

Development of a cloud-based application to ingest, validate, analyze, and map data from a large PM sensor network

BRANDON FEENSTRA

SOUTH COAST AIR QUALITY MANAGEMENT DISTRICT

Perspective:

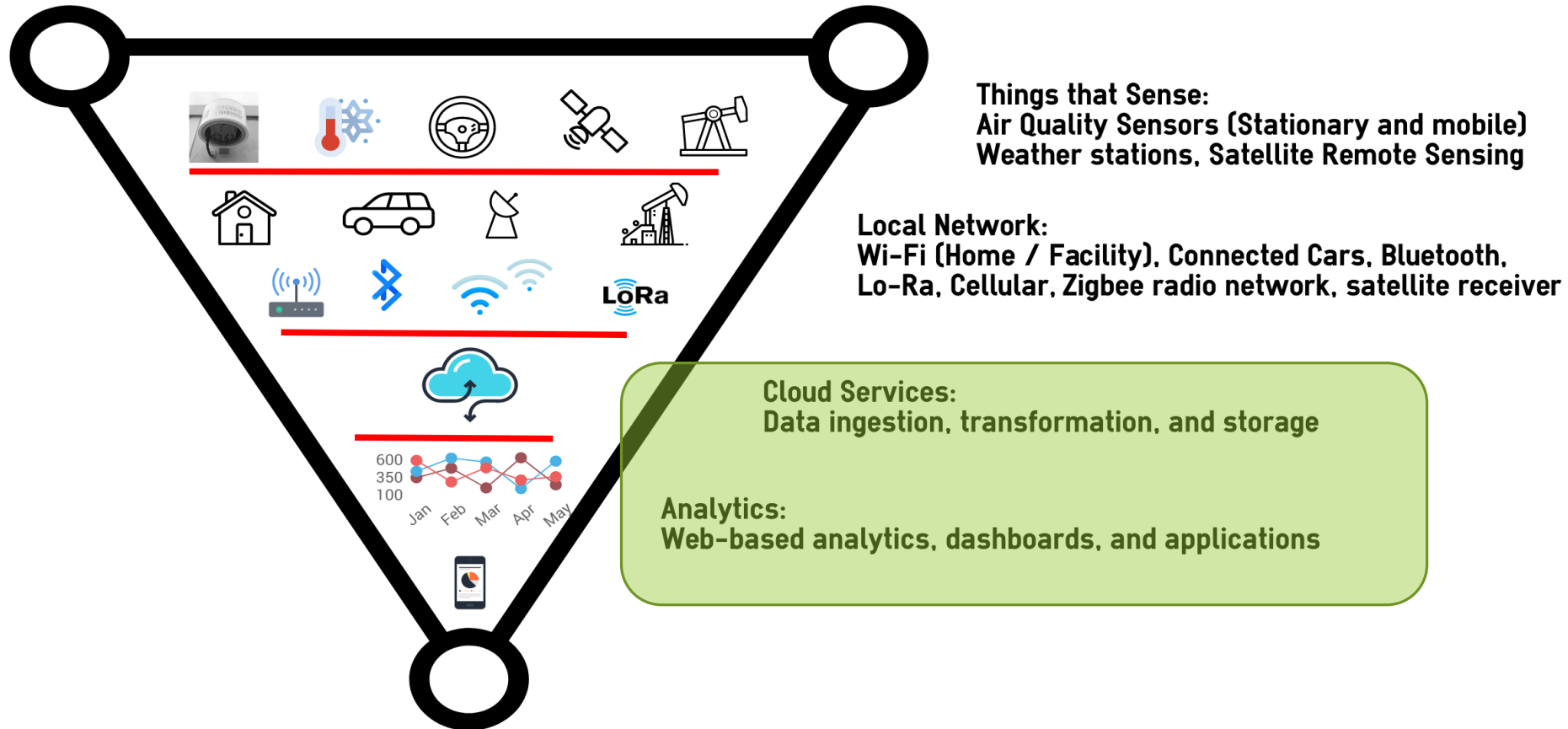
- ~~Sensor Developer~~
- ~~Single Project Researcher~~
- Government
- End User
- Multiple projects

