management

(Data Process)

- Real-time streaming analytics
- OD Demands
- Trending / Forecasting
- KPI analysis

- In Memory Processing
- Trade between cost, quality and time

Data Dissemination

(Data Management)

- Publish / subscribe
- Data collection
- Data Integrity, cleansing
- Data anonymity

- Share data efficiently

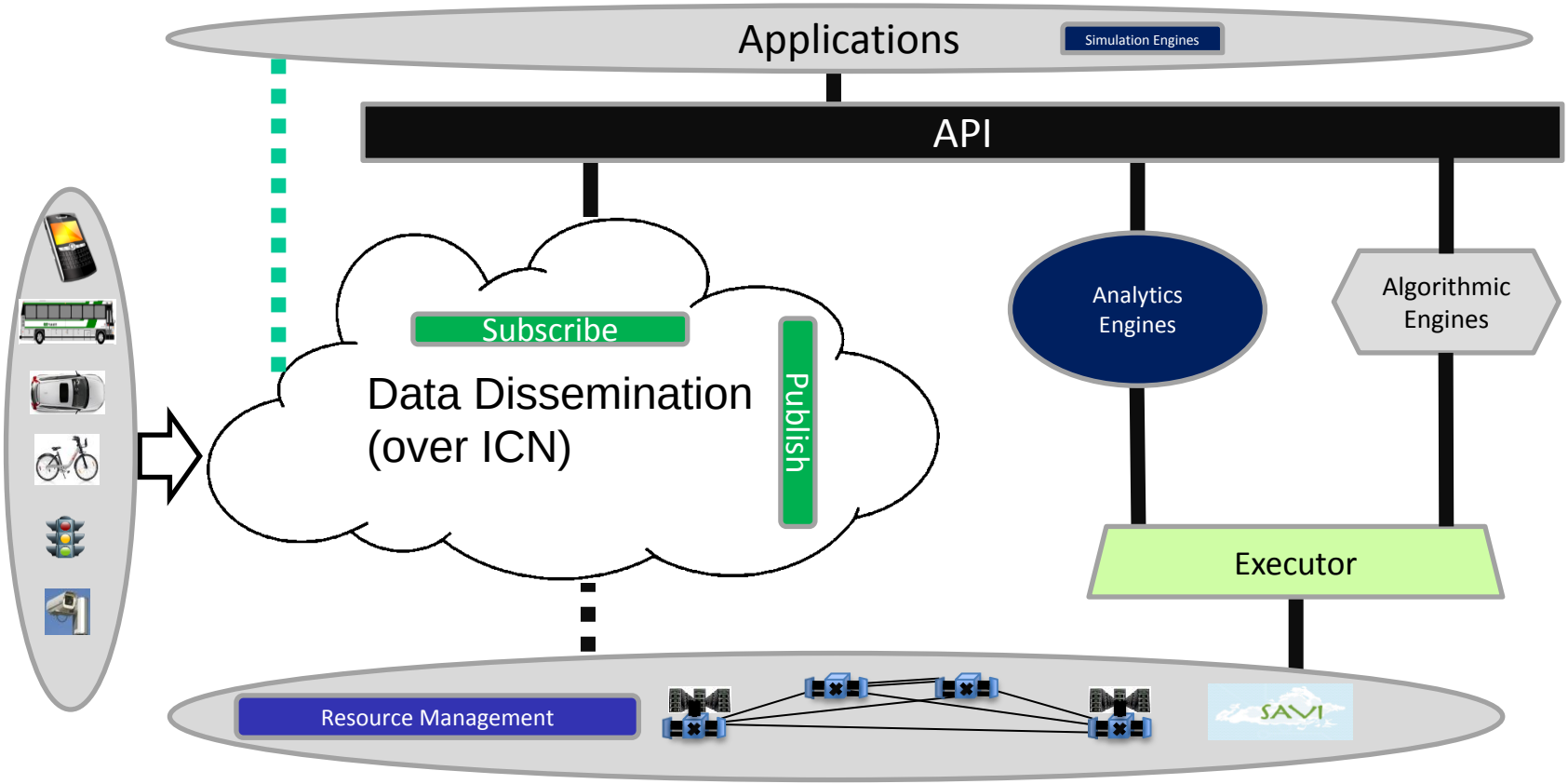
Connected Vehicles & Mobile Computing Cloud

(Resource Management)

- Sensors
- V2V, V2R, V2I
- Cloud resource management
  - Horizontal / vertical scaling (SAVI)

