INTELLIGENT HOME SECURITY WITH NVIDIA JETSON

PAUL DECARLO
SR. CLOUD ADVOCATE
@PJDECARLO

DECK AVAILABLE @ HTTPS://AKA.MS/INTELLIGENTHOMESECURITY

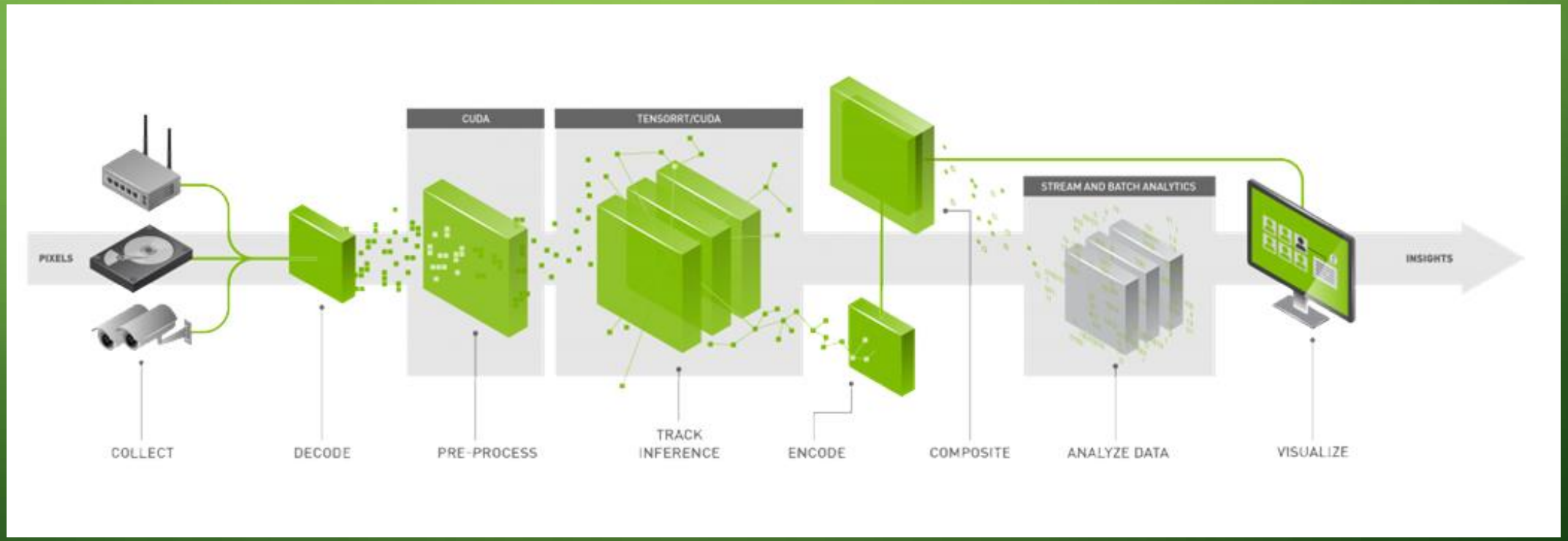
HARDWARE

- NVIDIA Jetson Device (Nano)