Outline

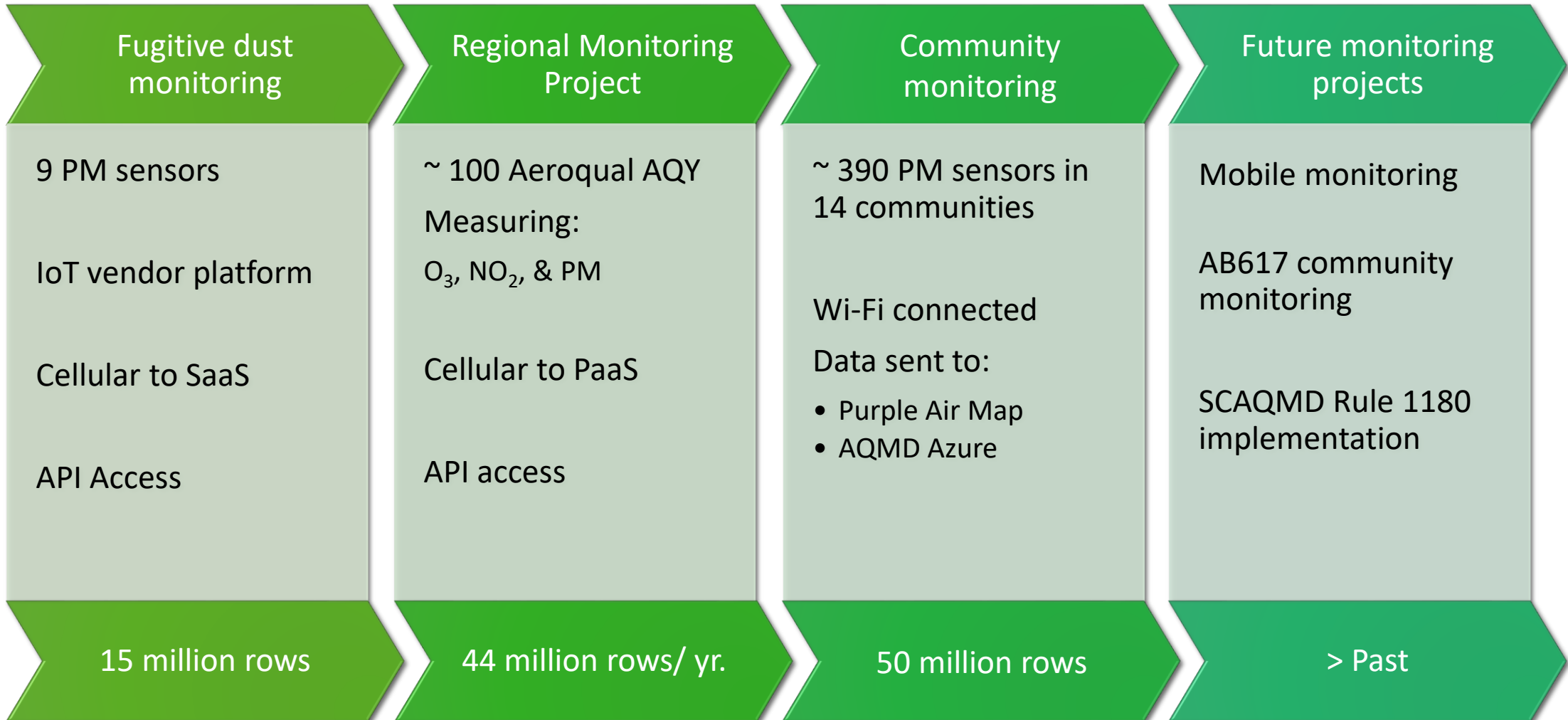
- SCAQMD Sensor Networks
- Development Drivers
- Requirements (High-level; Back- and front-end)
- Current Draft Architecture
- Lessons learned