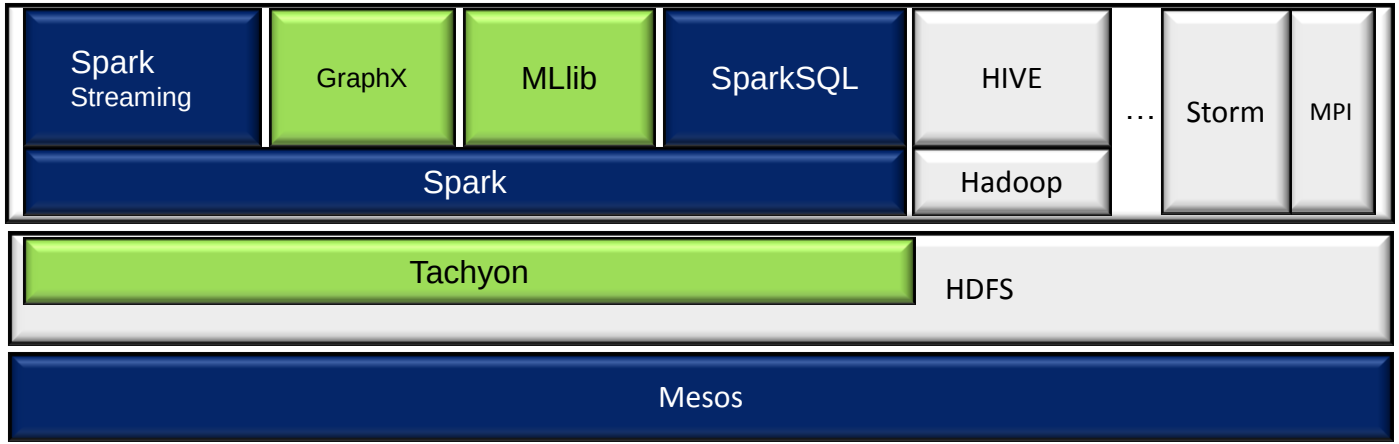
- Share resources efficiently

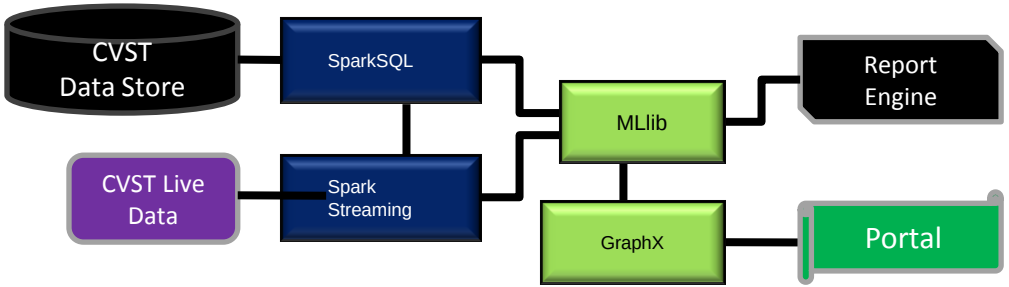
CVST Autonomic Loop

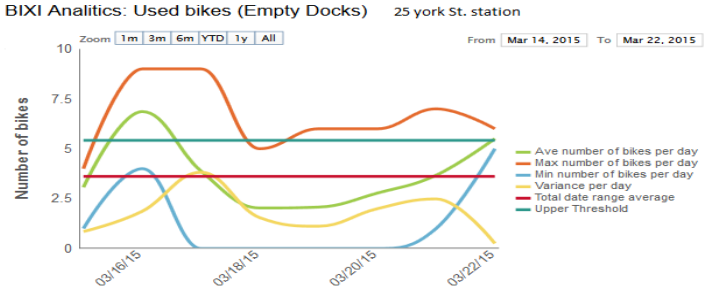


Analytics on CVST

- Multiple Analytics engines are running on CVST







CVST APIs

- General rule: all data (raw, processed, analyzed) available through APIs in CVST platform
- Any type of data inquiry from system is through APIs
- APIs have access level hierarchy
- Pull-based as well as push-based
  - This is enabled by publish/subscribe paradigm
- Hierarchy of APIs along with analytics-as-a-service create information, knowledge, understanding, wisdom
- Can build a strong business model

