Sam Vanhoutte

Azure MVP
< Belgium >

Time traveling in the cloud

Process time series data in Microsoft Azure

All samples available on:

▷ https://github.com/SamVanhoutte/azure-time-travel

SamVanhoutte

readme.md

- Belgium
- Microsoft Azure MVP (8 years)
- Advisor and invester in cloud-startups
- ex-CTO at Codit (17 years)
- Focus on Data & A.I., IoT and Serverless
- Passion for cycling & traveling

This session

- Time series characteristics
- Azure Time Series Insights & Azure Data Explorer
- Predictive maintenance scenario: architecture overview
- Azure Machine Learning (LSTM training)
- Azure Stream Analytics

Time series specifics

What makes Time Series data different?

Intents vs Facts

Intents	Facts
Messages	**Events**
Commands Query Job Assignment Update Request	Report Notification Measurement Trace Audit

Event Types

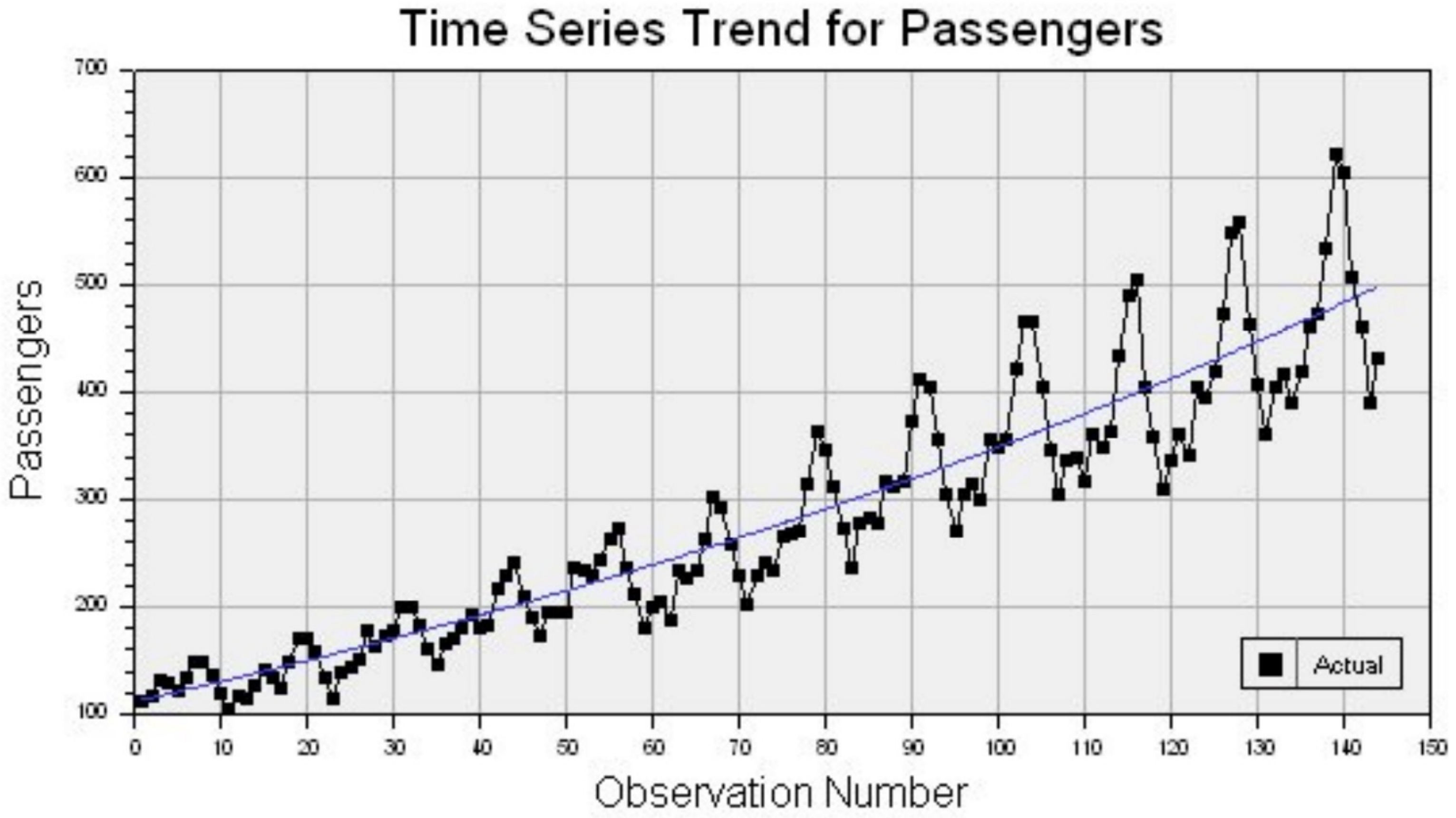
Events	Discrete events	Time Series
Report	Independent	Time ordered
Notification	Reporting state change	Context partitioned
Measurement	Actionable	Reporting conditions
Trace		Analyzable
Audit		