Sensor Network Model

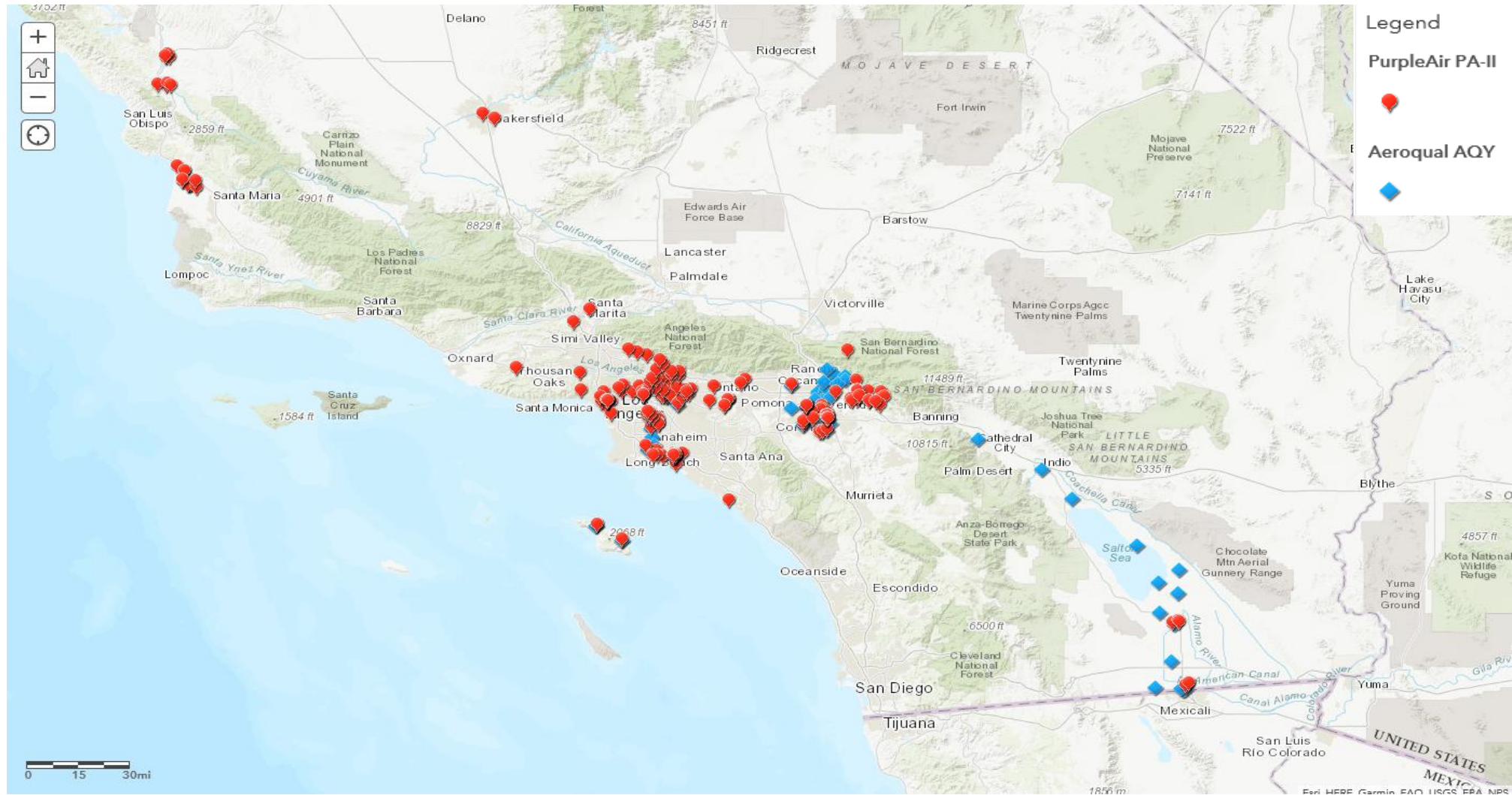
Model for Internet of Things (Air Quality)



SCAQMD Sensor Networks



Sensor Network Deployments



Why is a Data Management Platform needed?

Drivers for development

Multiple sensor / IoT data platforms used for various projects with data in different formats

Data analysis workloads larger than typical tools can handle ~

- 50 million rows of Purple air data and growing
- 44 million rows of Aeroqual data will be generated in 12 months
- 14 million rows of fence-line monitoring data

Limited data analytics available on individual IoT platforms

Limited external user experience with potential for confusing user experiences

Apply QA/QC functions to create validated data set

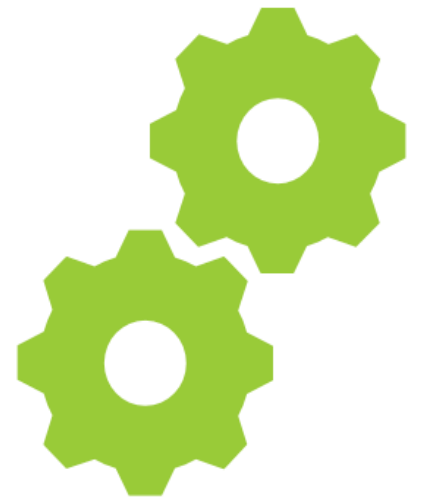
Apply correction algorithms for sensor performance limitations

Quickly visualize and provide results to public in a clear and meaningful manner

High level platform requirements:

Cloud-based computing platform to ingest, store, analyze, and visualize data

- Platform agnostic
- Device/sensor agnostic
- Scalable
- Secure
- Reliable
- Data standards based



Back-end Requirements

Server side of platform

- Includes infrastructure that does the heavy lifting of the project
- Not seen, yet requires significant effort to design and implement
- Essential for end-user experience in front-end application

Requirements:

- Ingest IoT air quality sensor data
- Manage IoT devices
- Stream data and perform stream analytics
- Store and process geo-spatial time series