API Examples

spec : CVST API User Instruction

Show/Hide | List Operations | Expand Operations | Raw

GET	/ttc	
GET	/ttc/name	Returns all availble route name for searching
GET	/ttc/tag	Returns all availble routetag for searching
GET	/ttc/id	Returns all availble vehicle_id for searching
GET	/ttc/{vehicle_id}	returns last hour updated for the stationID or
GET	/ttc/tag/{routeNumber}	return the routeTag data between
GET	/ttc/name/{route_name}	return the route_name data between
GET	/ttc/routes	Returns the all data of routesoute
GET	/ttc/routes/tag/{routeTag}	Return the data for a specific routeTag
GET	/ttc/routes/name/{title}	Return the data for a specific route name

[BASE URL: http://portal.cvst.ca/api/0.1/spec/_/resource_list.json , API VERSION: 0.1]

portal.cvst.ca/api/0.1/bixi/?date=realtime


```
{
    "bikes": 0,
    "coordinates": [
        -79.37617,
        43.66207
    ],
    "date_time": "Thu, 03 Sep 2015 16:10:01 -0000",
    "empty_docks": 13,
    "installed": true,
    "last_seen": "Thu, 03 Sep 2015 16:01:16 -0000",
    "last_update": "Thu, 03 Sep 2015 16:01:14 -0000",
    "station_id": 1,
    "station_name": "Jarvis St / Carlton St",
    "terminalName": "7055",
    "timestamp": 1441296601
},
{
    "bikes": 0,
    "coordinates": [
        -79.3806,
        43.6636
    ],
    "date_time": "Thu, 03 Sep 2015 16:10:01 -0000",
    "empty_docks": 6,
    "installed": true,
    "last_seen": "Thu, 03 Sep 2015 16:03:45 -0000",
    "last_update": "Thu, 03 Sep 2015 15:49:40 -0000",
    "station_id": 5,
    "station_name": "Church St / Alexander St",
    "terminalName": "7044",
    "timestamp": 1441296601
},
```

BiXi APIs

API Function	Syntax	Remarks
Latest update	http://portal.cvst.ca/api/0.1/bixi	
Records for specific time	http://portal.cvst.ca/api/0.1/bixi?timestamp=<ts>	<ts> could be epoch time or string type with <YYYY><MM><DD>T<HH><MM><time-zone>
Records for a specific station	http://portal.cvst.ca/api/0.1/bixi/<station_id>	
Records for an id within a period of time	http://portal.cvst.ca/api/0.1/bixi/<station_id>?starttime=<ts1>&endtime=<ts2>	
Records in Geojson format	http://portal.cvst.ca/api/0.1/bixi/geojson	

TTC APIs

API Function	Syntax
Latest update	http://portal.cvst.ca/api/0.1/ttc
Records for specific time	http://portal.cvst.ca/api/0.1/ttc?timestamp=<ts>
Route names	http://portal.cvst.ca/api/0.1/ttc/name
Records for specific route name	http://portal.cvst.ca/api/0.1/ttc/name/<route_name>
Records for specific route in a given period	http://portal.cvst.ca/api/0.1/ttc/name/<route_name> ?starttime=<ts1>&endtime=<ts2>
TTC Routes with tags	http://portal.cvst.ca/api/0.1/ttc/routes
Records per route tag	http://portal.cvst.ca/api/0.1/ttc/tag/<tg>

Highway Speed APIs

API Function	Syntax
Latest update	http://portal.cvst.ca/api/0.1/HW_speed
Record for a given time	http://portal.cvst.ca/api/0.1/HW_speed?timestamp=<ts>
Records average speed	http://portal.cvst.ca/api/0.1/HW_speed/AVG
Record for a given time	http://portal.cvst.ca/api/0.1/HW_speed/AVG?timestamp=<ts>
Road Spec and Id	http://portal.cvst.ca/api/0.1/HW_speed/street_name
Records for a specific sensor location id	http://portal.cvst.ca/api/0.1/HW_speed/<loc_id>
Records for a sensor within period of time	http://portal.cvst.ca/api/0.1/HW_speed/<loc_id>?starttime=<ts1> &endtime=<ts2>

Border APIs

API Function	Syntax
Latest update	http://portal.cvst.ca/api/0.1/border
Records for specific time	http://portal.cvst.ca/api/0.1/border?timestamp=<ts>
Records for a specific port id	http://portal.cvst.ca/api/0.1/border/<port_id>
Records per id per timestamp	http://portal.cvst.ca/api/0.1/border/<port_id>?timestamp=<ts>
Records per id for a period	http://portal.cvst.ca/api/0.1/border/<port_id>?starttime=<ts1>&endtime=<ts2>

Loop-Detector APIs (Old)

