

# Making petrol stations “talk”

Challenges when implementing an end to end IoT solution

# About DiUS

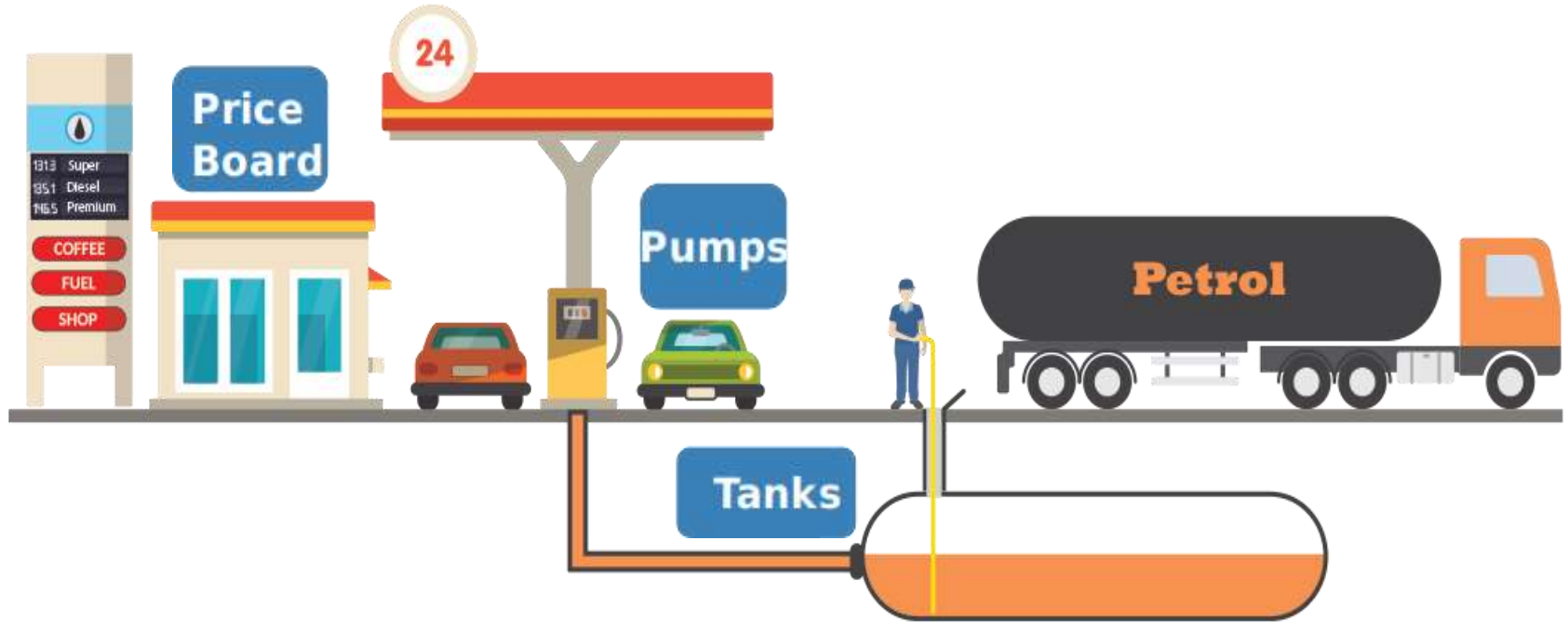
- Technology services company
- End to end solutions (including hardware)
- AWS Innovation Partner of the Year
- [dius.com.au](https://dius.com.au)