Front-end Requirements

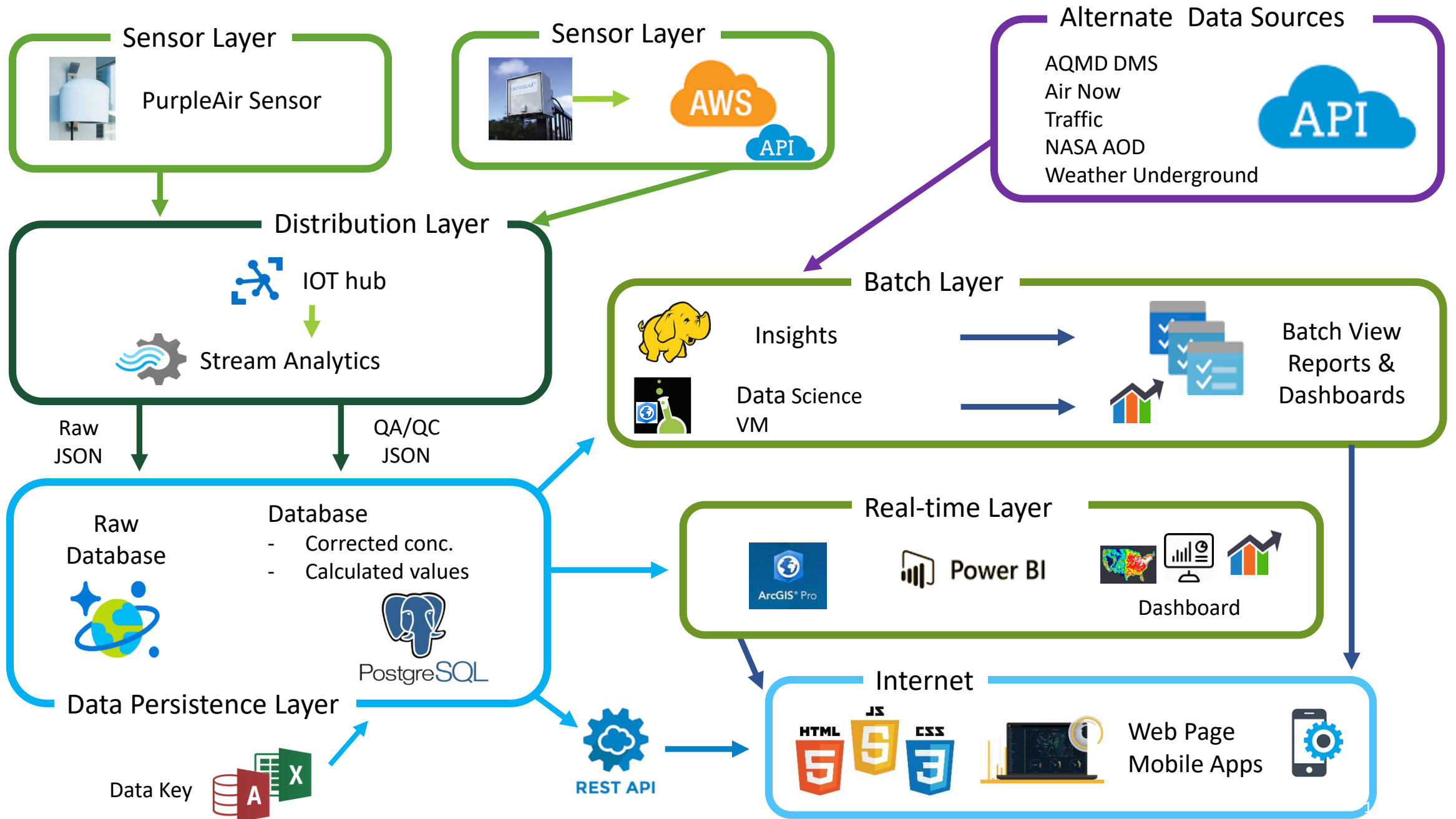
Visual experience of the platform

- Conversion of data to graphical interface for web viewing
- Dependent on back-end infrastructure
- Involves HTML, CSS, and JavaScript

Requirements:

- Create dashboard through a query builder platform
- Ability publish dashboards
- Ability for geo-spatial time series data analysis