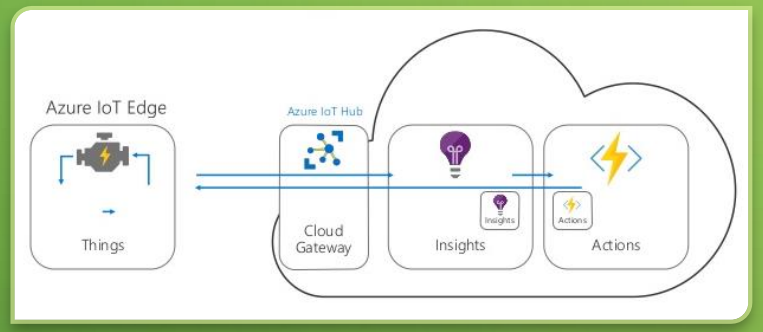
- 7-Inch Color Monitor

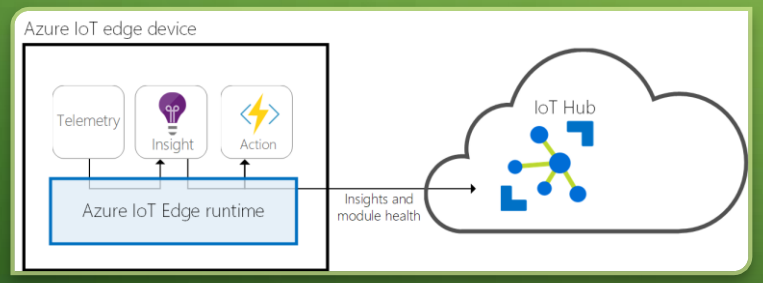
- HDMI Splitter

DEEPSTREAM SDK IOT EDGE MODULE



DEPLOY MODULES USING AZURE IOT EDGE





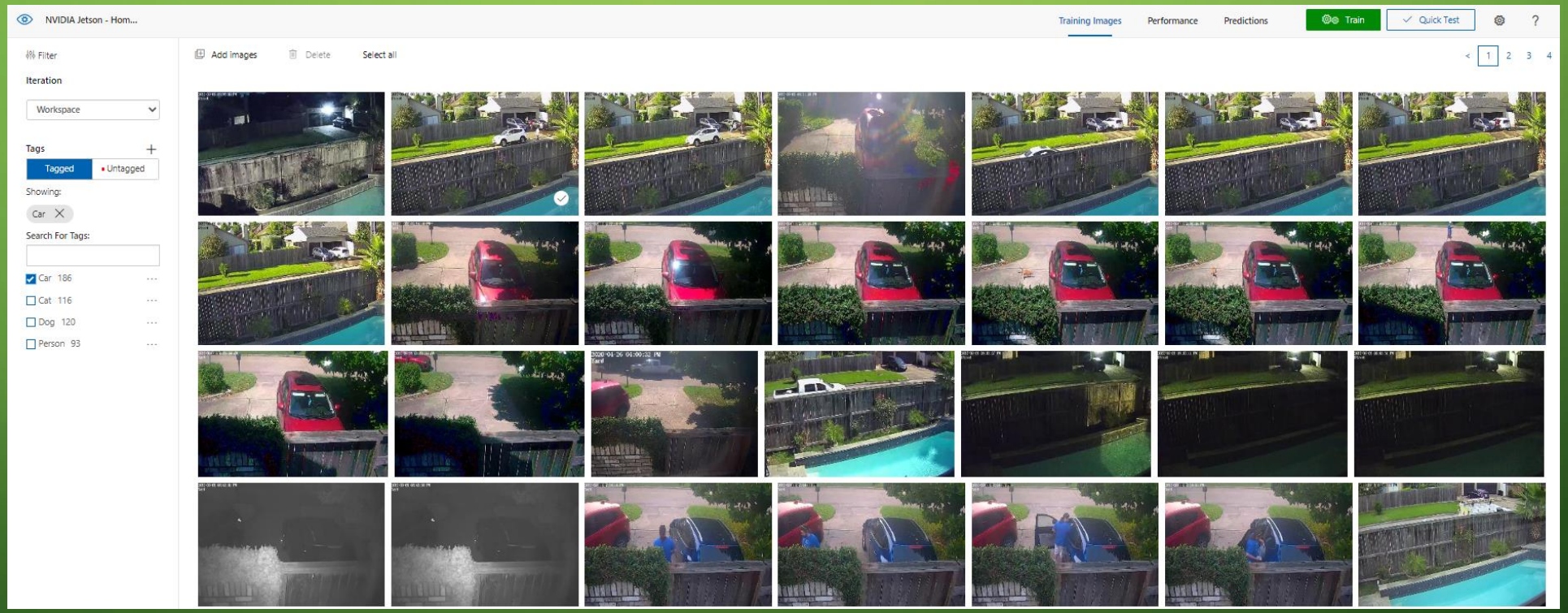
- Built on container technology as 'modules'
- Modules support Python, NodeJS, .Net Core, Java, & C
- Low-latency AMQP / MQTT data transport
- Operate in offline / intermittent network conditions
- Supports Linux X64 | ARM32/64 , Windows X64