API Function	Syntax
Latest update MTO	http://portal.cvst.ca/api/v1.0/realtime/MTO_LOOP_DETECTORS
Latest update COT	http://portal.cvst.ca/api/v1.0/realtime/COT_LOOP_DETECTORS
Records for a specific time MTO	http://portal.cvst.ca/api/v1.0/history/<yyyy-mm-dd-hh-mm/MTO_LOOP_DETECTORS Example: http://portal.cvst.ca/api/v1.0/history/2015-03-28-10-15/MTO_LOOP_DETECTORS
Records for a specific time COT	http://portal.cvst.ca/api/v1.0/history/<yyyy-mm-dd-hh-mm/COT_LOOP_DETECTORS

Air Pollutants APIs

API Function	Syntax
Latest update	http://portal.cvst.ca/api/0.1/airsense
Records for specific time	http://portal.cvst.ca/api/0.1/airsense?timestamp=<ts>
Records for a period of time	http://portal.cvst.ca/api/0.1/airsense?starttime=<ts1>&endtime=<ts2>
Records for a specific id	http://portal.cvst.ca/api/0.1/airsense/<device_id>

Twitter APIs

API Function	Syntax	
Latest update	http://portal.cvst.ca/api/0.1/twitter	
Records for specific time	http://portal.cvst.ca/api/0.1/?timestamp=<ts>	Only epoch for now
Records for a period of time	http://portal.cvst.ca/api/0.1/twitter?starttime=<ts1> &endtime=<ts2>	Only epoch for now
Records for a tweet id	http://portal.cvst.ca/api/0.1/twitter/tweet_id=<id>	

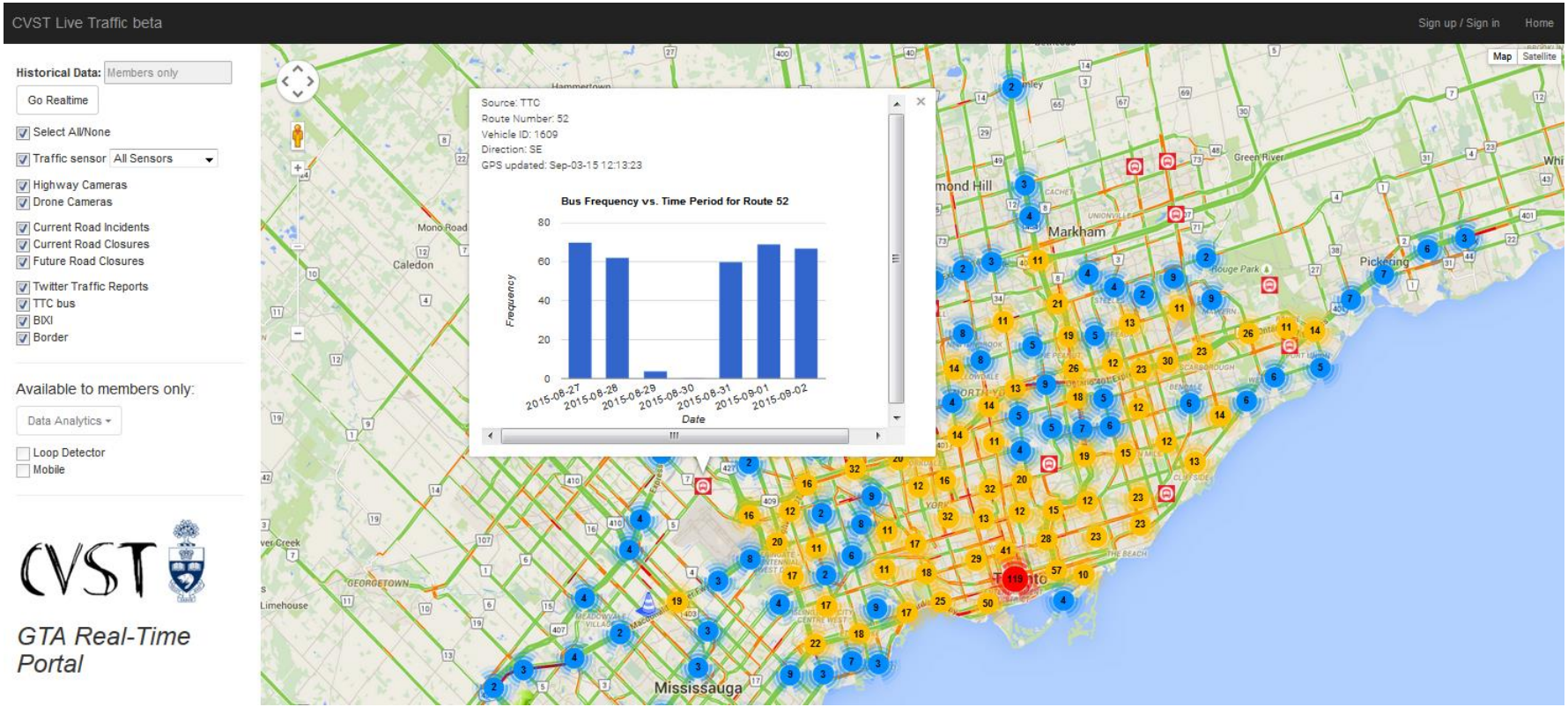
There is along list of twitter APIs for filtering through specific locations and specific type of tweets (such as incident, collision, etc). Please refer to the example sheet for details