Examples

- Stock prices
- Weather reports
- Electricity demand
- Revenue numbers
- Temperature readings
- Number of passengers
- Criminality numbers

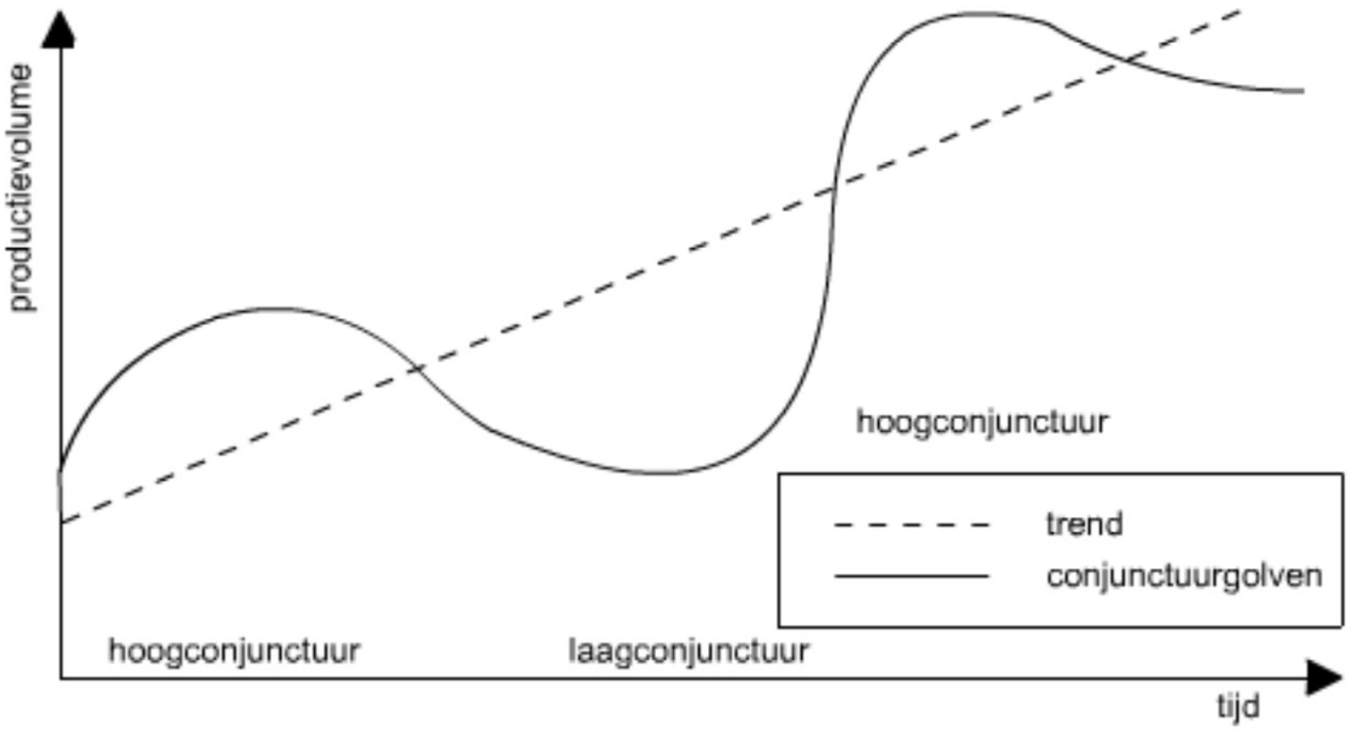
4 Time series components : Trend

- ▷ The long term variation



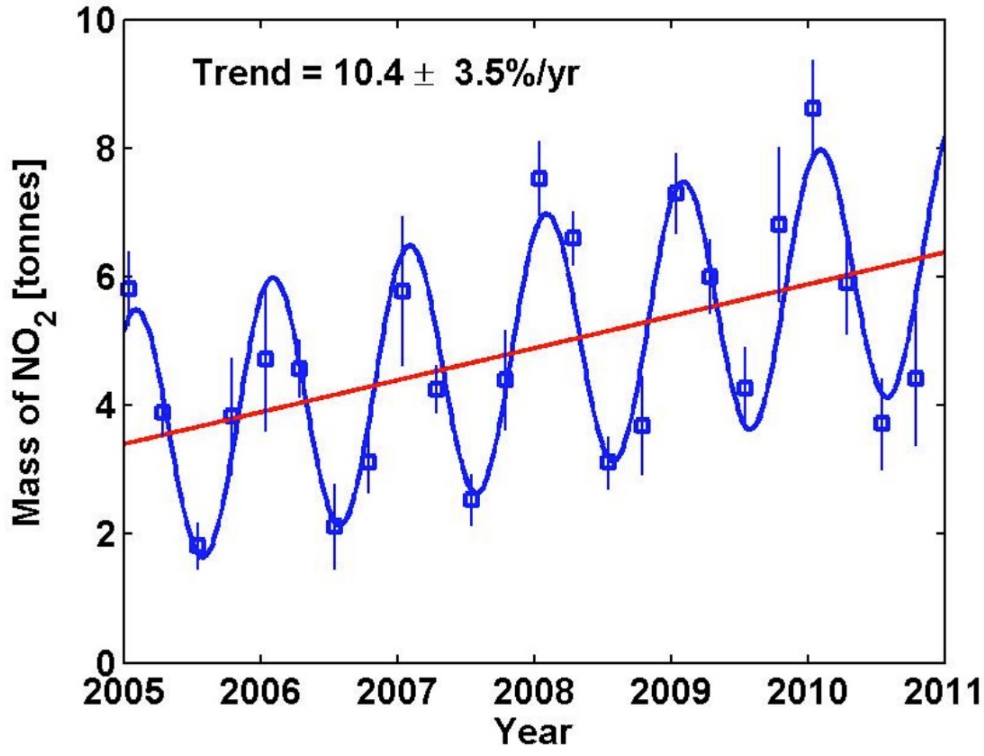
4 Time series components : Cyclical effect