- OSS and available @ https://github.com/Azure/iotedge

IOT EDGE CAMERA TAGGING MODULE

- Capture contextual images at site of deployment
- Push images to CustomVision.AI or Blob Storage (Local and Cloud)
- Automate via Direct Methods
- Facilitate ML Ops with automated sampling and iterative training

TRAIN AND EXPORT MODEL FROM CUSTOMVISION.AI



OBJECT DETECTION DATA OVERLOAD


```
{
    "version": "4.0",
    "id": 39,
    "@timestamp": "2020-05-06T00:51:23.071Z",
    "sensorId": "Yard",
    "objects": [
        "-1|532.308|150.769|1110.77|513.846|Car",
        "-1|6.15385|67.6923|323.077|424.615|Car"
    ]
},
{
    "version": "4.0",
    "id": 44,
    "@timestamp": "2020-05-06T00:51:23.234Z",
    "sensorId": "Yard",
    "objects": [
        "-1|532.308|147.692|1110.77|516.923|Car",
        "-1|6.15385|67.6923|323.077|427.692|Car"
    ]
}
```


- Messages are produced extremely fast (as high as 30 per second depending on algorithm and batch configuration options)
- Great for reacting to real-time events at the edge
- Way too much data to for summary reporting

```
WITH
FlattenedDetections AS
(
    SELECT
        DeepStreamInput.sensorId,
        (SUBSTRING (arrayElement.ArrayValue,
         REGEXMATCH(arrayElement.ArrayValue, '[a-z]'), LEN(arrayElement.ArrayValue))) as object,
        DeepStreamInput.[@timestamp], COUNT(DeepStreamInput.[@timestamp]) as matches
    FROM
        [DeepStreamInput] AS DeepStreamInput TIMESTAMP BY DeepStreamInput.[@timestamp]
        CROSS APPLY GetArrayElements(objects) AS arrayElement
    WHERE
       DeepStreamInput.[@timestamp] != CAST('1970-01-01T00:00:00.000Z' AS datetime) /*filter RTSP disconnections*/
       GROUP BY DeepStreamInput.[sensorId],
                arrayElement,
                DeepStreamInput.[@timestamp],
                SYSTEM.TIMESTAMP()
)

SELECT
    Count(object) AS count, /*Counting function*/
    sensorId, object, [@timestamp]
INTO [AggregatedDetections]
FROM FlattenedDetections
    WHERE matches = 1 /*Filter duplicates where (timestamp and object) are equal)*/
    GROUP BY
        sensorId,
        object,
        [@timestamp],
        TumblingWindow(second, 30)

SELECT
    FLOOR(AVG(count)) as count, /*Smoothing function*/
    sensorId, object, System.Timestamp AS [@timestamp]
INTO [SummarizedDetections]
FROM AggregatedDetections
GROUP BY
    sensorId,
    object,
    TumblingWindow(second, 30)
```

FLATTEN, AGGREGATE, AND SUMMARIZE DETECTIONS

- Stream Analytics Job deployed at Edge
- Parse object detections into flattened list
- Remove duplicates and count objects by @timestamp
- Floor the average count over a 30 second tumbling window