Design Idea Suggestions

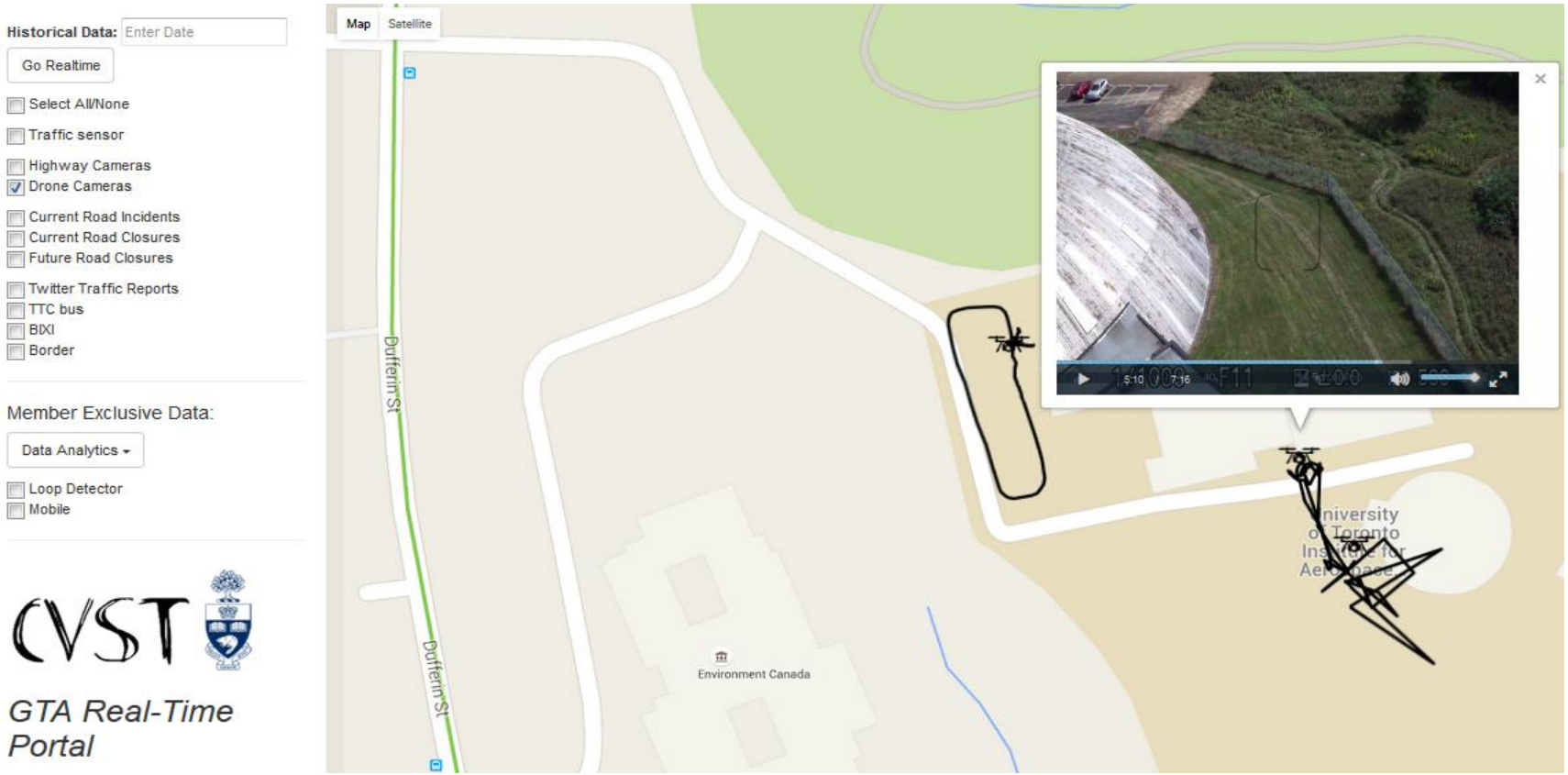
- Developing a Publish / Subscribe system to support push-based API
  - Using FIWARE ORION or any other pub/sub package
- Analytics on offered data
- Analytics APIs
- Innovative mobile app using CVST Data

Thank You!