# Petrol stations



# Petrol station



# Tank connections

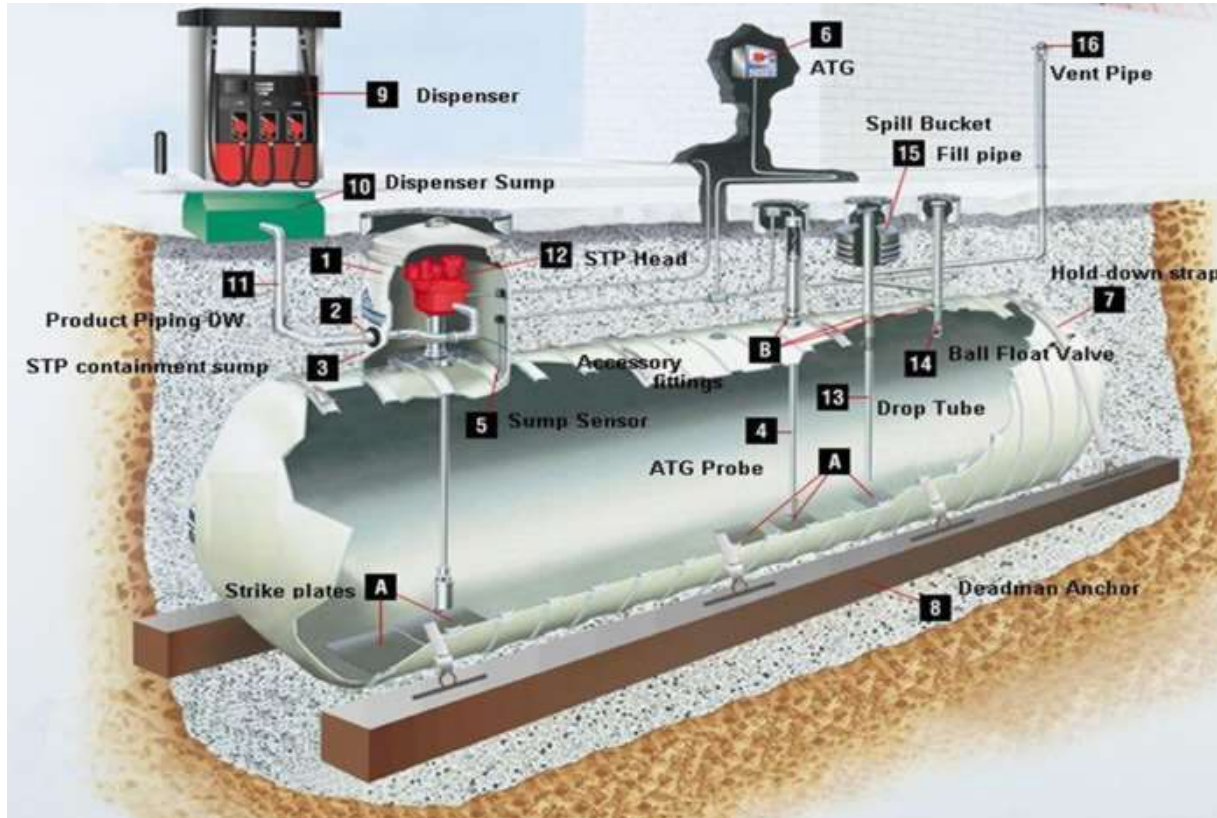
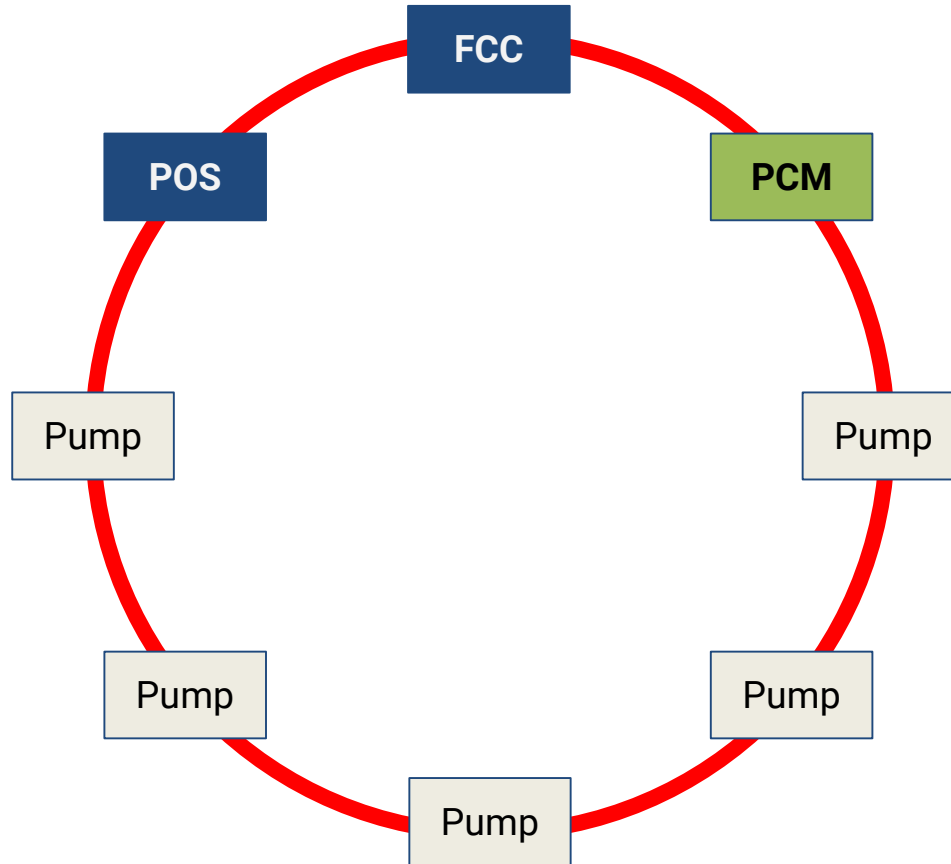


Image from [http://www.interspecllc.net/Content/Images/UST\\_Diagram.jpg](http://www.interspecllc.net/Content/Images/UST_Diagram.jpg)