EXAMPLE SUMMARIZED PAYLOAD

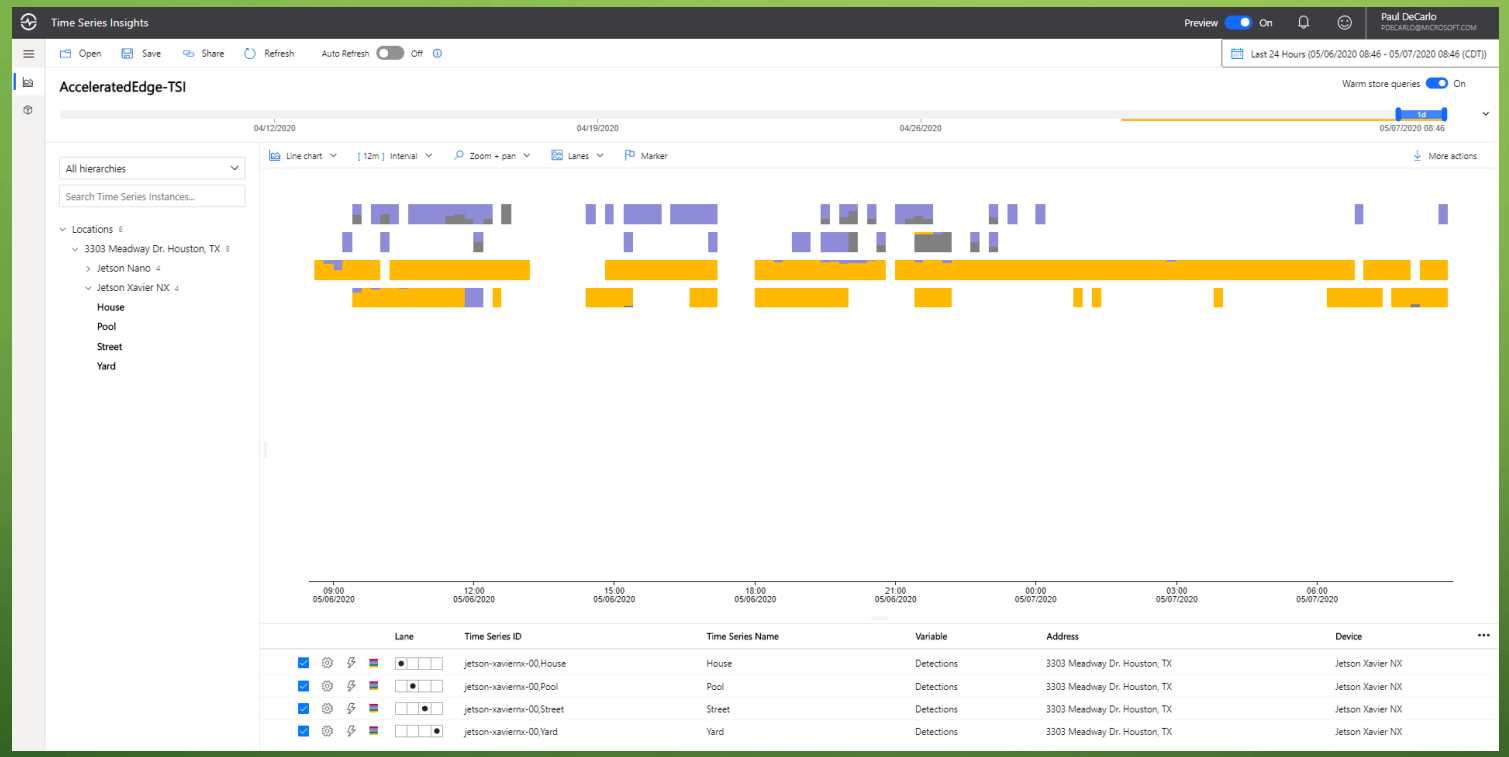
{"count":1.0,"sensorId":"Yard","object":"Car","@timestamp":"2020-05-07T13:37:00.0000000Z"}

{"count":1.0,"sensorId":"Street","object":"Car","@timestamp":"2020-05-07T13:37:00.0000000Z"}