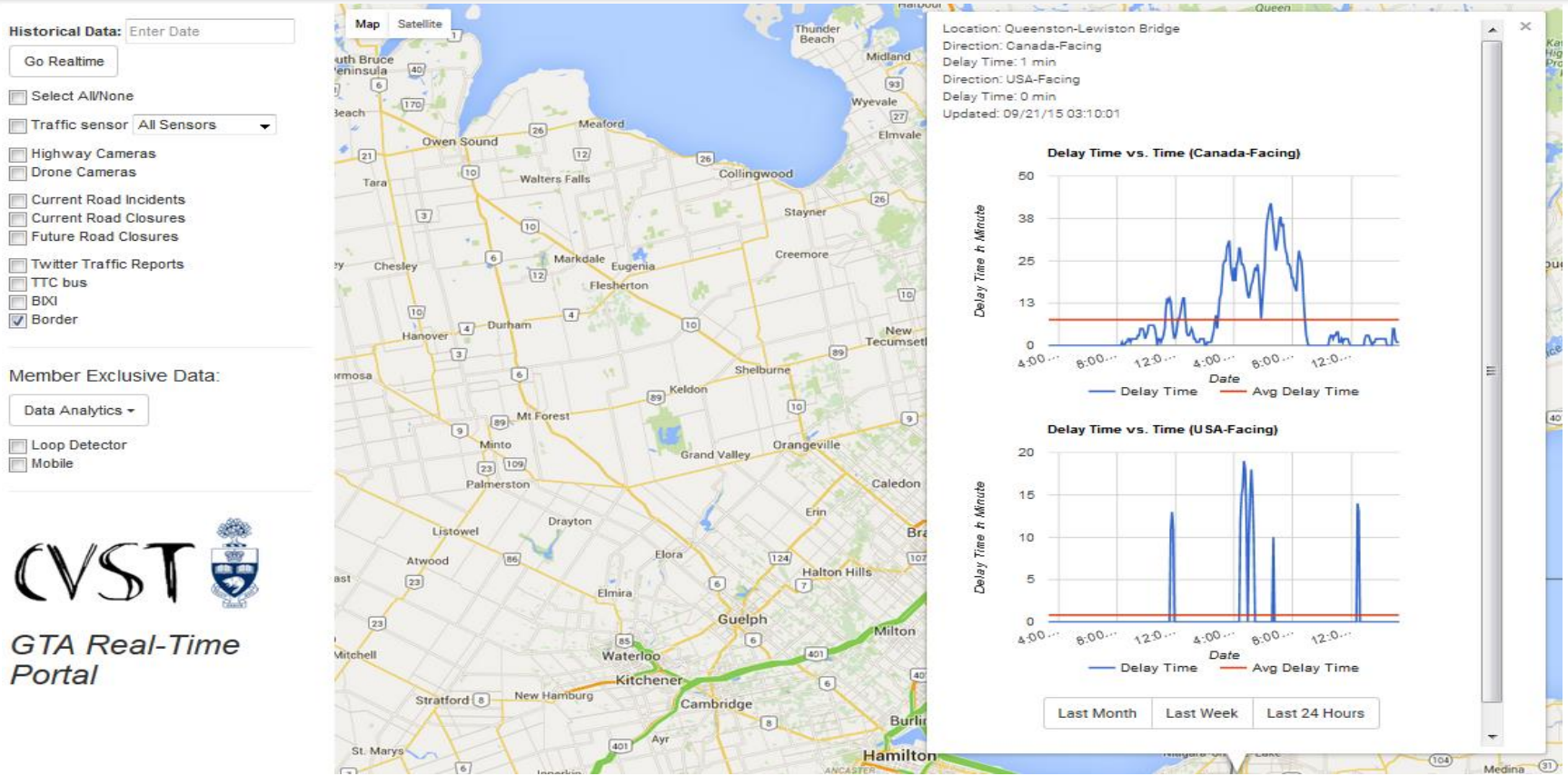
CVST Portal


CVST Live Traffic beta
Sign up / Sign in
Home
Historical Data: Members only
Go Realtime
Select All/None
Traffic sensor All Sensors
Highway Cameras
Drone Cameras
Current Road Incidents
Current Road Closures
Future Road Closures
Twitter Traffic Reports
TTC bus
BIXI
Border
Available to members only:
Data Analytics
Loop Detector
Mobile
GTA Real-Time Portal
Source: TTC
Route Number: 52
Vehicle ID: 1609
Direction: SE
GPS updated: Sep-03-15 12:13:23
Bus Frequency vs. Time Period for Route 52
Frequency
Date
2015-08-27
2015-08-28
2015-08-29
2015-08-30
2015-08-31
2015-09-01
2015-09-02
Map
Satellite
Markham
Pickering
Mississauga
Caledon
Toronto