Draft Cloud Architecture



Connected Sensors

258

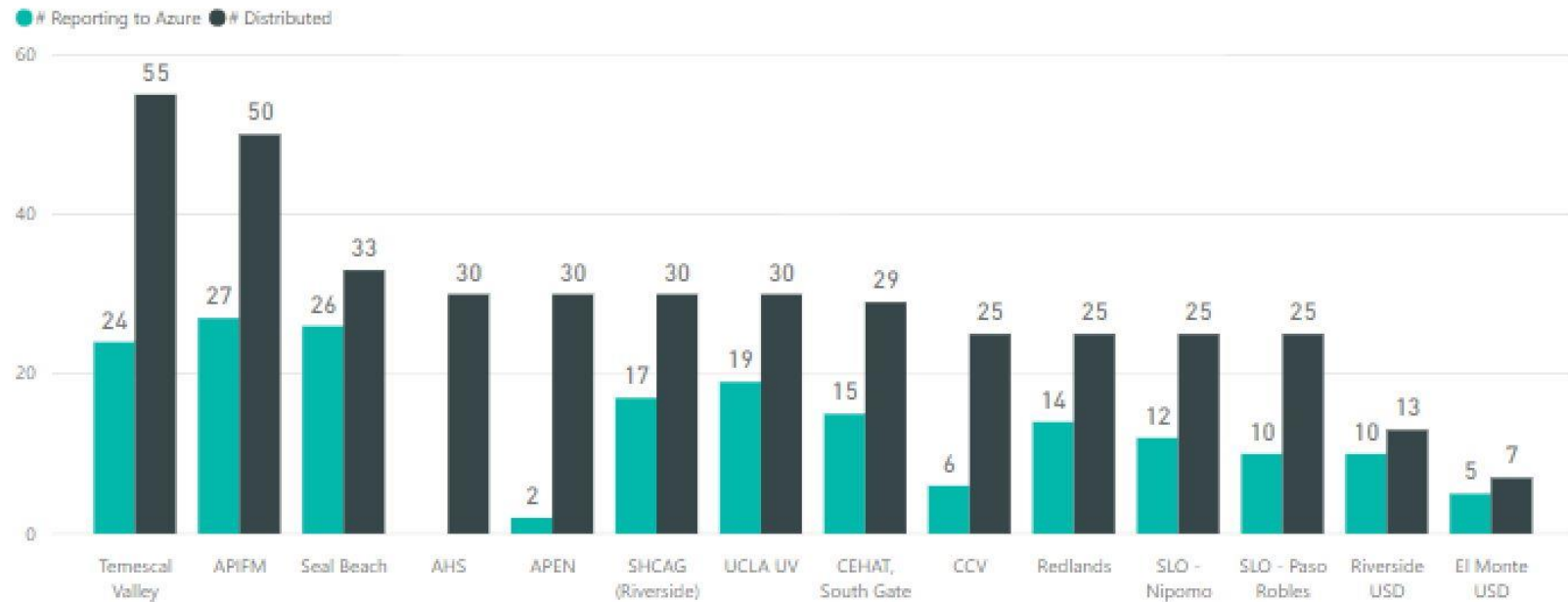
Count of SensorId

Rows of Data

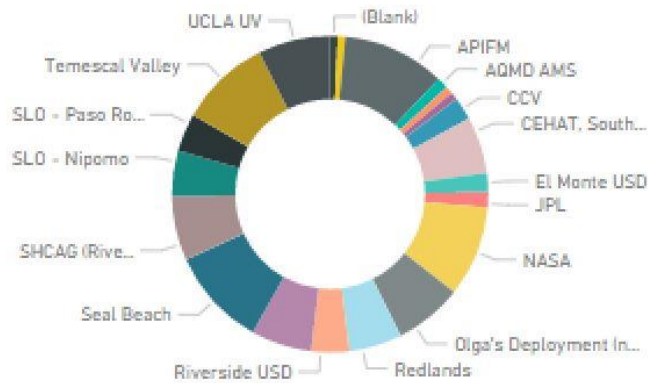
1.50M

Count of SensorId

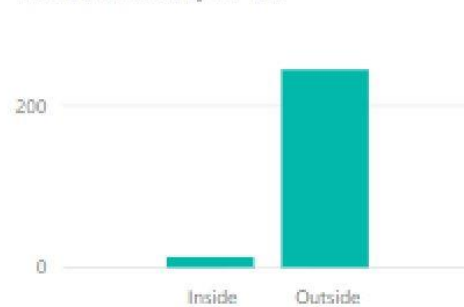
STAR Grant Community Deployment Status (Last 7 days)



Count of SensorId by Group



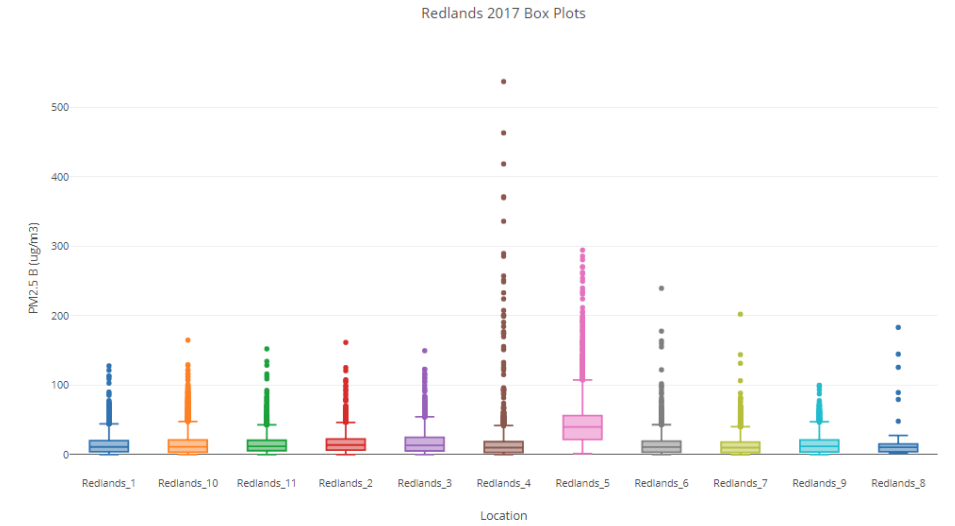
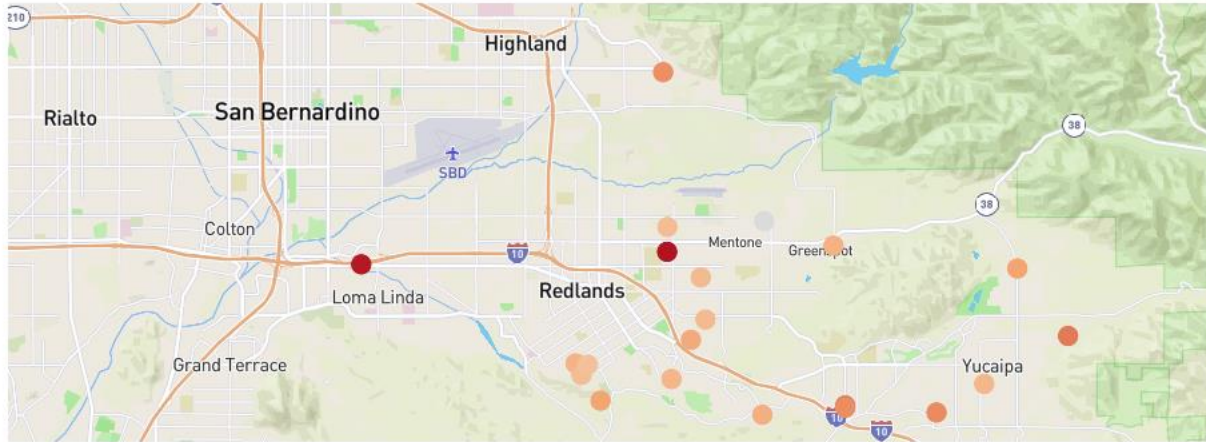
Count of SensorId by In / Out