# Petrol station current loop



# ATG - Automatic Tank Gauge



- Connects to tanks
- Collects relevant tank information
- Limited storage



# ATG - Automatic Tank Gauge

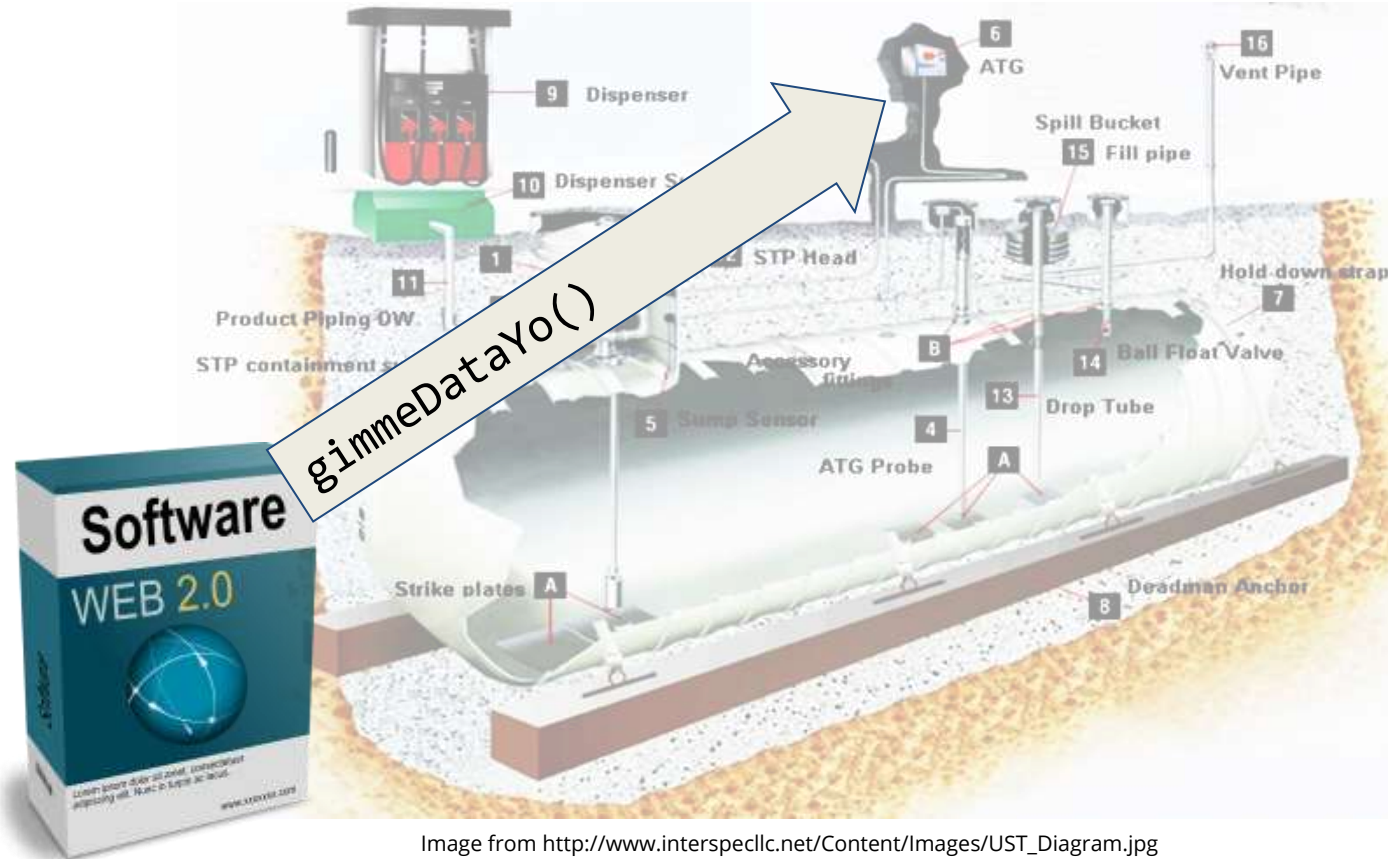


- Connects to tanks
- Collects relevant tank information
- Limited storage
- Introvert





# Data retrieval



# The problem

- Old hardware and protocols
- Reactive system
- Slow response times



# The problem



**EMS**



What if we can  
predict problems  
before they occur?



# The problem



**EMS**

How can I monitor  
the pumps on the  
forecourt as well?