Drone on CVST

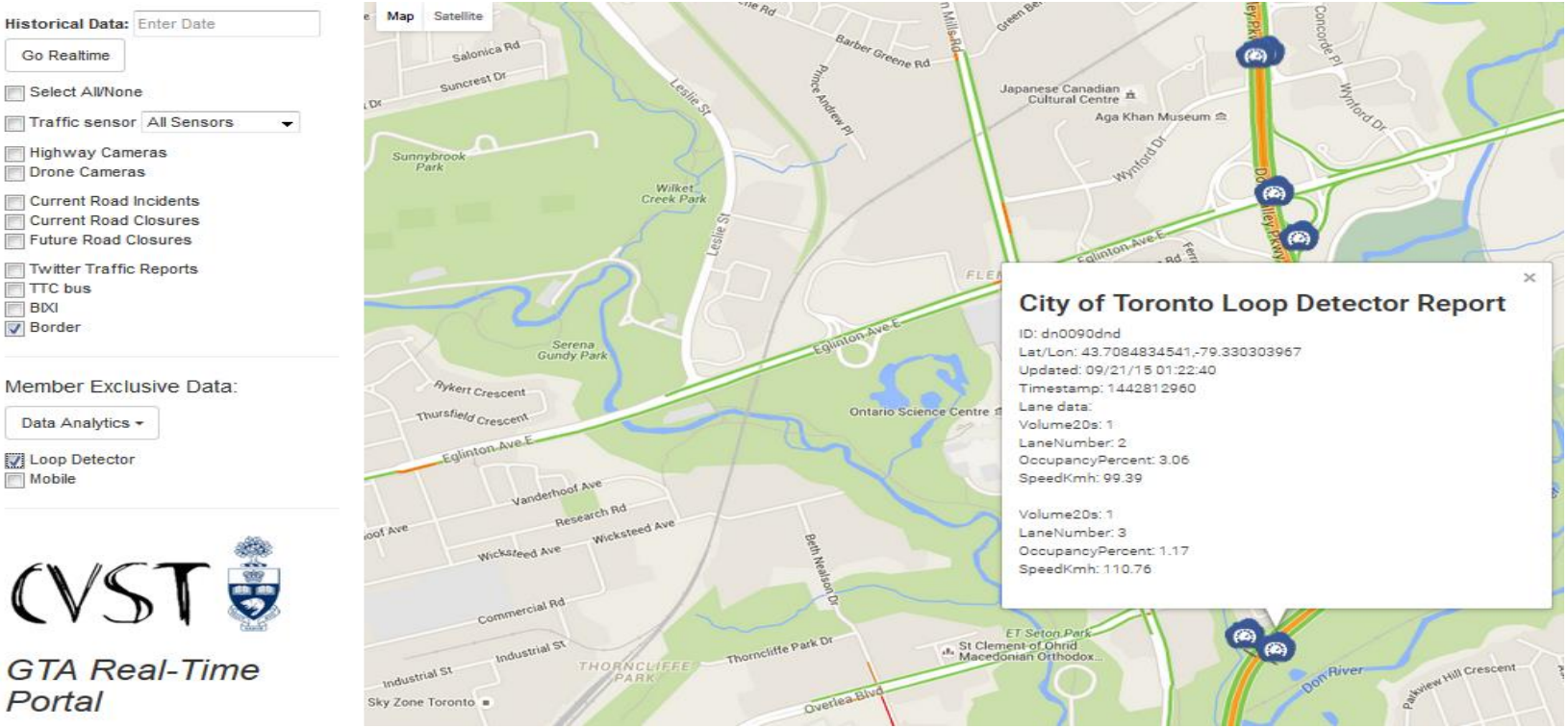


Example: Border Data on CVST



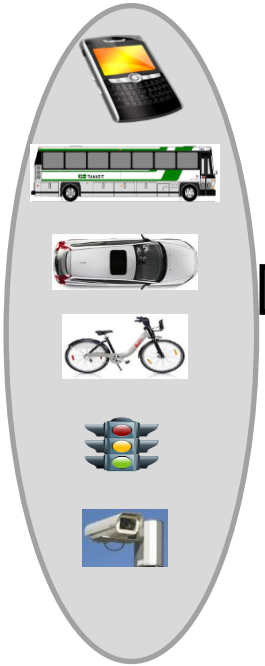
Example: Loop Detector Data

- Per lane statistics



CVST Platform Functional Diagram

Data Dissemination layer is consistent with GB979



Data Dissemination (over ICN)