- Fluctuations around the long term variation
- Typical impact of economic or political circumstances



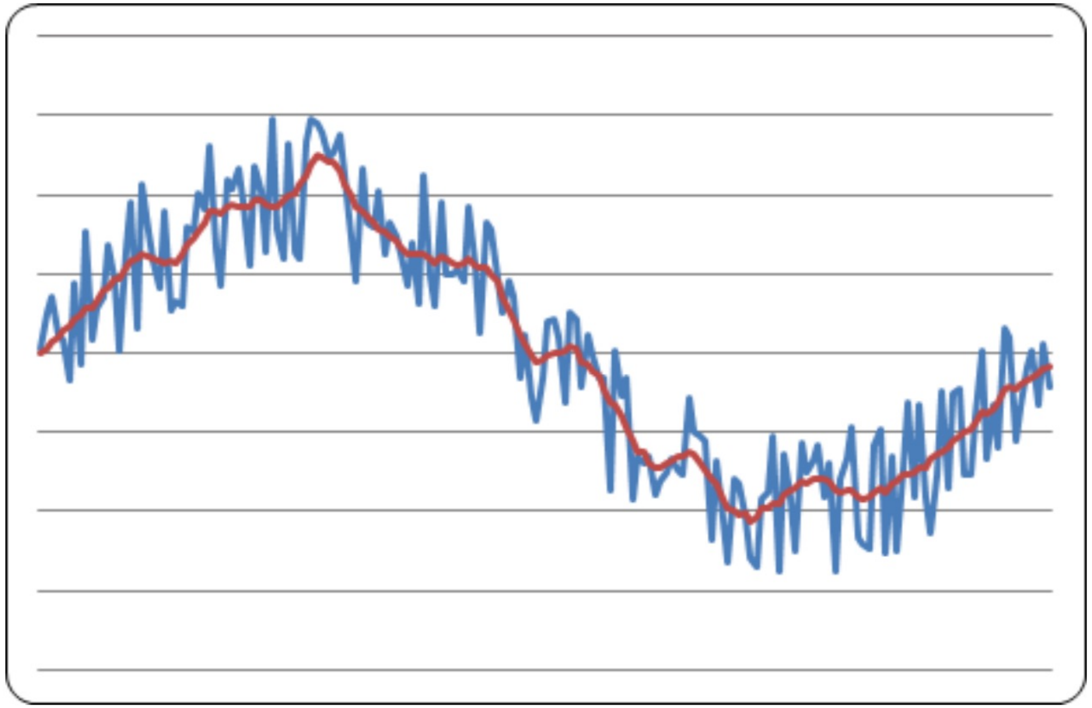
4 Time series components : Seasonality

- Variations that repeat over period
- Shorter periods in the longer dataset



4 Time series components : Residual effect

- Random (mostly smaller) variations without pattern
- Impact by external influences



Scenario: engine telemetry

Leveraging telemetry from engines
for predictive maintenance

Predictive maintenance data set

- ▷ Public dataset (Nasa Turbo fan)
  - o Damage propagation for aircraft engine
  - o Run-to-failure simulation
- ▷ Aircraft gas turbines
- ▷ Dataset contains time series (cycles) for all measurements of 100 different engines

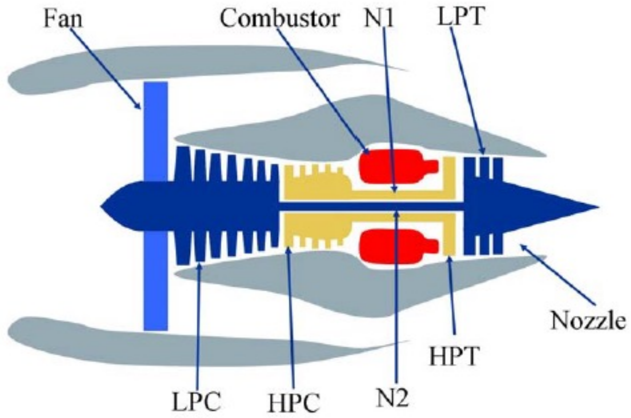


Figure 1. Simplified diagram of engine simulated in C-MAPSS [11].