{"count":1.0,"sensorId":"House","object":"Person","@timestamp":"2020-05-07T13:37:00.0000000Z"}

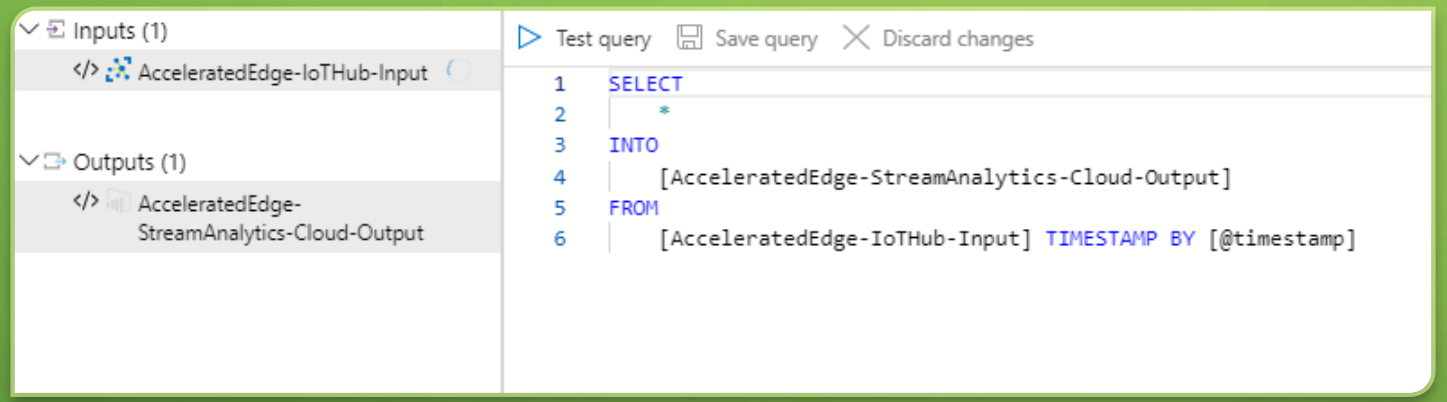
1 Car in the Yard, 1 Car in the Street, 1 Person in the House

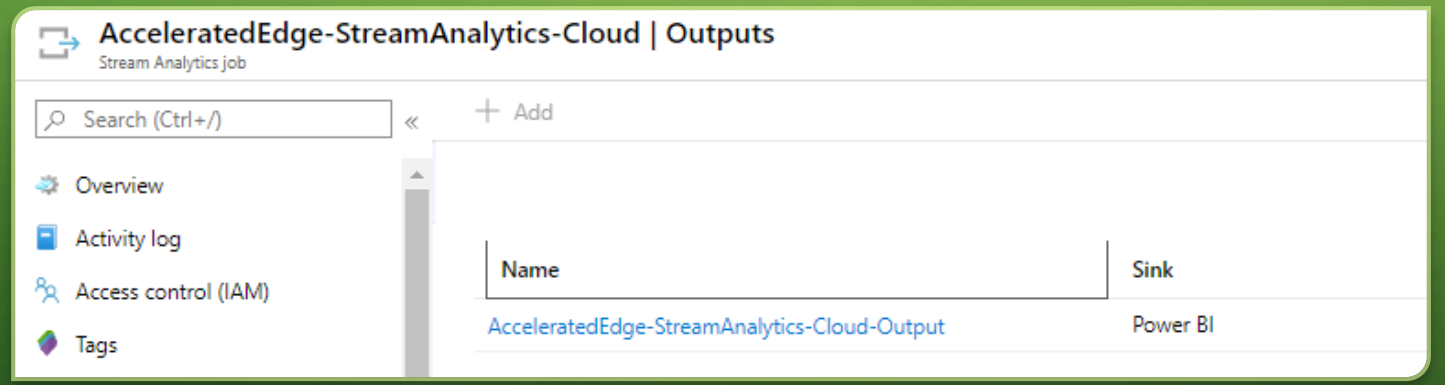
MODELING DATA INTO TSI



- Partition data by iothub-connection-device-id / sensorid
- Create Hierarchy to organize sensorId by location
- Create ObjectDetectionType to report detections of interest
- Easily visualize / summarize data over time intervals