Data Anonymization

Data Format

Data Cleansing

No

Data Validation

Yes

Publisher Client

Subscriber Client

Applications

API

Pub/Sub over ICN

Congestion Pricing

Routing

Alg. Engine

Analytics Engine

Simulation

Report Engine

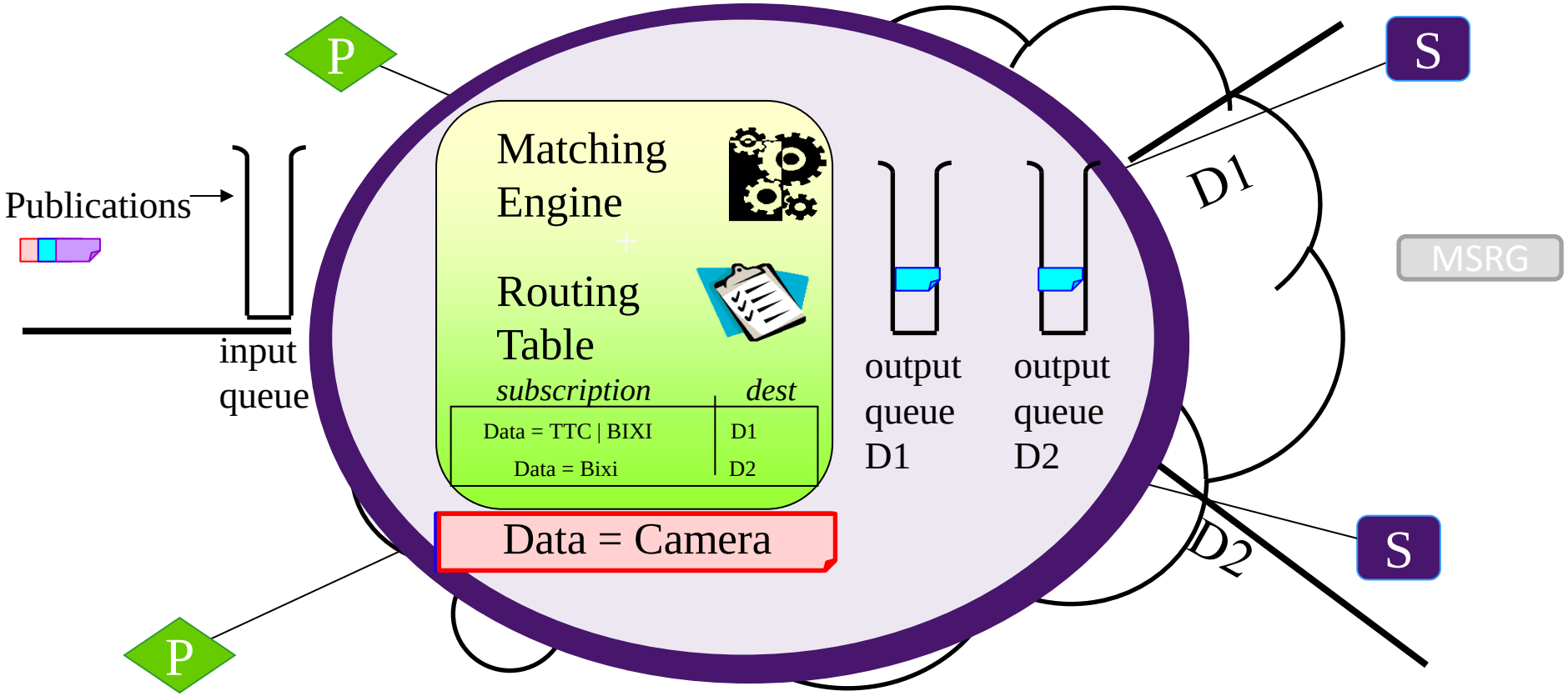
Subscribe

Database Subscriber (Raw Data)

Portal

portal.cvst.ca

Publish/Subscribe in CVST



Pluggable Engine Architecture