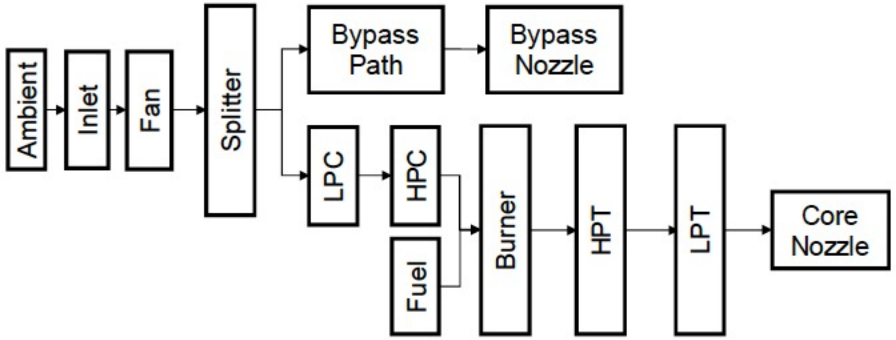


Figure 2. A layout showing various modules and their connections as modeled in the simulation [11].

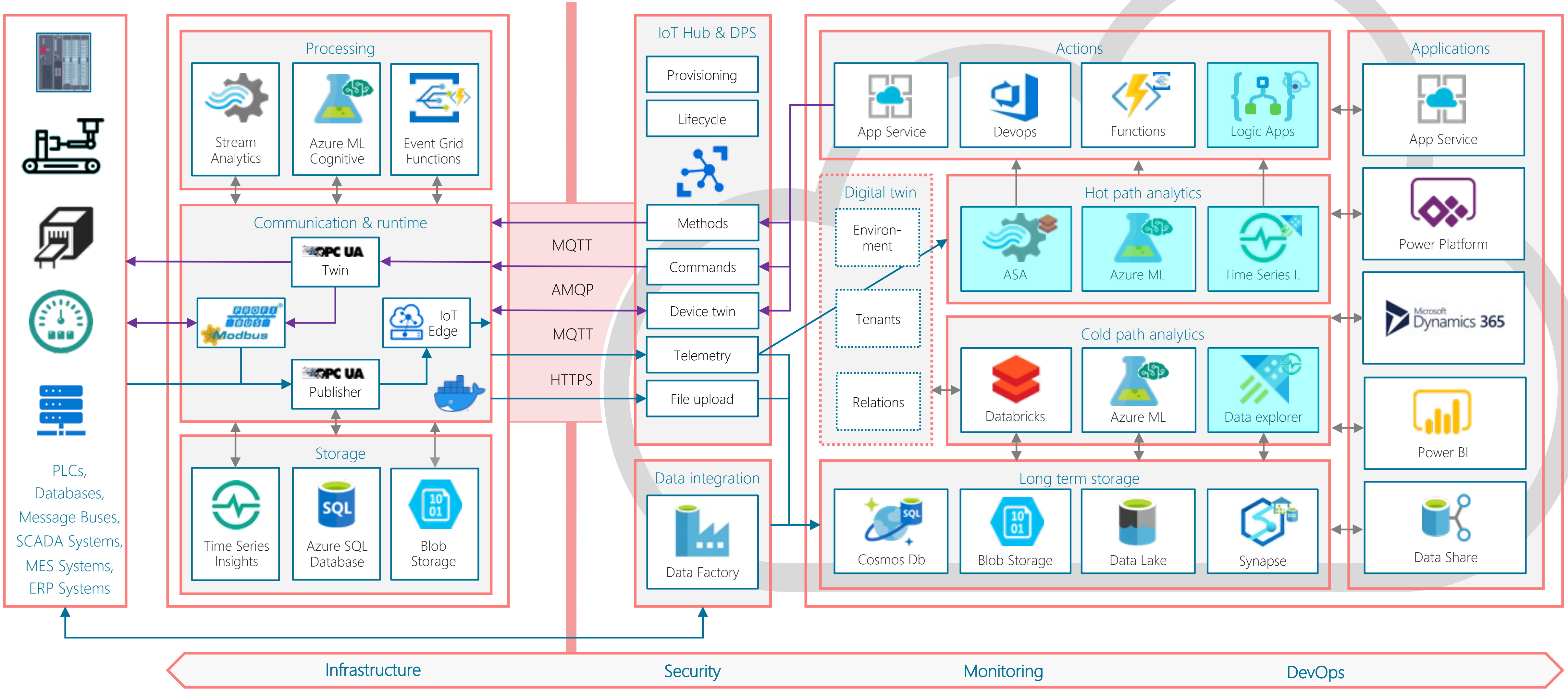
https://ti.arc.nasa.gov/tech/dash/groups/pcoe/prognostic-data-repository/#turbofan