# DiUS embedded team



Joo



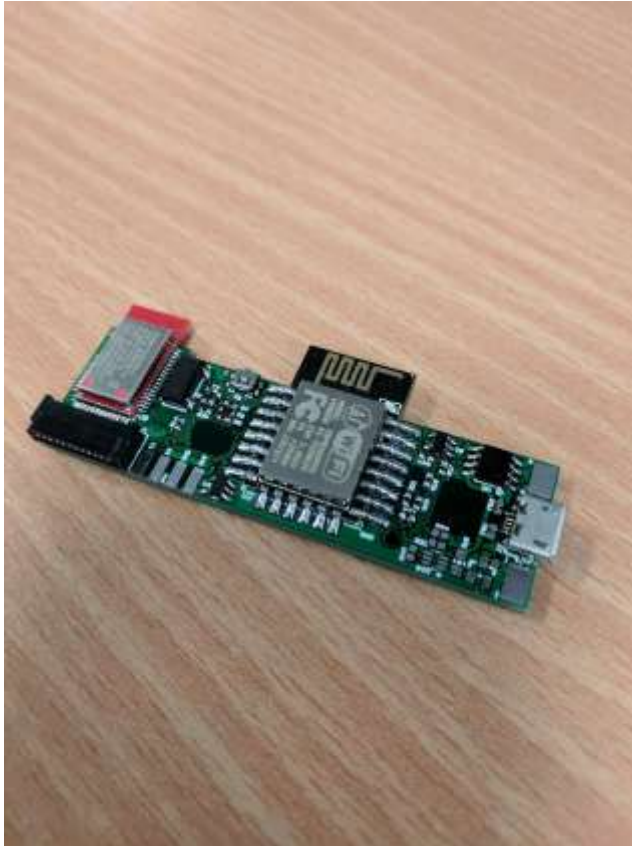
Johnny



Bernie



# Power sensor

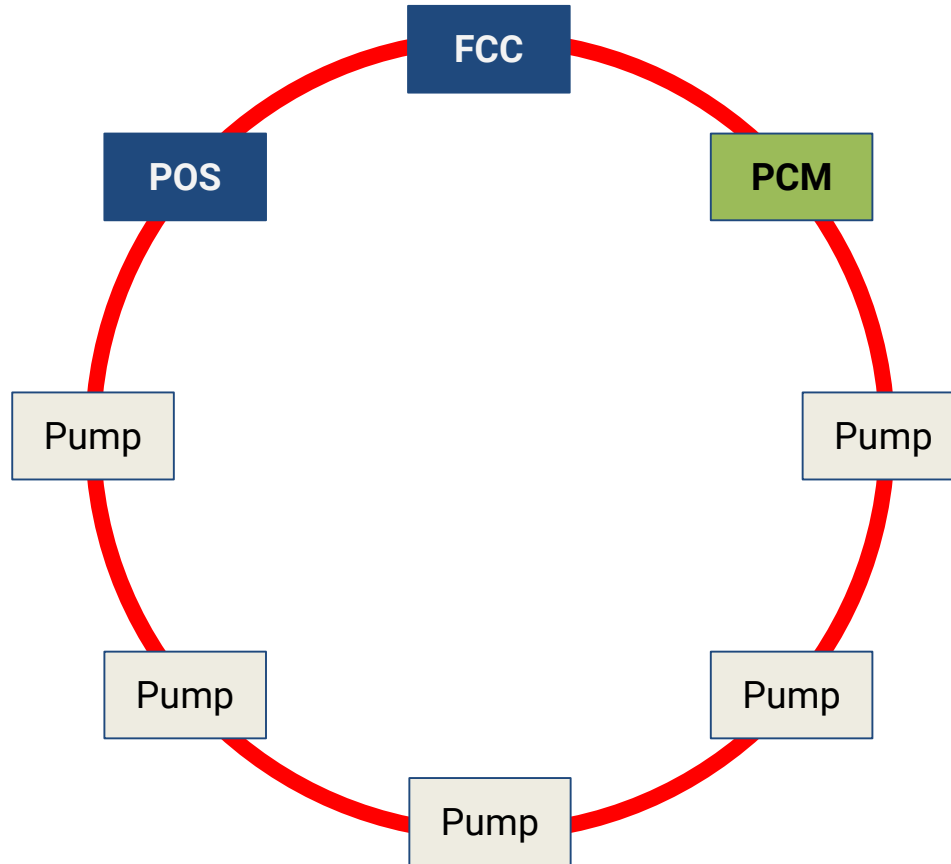




# PCM - Pump Communication Module



# Petrol station current loop





# PCM - Pump Communication Module



1. Fuel dispensed events
2. Pump online / offline
3. Fuel not dispensed events
4. "Current loop" status



# SDA - Station Data Aggregator



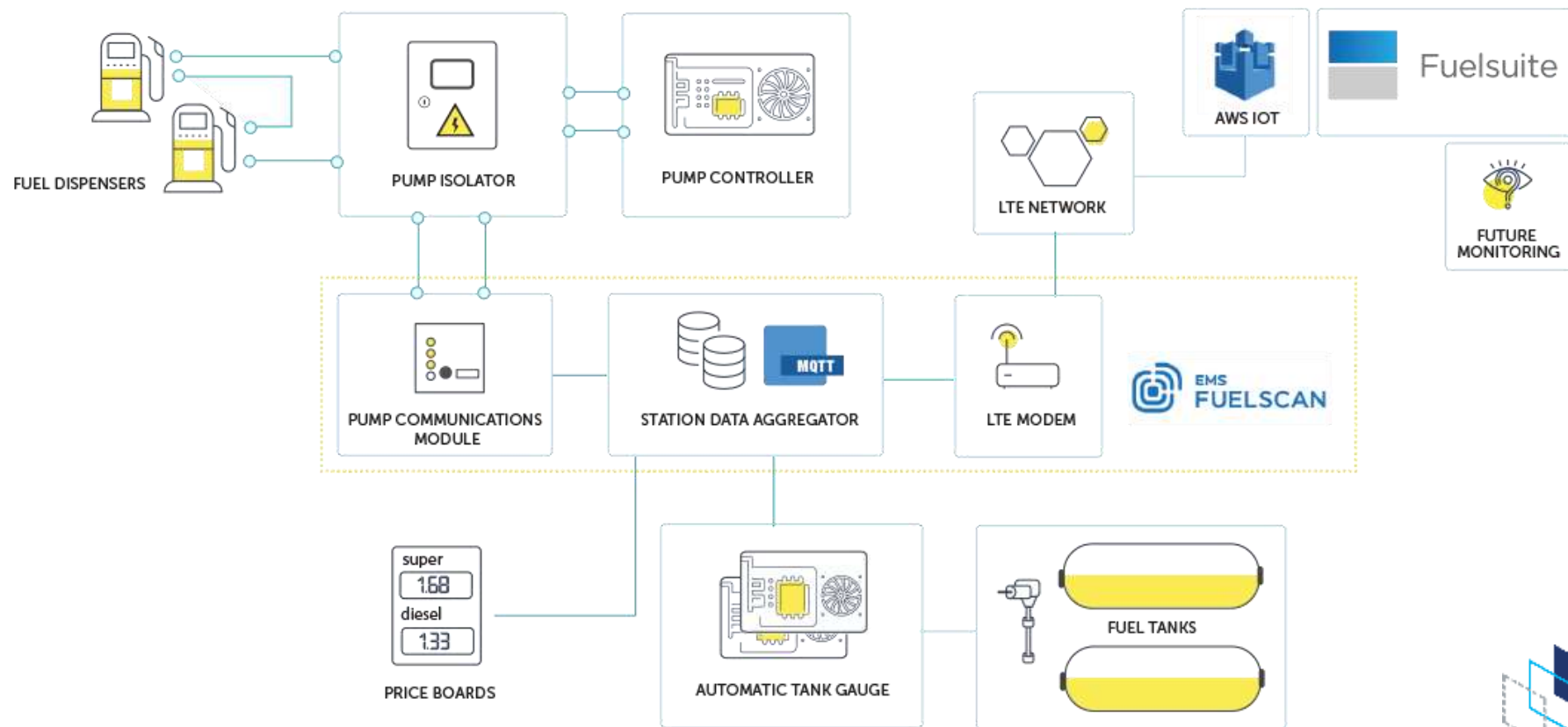
# SDA - Station Data Aggregator



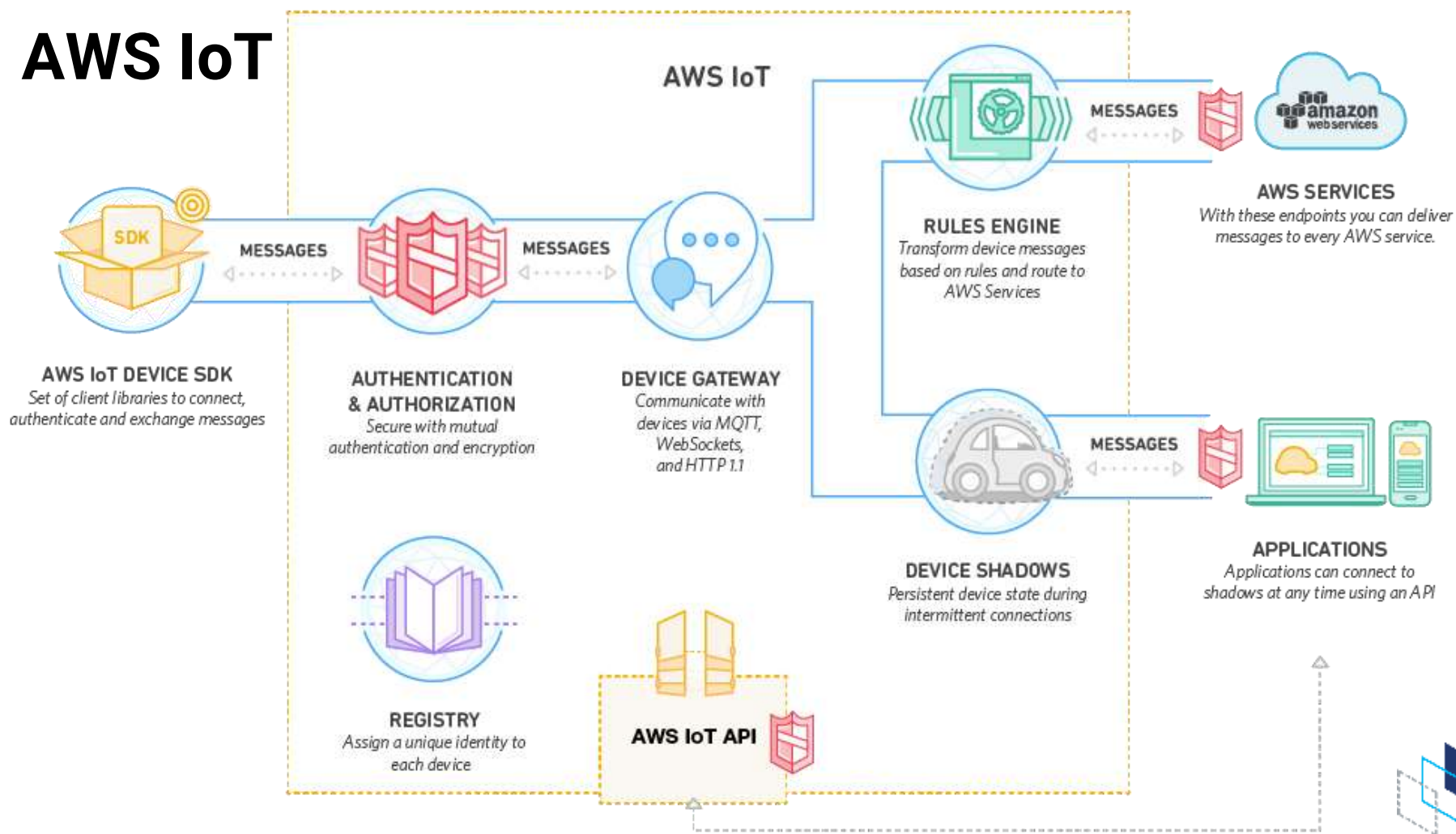
1. System information events
2. System configuration management
3. Priceboard configuration
4. Data aggregation and transformation