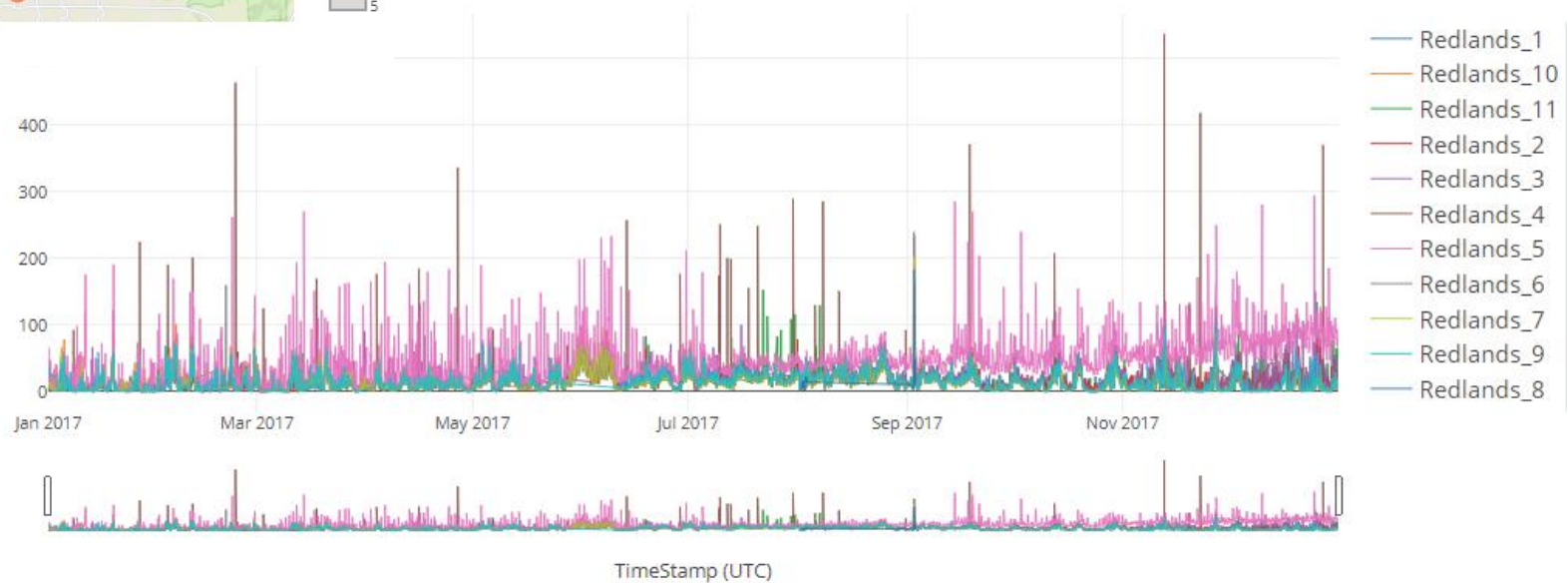
SensorId	MAC #	Name	Group	Pr...
18:fe:34:ddb0:97	18:fe:34:ddb0:97	Highland 1	Redlands	Re...
18:fe:34:ddb1:69	18:fe:34:ddb1:69	Redlands 10	Redlands	Re...
18:fe:34:ddb2:3a	18:fe:34:ddb2:3a	Redlands 2	Redlands	Re...
18:fe:34:ddb2:b	18:fe:34:ddb2:b	Redlands 4	Redlands	Re...
18:fe:34:ddb2:d6	18:fe:34:ddb2:d6	Redlands 9	Redlands	Re...
18:fe:34:ddb3:6f	18:fe:34:ddb3:6f	Redlands 11	Redlands	Re...
18:fe:34:ddb3:e	18:fe:34:ddb3:e	Redlands 5	Redlands	Re...
18:fe:34:ddb4:4d	18:fe:34:ddb4:4d	Yucaipa 7	Redlands	Re...
18:fe:34:ddb8:fe	18:fe:34:ddb8:fe	Redlands 7	Redlands	Re...
18:fe:34:ddb9:aa	18:fe:34:ddb9:aa	Yucaipa 3	Redlands	Re...