Simulator module

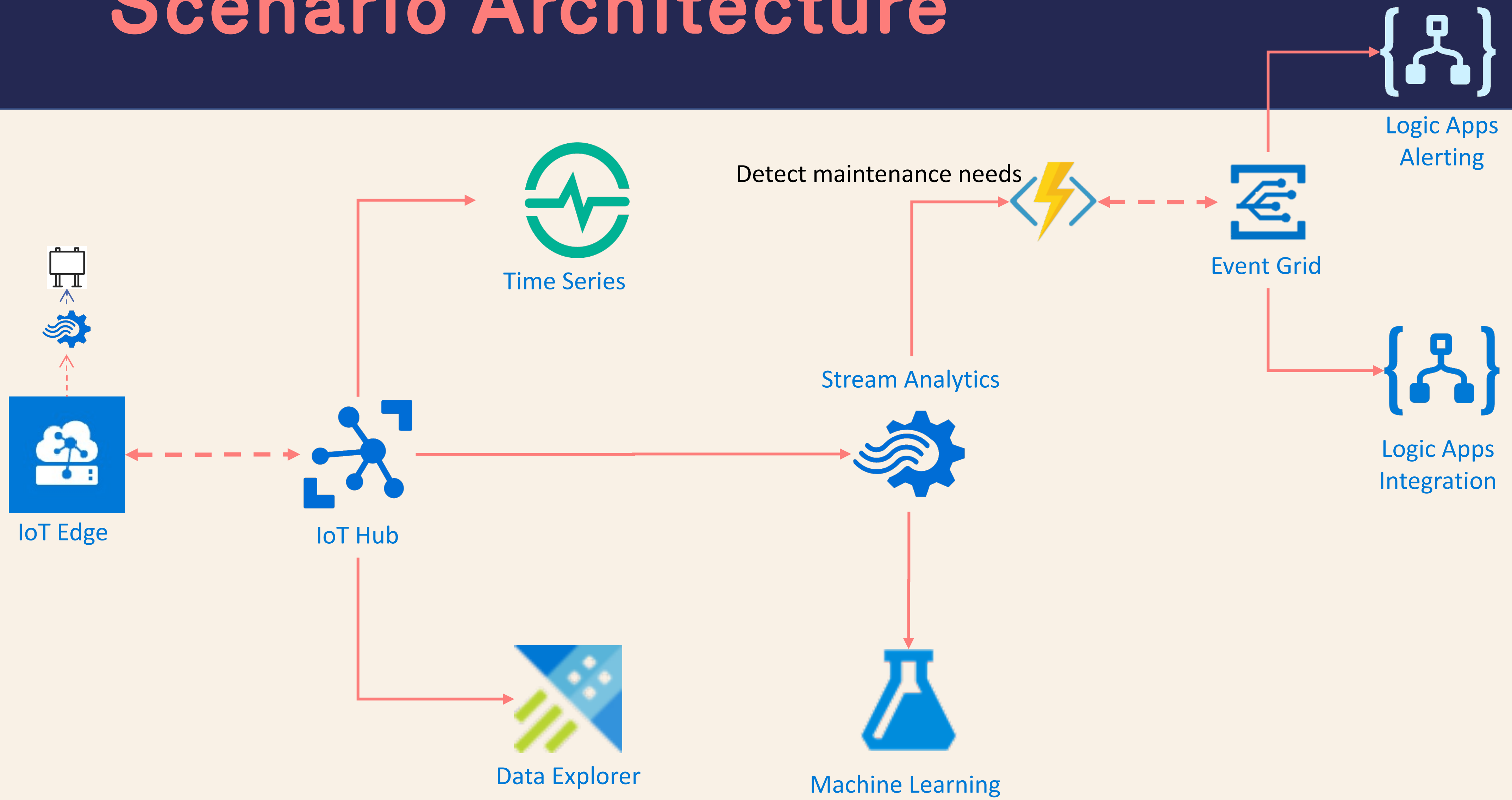
- Available for your use to test and play with time series data
  - Bring your own IoT Hub & start processing
- Documentation
  - https://github.com/SamVanhoutte/azure-time-travel/blob/main/docs/simulator.md
  - `docker run --rm -it -e iothub__IoTHubOwnerConnectionString=xxx savanh/engine-module:latest`

credits: Codit

IoT Reference Architecture



Scenario Architecture



Microsoft Azure Time Series Insights

Fast lane to Time Series data exploration

Azure Time Series Insights

Fully managed E2E PaaS solution to
ingest, process, store and query
highly contextualized,
time-series-optimized
IoT-scale data