# Fuelscan



# AWS IoT



# Fuelsuite

**Fuelsuite** | VIVA ENERGY | ADMIN | Search for Site ID, Site Name, Network | ENE ADMIN

**Viva Energy, VIC CEXP**

STATUS OVERVIEW | ALARMS | INVENTORY

17 Alarms: Only active alarms

Accessed polling status for 6 sites. Submitted @ Fuelcell 5.

SITE NAME & ID	SYSTEM	TANK	SENSOR	VIEW	LINE	VEI	COMPLIANCE	FOLLOWED
ALBIONDALE 1002	✓	✓	▲ +1	✓	✓	✓	▲ +2	✓
BRECKON FARM 1004	✓	▲ +2	✓	✓	✓	✓	▲ +1	✓
BRAESIDE 1006	✓	▲ +1	✓	✓	✓	✓	✓	✓
MORNINGTON 1008	✓	▲ +2	✓	✓	✓	✓	✓	✓
REYNOLDS 1007	▲ +2	✓	✓	✓	✓	✓	✓	✓
Sandwich 1005	✓	▲ +1	▲ +2	✓	✓	✓	✓	✓
SEELING NORTH 1003	✓	▲ +1	✓	✓	✓	✓	✓	✓
SNYDDEN 1001	✓	▲ +1	✓	✓	✓	✓	✓	✓

Who We Are | Contact Your Analyst

**Fuelsuite** | VIVA ENERGY | ADMIN | Search for Site ID, Site Name, Network | ENE ADMIN

**MORNINGTON - F006**

STATUS OVERVIEW | ALARMS | INVENTORY | DELIVERIES | COMPLIANCE | LEAK TESTS | ENTER SITE DATA

17 Alarms: All Active

**ACTIVE ALARMS**

SENSOR & UNIT	ALARM TYPE	STATUS	LAST & TIME (MM/DD/YY)
Tank 1 - U-PH	Manufacture gas BT alarm	Active - Alarm	16/08/2017 10:00:00
Tank 2 - U-PH	Manufacture gas BT alarm	Active - Alarm	16/08/2017 10:00:00

0 Files

**ALARM HISTORY - Past 30 Days**

SENSOR & UNIT	ALARM TYPE	STATUS	LAST & TIME (MM/DD/YY)
Tank 1 - U-PH	Delivery needed warning	Clear	16/08/2017 10:00:00
Tank 2 - U-PH	Delivery needed warning	Clear	16/08/2017 10:00:00
Tank 1 - U-PH	Delivery needed warning	Clear	16/08/2017 10:00:00
Tank 2 - U-PH	Delivery needed warning	Clear	16/08/2017 10:00:00
Tank 1 - U-PH	Delivery needed warning	Clear	16/08/2017 10:00:00
Tank 2 - U-PH	Delivery needed warning	Clear	16/08/2017 10:00:00
Tank 1 - U-PH	Delivery needed warning	Clear	16/08/2017 10:00:00
Tank 2 - U-PH	Delivery needed warning	Clear	16/08/2017 10:00:00
Tank 1 - U-PH	Delivery needed warning	Clear	16/08/2017 10:00:00
Tank 2 - U-PH	Delivery needed warning	Clear	16/08/2017 10:00:00
Tank 1 - U-PH	Delivery needed warning	Clear	16/08/2017 10:00:00
Tank 2 - U-PH	Delivery needed warning	Clear	16/08/2017 10:00:00
Tank 1 - U-PH	Delivery needed warning	Clear	16/08/2017 10:00:00
Tank 2 - U-PH	Delivery needed warning	Clear	16/08/2017 10:00:00
Tank 1 - U-PH	Delivery needed warning	Clear	16/08/2017 10:00:00
Tank 2 - U-PH	Delivery needed warning	Clear	16/08/2017 10:00:00
Tank 1 - U-PH	Delivery needed warning	Clear	16/08/2017 10:00:00
Tank 2 - U-PH	Delivery needed warning	Clear	16/08/2017 10:00:00
Tank 1 - U-PH	Delivery needed warning	Clear	16/08/2017 10:00:00
Tank 2 - U-PH	Delivery needed warning	Clear	16/08/2017 10:00:00