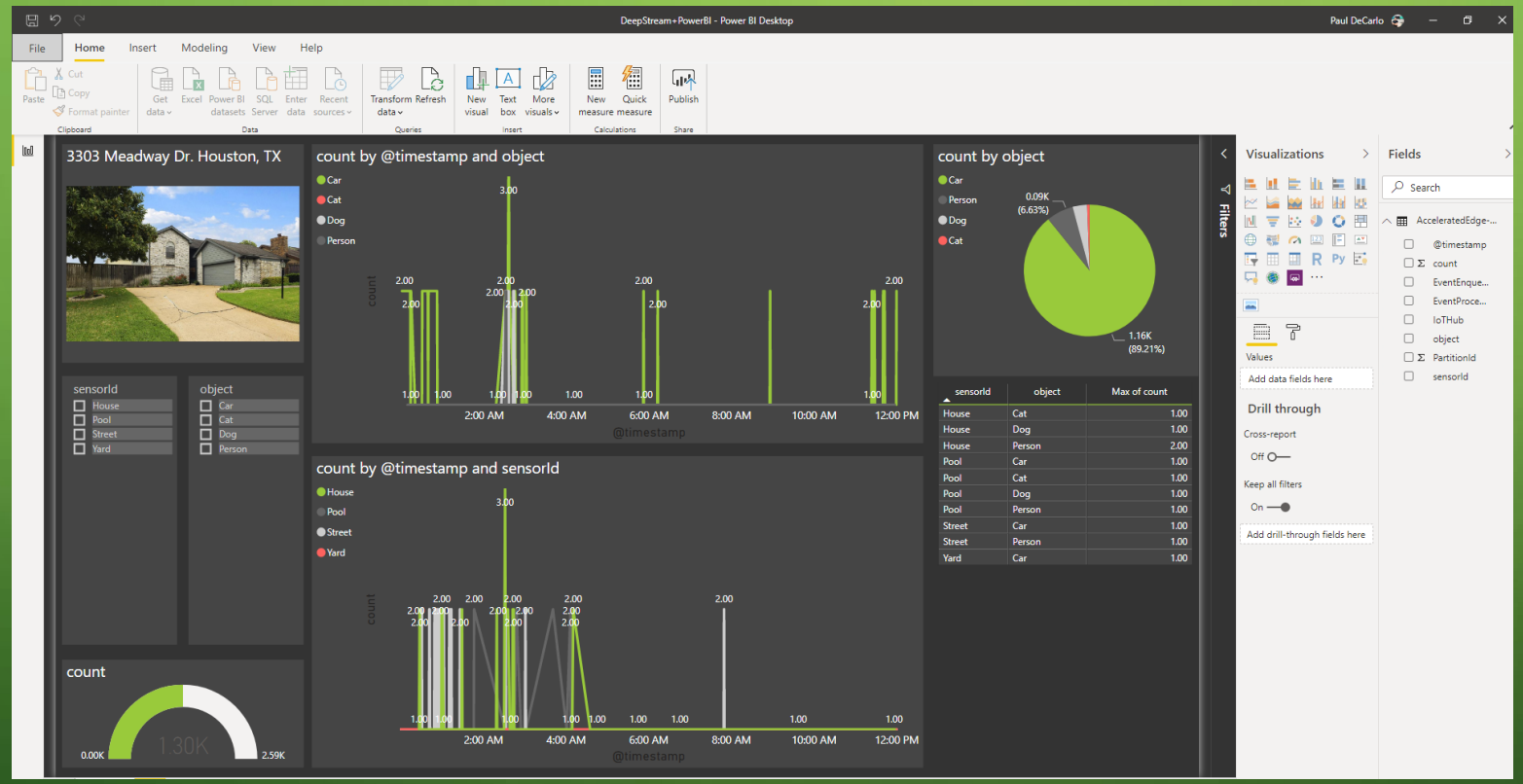
FORWARD STREAMING DATA TO POWER BI





- Select all incoming summarized data from IoT Hub
- Forward into Power BI Output Sink