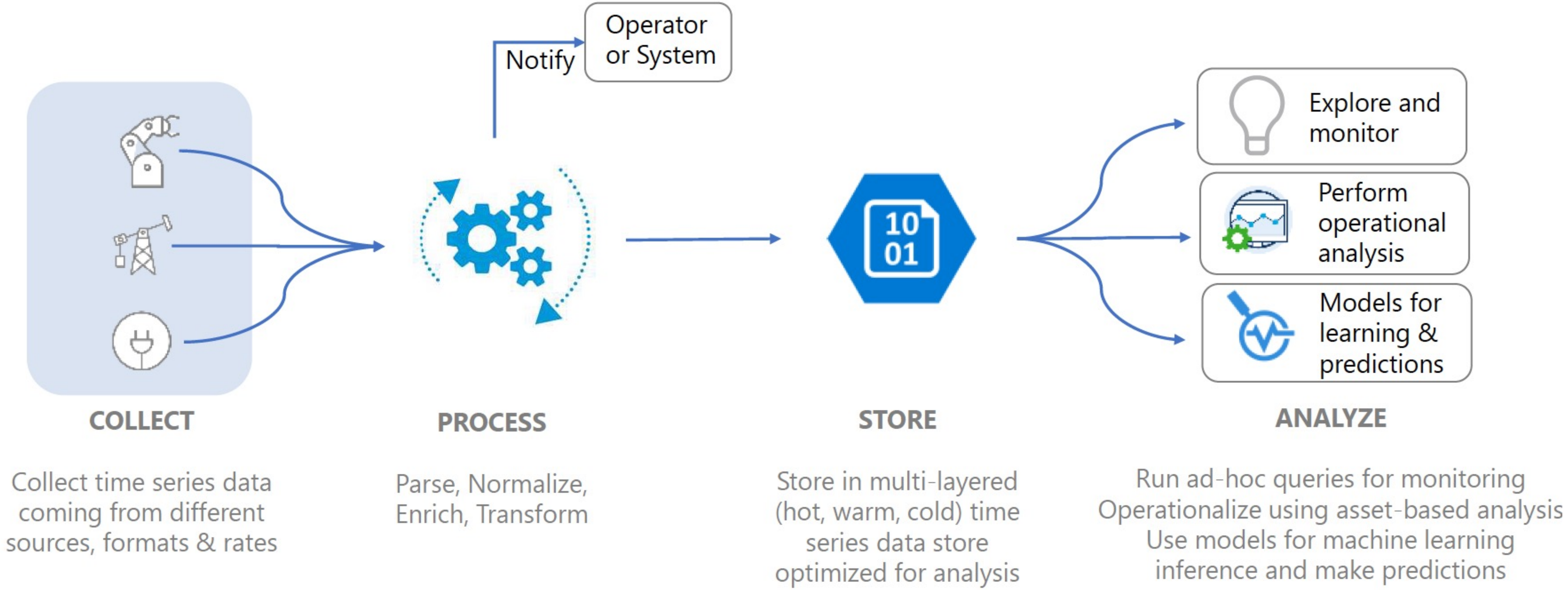
IoT data **lacks structural consistency**

IoT data **needs contextualization**

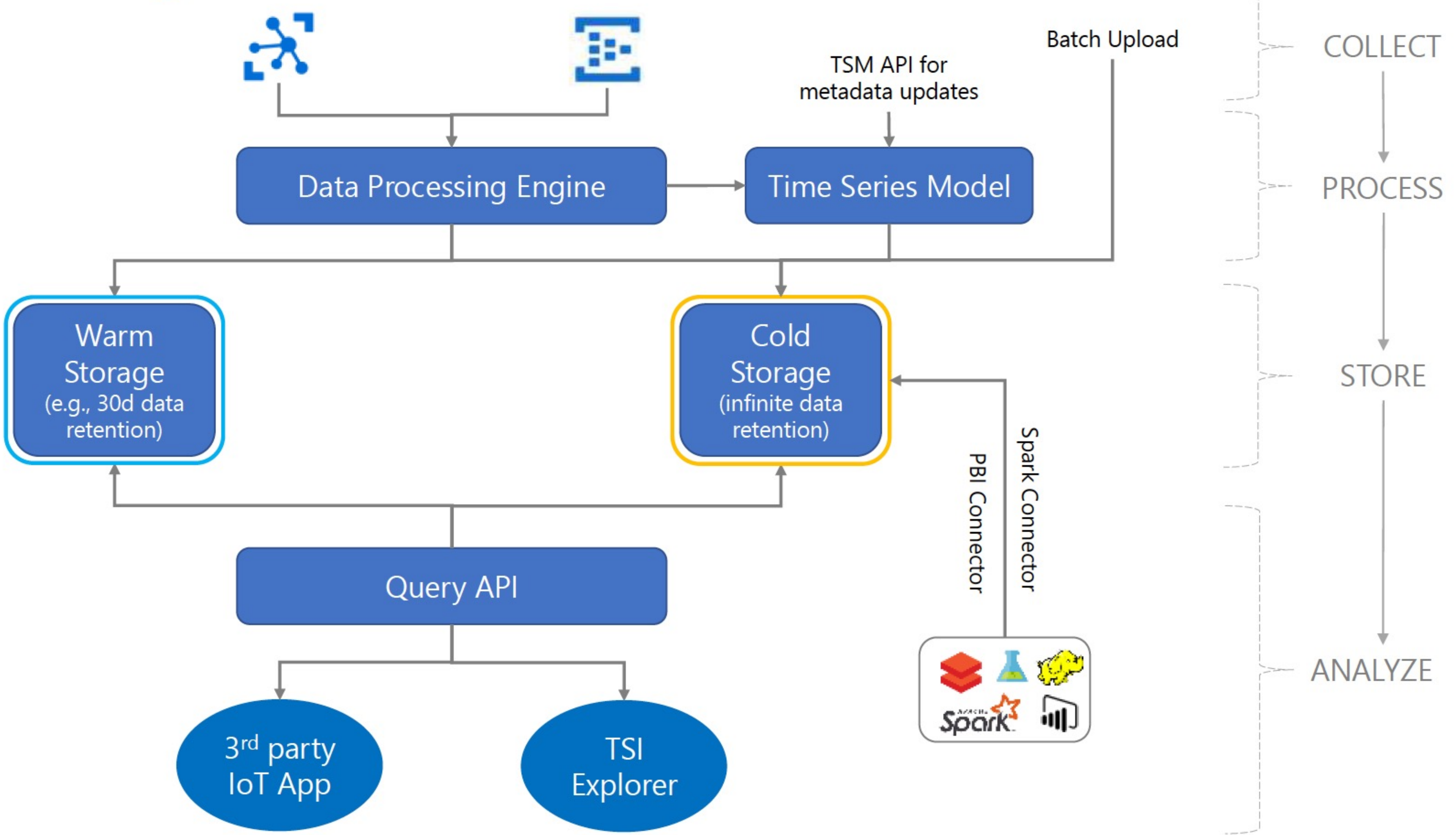
IoT data is **used with other data** from 1st or 3rd party sources