**Fuelsuite** | SITES | REPORTS | ADMIN | Search for Site ID, Site Name, Network | ENE ADMIN

**Coles Express - Euroa Retail**

SITE OVERVIEW | ALARMS | INVENTORY | CHARTS | COMPLIANCE | DELIVERIES | LEAK TESTS | NOTES | Enter Site Data

Inventory and water level chart

DATA RANGE: 16 July 2017 | 16 Aug 2017

Legend: T1 (Fuel Level), T2 (Water Level), T3 (Fuel Level), T4 (Water Level), T5 (Fuel Level), T6 (Water Level)

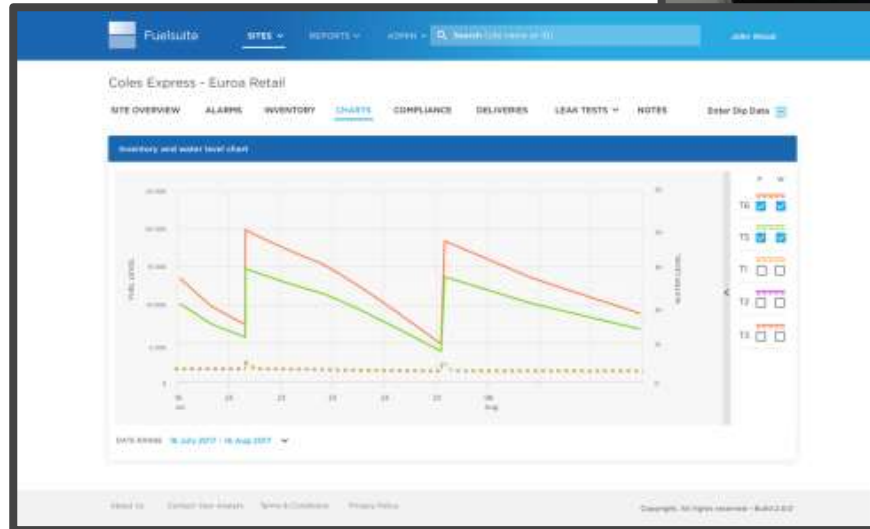
Who We Are | Contact Your Analyst | Site & Database | Privacy Policy | Copyright, All Rights reserved - 2017 FUELSUITE





# Fuelscan + Fuelsuite

- Proactive system
- Predictive analytics
- Identify unknown unknowns



# Challenges

- Monitor and manage remote devices at a large scale nationally?
- Visualise the data captured?
- Manage all this data coming down?





# Device configuration

# Device shadow

## What is it?

- **A JSON document that is used to store and retrieve current state information for a device.**

- AWS IoT Developer Guide

- <http://docs.aws.amazon.com/iot/latest/developerguide/iot-thing-shadows.html>



# Device shadow

```
{
  "state": {
    "desired": {
      "online": true
    },
    "reported": {
      "online": false
    }
  }
}
```



# Device shadow

**Fuelsuite**

```
{  
  "state": {  
    "desired": {  
      "online": true  
    },  
    "reported": {  
      "online": false  
    }  
  }  
}
```

**Fuelscan**



# Priceboard shadow JSON

```
{
  "state": {
    "reported": {
      "priceboard": {
        "display": {
          "1": "Unleaded",
          "3": "Diesel"
        }
      }
    }
  }
}
```



# Priceboard shadow JSON

```
{
  "state": {
    "desired": {
      "priceboard": {
        "display": {
          "1": "Diesel"
        }
      }
    },
    "reported": {
      "priceboard": {
        "display": {
          "1": "Unleaded",
          "3": "Diesel"
        }
      }
    }
  }
}
```



# Priceboard shadow JSON

```
{
  "state": {
    "desired": {
      "priceboard": {
        "display": {
          "1": "Diesel"
        }
      }
    },
    "reported": {
      "priceboard": {
        "display": {
          "1": "Unleaded",
          "3": "Diesel"
        }
      }
    }
  }
}
```



```
{
  "state": {
    "reported": {
      "priceboard": {
        "display": {
          "1": "Diesel",
          "3": "Diesel"
        }
      }
    }
  }
}
```



# Priceboard shadow JSON

```
{
  "state": {
    "desired": {
      "priceboard": {
        "display": {
          "1": "Diesel",
          "3": null
        }
      }
    },
    "reported": {
      "priceboard": {
        "display": {
          "1": "Unleaded",
          "3": "Diesel"
        }
      }
    }
  }
}
```



```
{
  "state": {
    "reported": {
      "priceboard": {
        "display": {
          "1": "Diesel",
          "3": "Diesel"
        }
      }
    }
  }
}
```





# Device shadow

- **Caveat**
  - a request for deletion of a property in the desired section does not get passed through to the device for deletion
- **AWS Docs**
  - To delete a key, send a null value
  - `{ ... "key": null, ... }`



# Priceboard shadow JSON

```
{
  "state": {
    "desired": {
      "priceboard": {
        "display": {
          "1": "Diesel",
          "3": []
        }
      }
    },
    "reported": {
      "priceboard": {
        "display": {
          "1": "Unleaded",
          "3": "Diesel"
        }
      }
    }
  }
}
```



# Priceboard shadow JSON

```
{
  "state": {
    "desired": {
      "priceboard": {
        "display": {
          "1": "Diesel",
          "3": []
        }
      }
    },
    "reported": {
      "priceboard": {
        "display": {
          "1": "Unleaded",
          "3": "Diesel"
        }
      }
    }
  }
}
```



```
{
  "state": {
    "reported": {
      "priceboard": {
        "display": {
          "1": "Diesel"
        }
      }
    }
  }
}
```



# Device shadow

**Be explicit.**



# Message consumption

# Message consumption

## Many types of messages

- **Alarms**
- **Inventory**
- **Leak tests**
- **Tank tests**
- **Deliveries**
- **Tank configuration**
- **Sensor configuration**
- **Pricing**



# Message consumption

What is the problem?