Interactive Dashboard

Redlands Deployment Locations



Redlands Time Series 2017



[Dashboard Link](#)

Best Practices

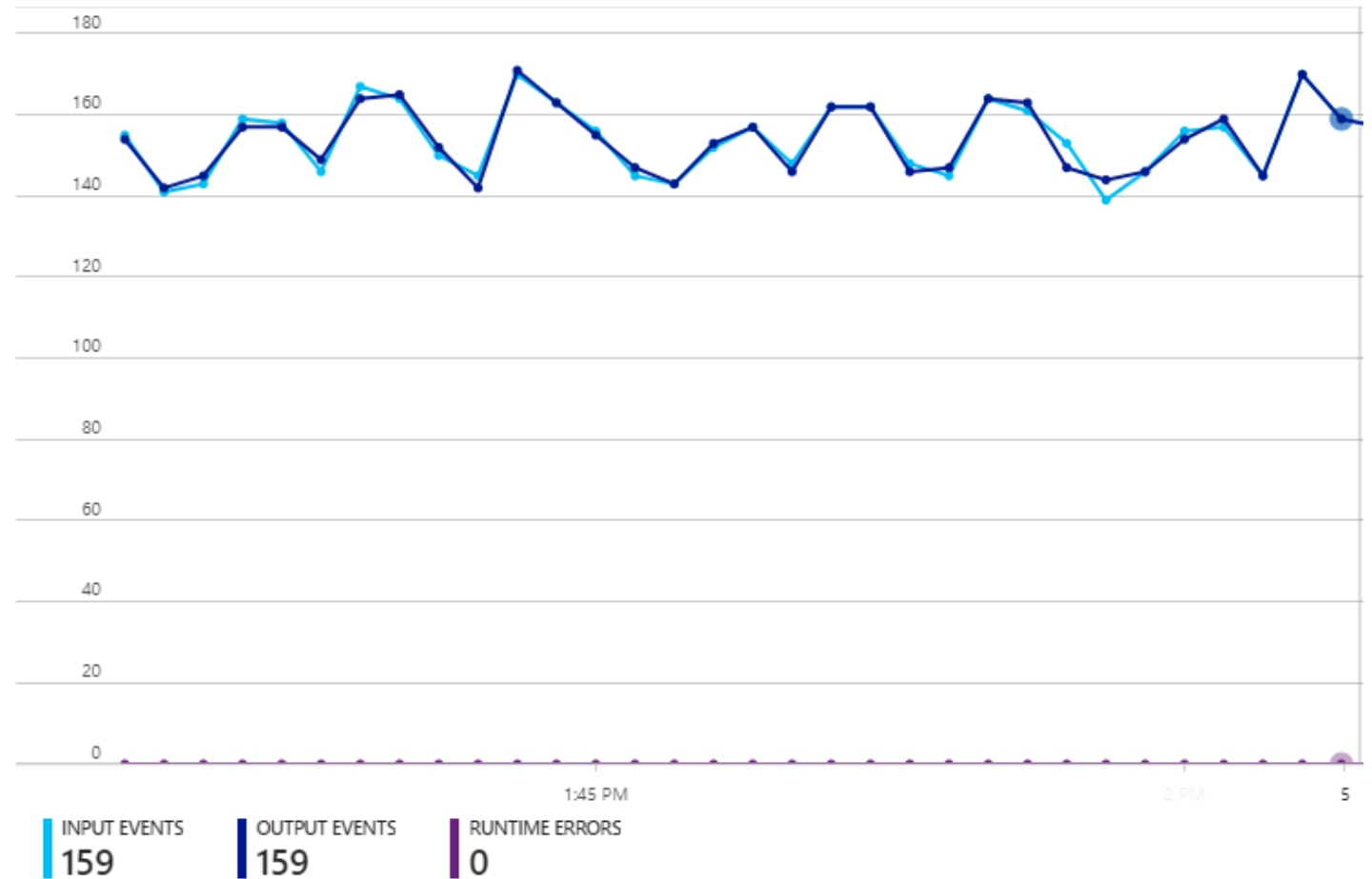
Data Streams

Inputs / Outputs

- Track metrics
- Provision appropriately
- Set alerts
- Know your anomalies

Monitoring Stream Analytics Input and Outputs 

NO SUBTITLE 



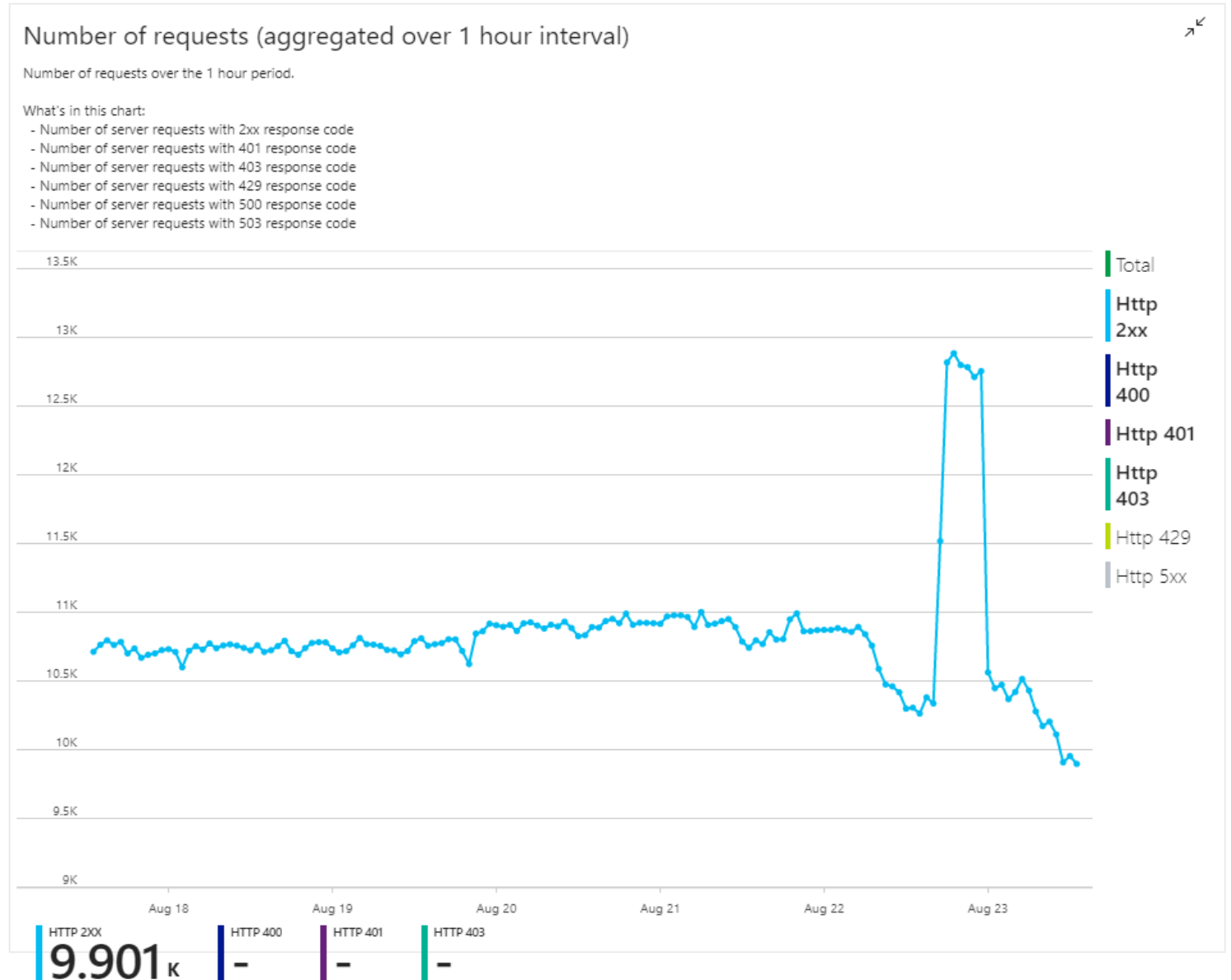
[View alerts for this resource \(1 configured\)](#)

Best Practices

Database

of Request / hour

- Track metrics
- Provision appropriately
- Set alerts
- Know your anomalies



Thank you

South Coast Air Quality Management District

AQ-SPEC team

Brandon Feenstra

bfeenstra@aqmd.gov



Extra Slides

Solution Approaches

- ❑ Ground up custom build
 - Time intensive, costly, potentially limited flexibility for future changes
 - More difficult to keep device agnostic
- ❑ Open source implementation
 - Less time intensive as build upon prior work
 - Pay for development of features that need that are not already available with OS
- ❑ Data standards based approach
 - Leverage tools (visualization and analysis) already built on data standards
 - More likely for broad acceptance of work when based on a data standard
- ❑ Hybrid approach