IoT data is **characterized by infinite retention**

Canonical Industrial IoT Data pipeline

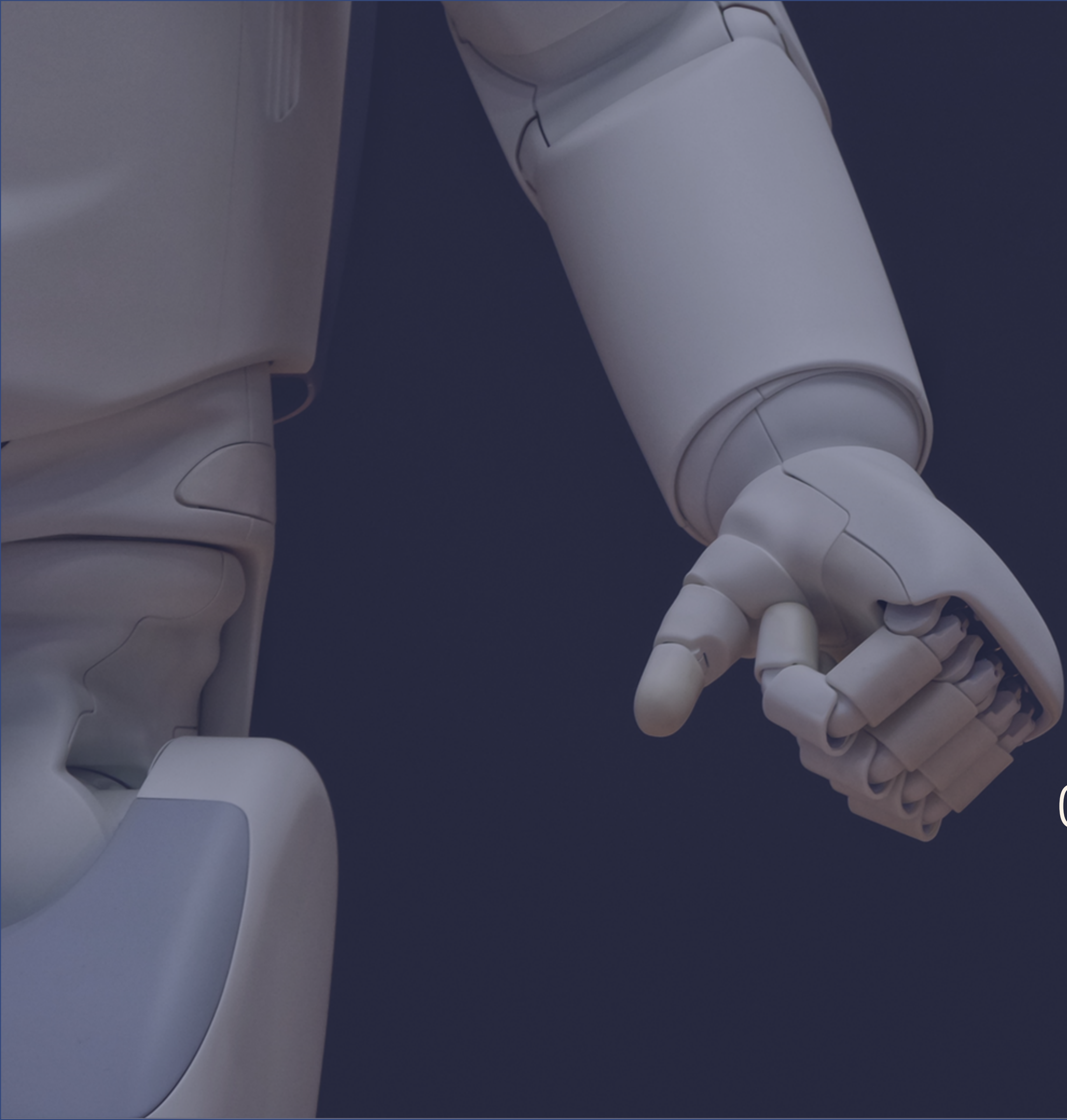


Architecture & setup



Metadata

- Types:
  - Calculations & aggregates over raw telemetry
  - Tags for sensors
- Hierarchies
  - Structure of the assets
  - Builds > Rooms > Devices