- We need to configure a petrol station by consuming messages ensuring historical data was captured



# Tank configuration message

```
{
  ...
  "LabelList": [
    {
      "Tank": 1,
      "Label": "DIESEL"
    },
    {
      "Tank": 2,
      "Label": ""
    }
  ],
  ...
}
```





# Tank activation message

```
{
  ...
  "ConfigList": [
    {
      "Tank": 1,
      "Flag": 1
    },
    {
      "Tank": 2,
      "Flag": 0
    }
  ],
  ...
}
```



# Alarm event message

```
{  
  ...  
  "AlarmEvent": {  
    "Tank": 1,  
    "Category": 2,  
    "Type": 5,  
    "SensorCategory": 2,  
    "State": 2,  
    "EventTime": "1706020638"  
  },  
  ...  
}
```



# Message consumption



AWS IoT



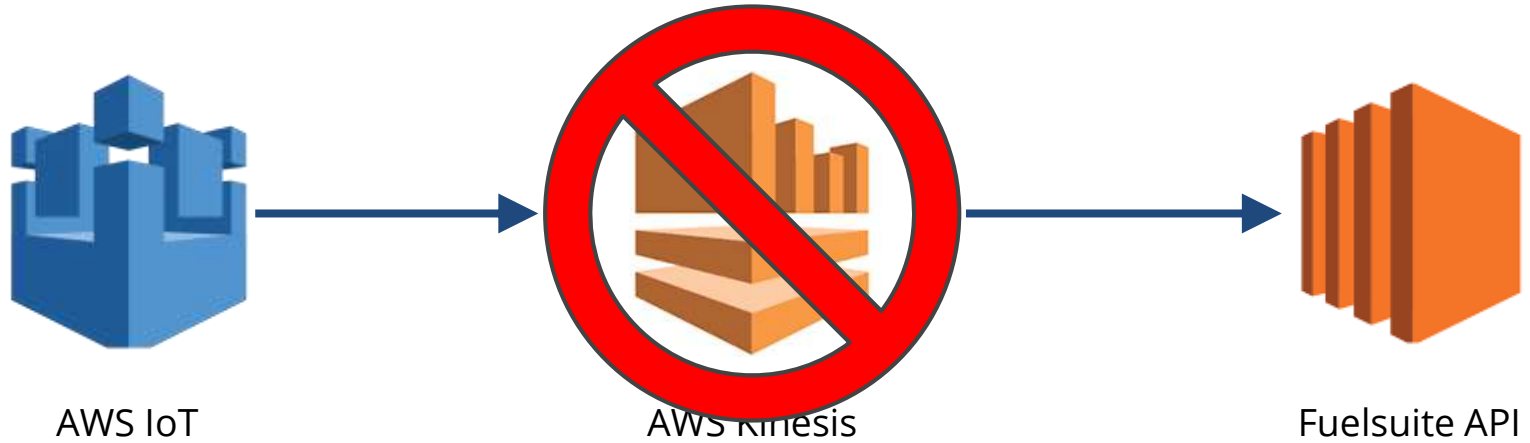
AWS Kinesis



Fuelsuite API



# Message consumption



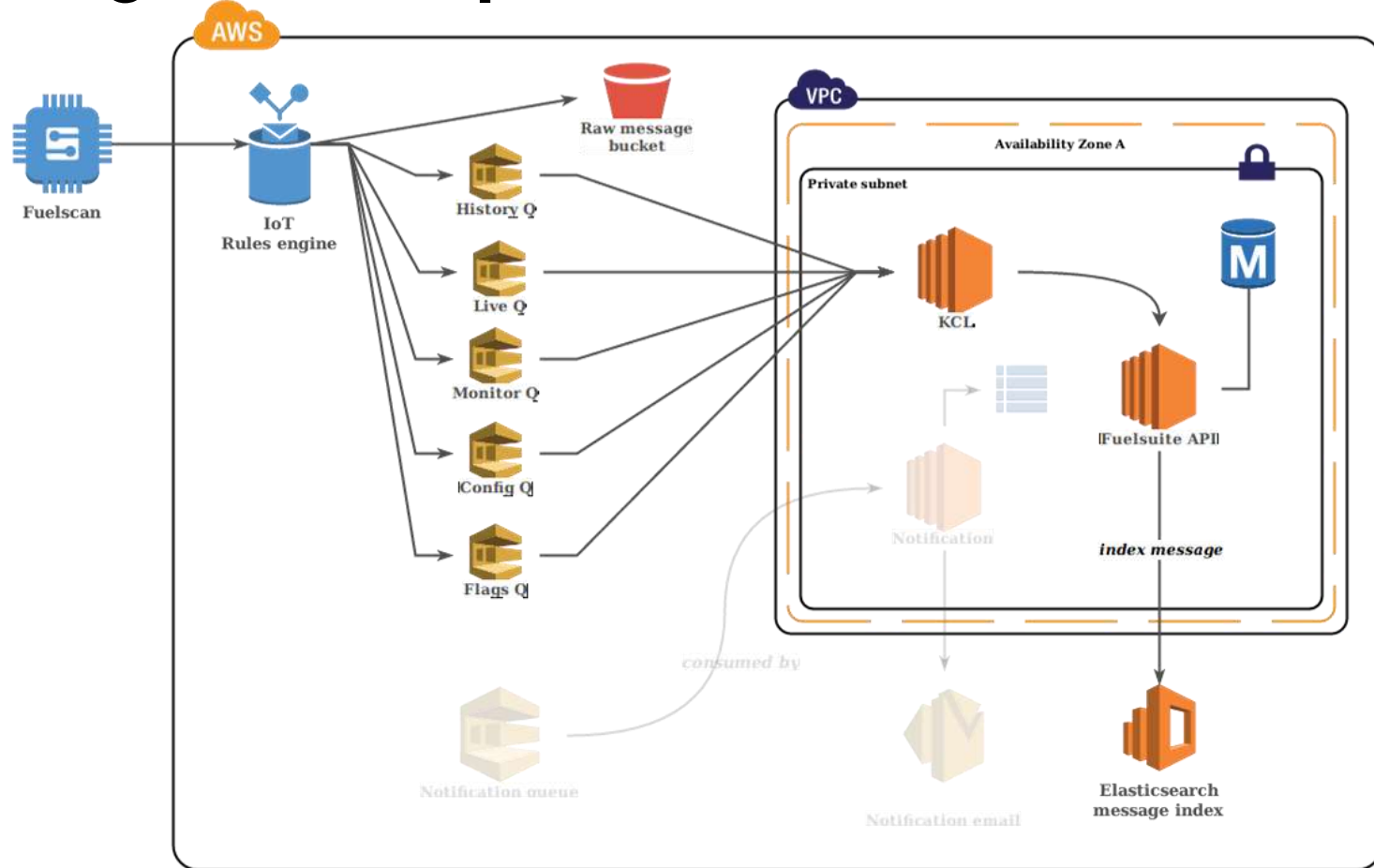
# Message consumption

## Classes of messages

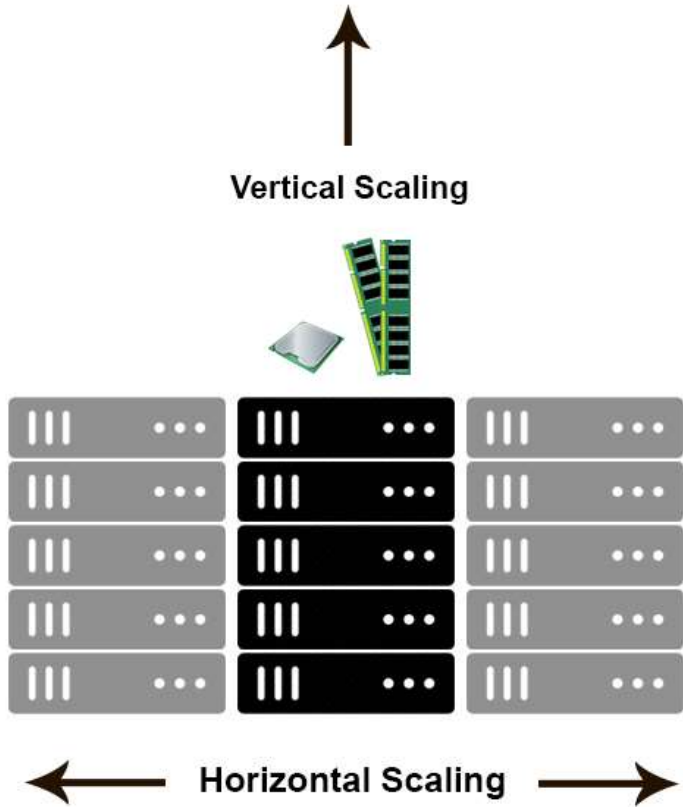
- Configuration
- Configuration flags
- Live
- Monitor
- History



# Message consumption



# Scalability



- **Horizontal scaling**
  - Add more boxes
- **Vertical scaling**
  - Add more resources to a box



# Scalability



```
unitech@t450: ~/keymetrics/pm2/examples
>>> pm2 show process-metrics
Describing process with id 1 - name process-metrics

status      online
name        process-metrics
restarts    0
uptime      13s
script path /home/unitech/keymetrics/pm2/examples/process-metrics.js
script args N/A
error log path /home/unitech/.pn2/logs/process-metrics-error-1.log
out log path /home/unitech/.pn2/logs/process-metrics-out-1.log
pid path    /home/unitech/.pn2/pids/process-metrics-1.pid
interpreter node
interpreter args N/A
script id   1
exec cwd    /home/unitech/keymetrics/pm2/examples
exec mode   fork_node
node.js version 6.9.1
watch & reload X
unstable restarts 0