CREATE A POWER BI REPORT



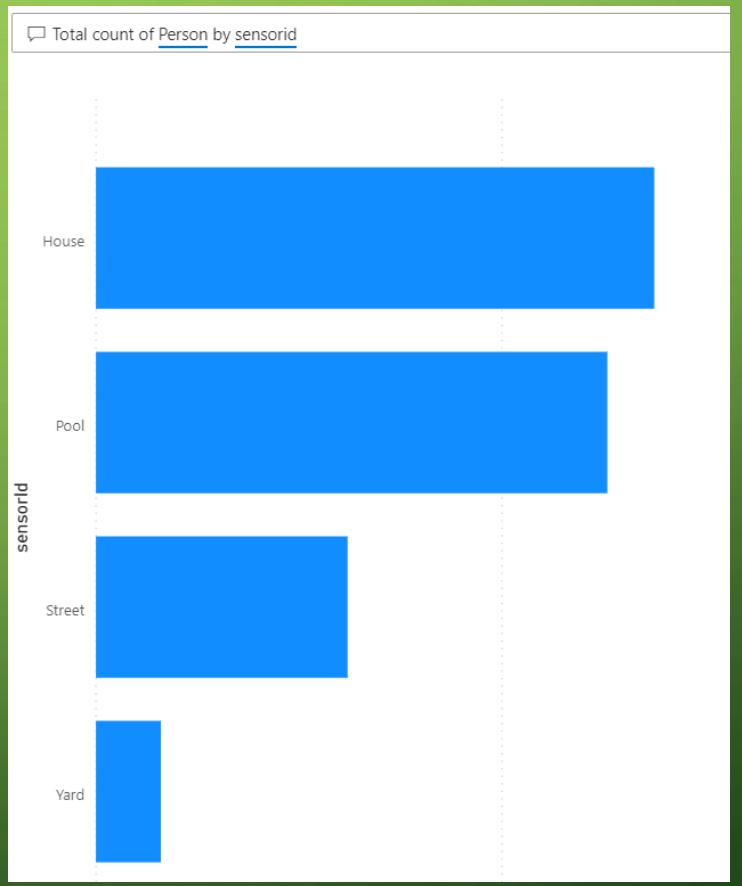
- Report based on live streaming data updated every 30 seconds as messages flow into IoT Hub
- Slice based on any combination of time, sensorid, or object
- Summarize data at a glance

PUBLISH DASHBOARD AND ASK QUESTIONS

Show last time Cat detected with sensorid

Showing results for *last Cat sensorid*

sensorId	Latest @timestamp
Pool	05/07/20 01:52:30 PM
Yard	05/07/20 01:11:00 PM
House	05/07/20 04:10:30 AM
Total	**05/07/20 01:52:30 PM**



RESOURCES

- **This PowerPoint Presentation**
  - https://aka.ms/IntelligentHomeSecurity
- **Introduction to the Azure IoT Edge Camera Tagging Module**
  - https://aka.ms/CameraTaggingIntro
- **IoT Edge at GTC Digital 2020**
  - https://aka.ms/EdgeAtGTC
- **Visual Anomaly Detection using NVIDIA Deepstream IoT (WorkShop)**
  - https://aka.ms/DeepStreamOnEdgeWorkshop
- **DeepStream edge-to-cloud integration with Azure IoT**
  - https://aka.ms/DeepStreamIoTEdgeWebinar

DETAILED REPRO INSTRUCTIONS @

aka.ms/intelligentvideoanalytics