created at  2016-11-28T13:04:55.760Z
```

#### Revision control metadata

```
revision control  git
remote url       git@github.com:Unitech/pm2.git
repository root  /home/unitech/keymetrics/pm2
last update      2016-11-28T13:05:08.882Z
revision         915849c1ac4a5e2af8dc846d36abb0b089e800b2
comment          Merge branch 'master' into development
branch           development
```

#### Code metrics value

```
Loop delay  0.33ms
Counter     128
```

Add your own code metrics: <http://bit.ly/code-metrics>  
Use 'pm2 logs process-metrics [--lines 1000]' to display logs  
Use 'pm2 monit' to monitor CPU and Memory usage process-metrics



# Data management

# Classes of messages

- **Configuration**
- **Configuration flags**
- **Live**
- **Monitor**
- **History**



# Alarm event message

## Frequency

- **every time there's a change**
- **checked every 30 seconds**

```
{  
  ...  
  "AlarmEvent": {  
    "Tank": 1,  
    "Category": 2,  
    "Type": 5,  
    "SensorCategory": 2,  
    "State": 2,  
    "EventTime": "1706020638"  
  },  
  ...  
}
```



# Inventory levels message

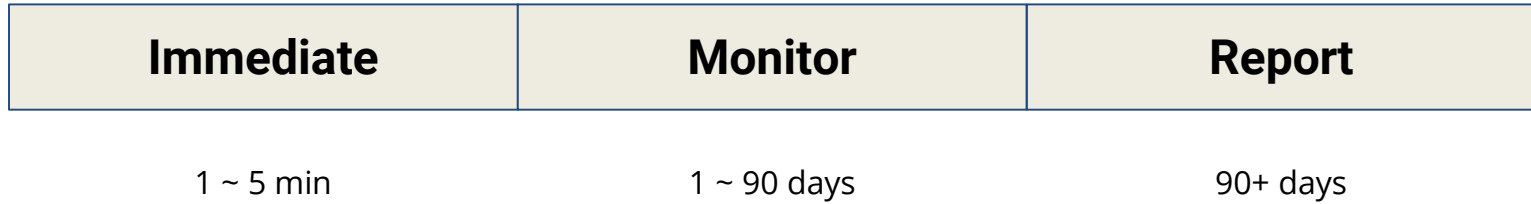
## Frequency

- every 30 seconds

```
{
  ...
  "InventoryList": [
    {
      "Tank": 1,
      "Type": "2",
      "Status": 0,
      "Volume": 34943.12,
      "TCVolume": 34856.07,
      "Ullage": 17306.9,
      "Height": 1495.77,
      "Water": 0,
      "Temperature": 16.95,
      "WaterVolume": 0
    },
    ...
  ],
  ...
}
```



# Windows of interest



# Windows of interest



- Identify time/date range
- Identify frequency of access of data
- Storage requirements for each window



# Closing thoughts

# Lesson learned

- Real time does not mean “streaming”
- Ensure UX understands the domain
- Keep it lean





# Thank you!

@tarciosaraiva

@dius\_au