- Instances
  - Device metadata (links to 1 type definition & multiple hierarchies)



Azure Time Series Insights

Data Exploration

```
C:\Users\samva>demo.exe_
```

All samples available on:

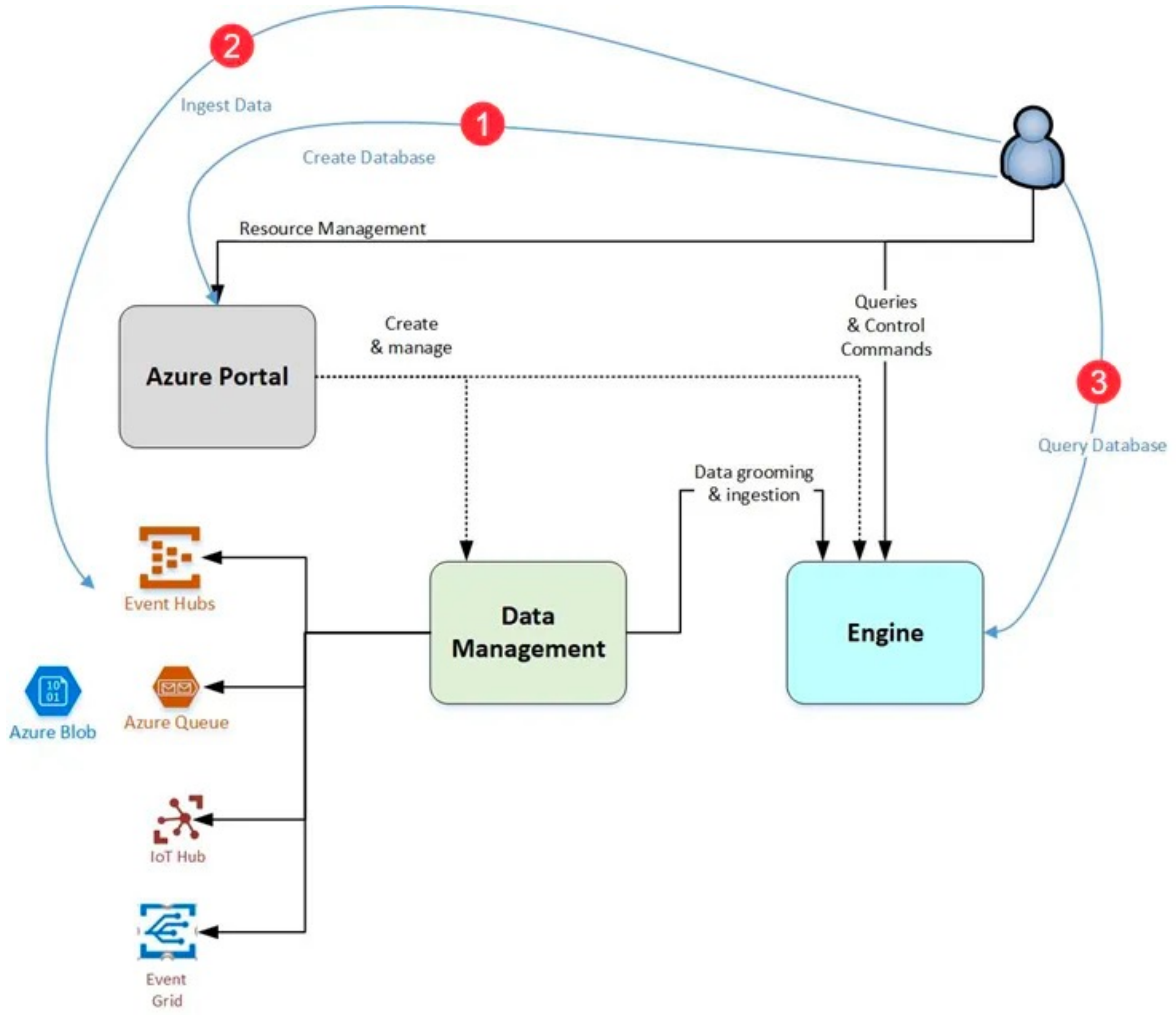
▷ https://github.com/SamVanhoutte/azure-time-travel

Azure Data Explorer

Heavy lifting your Time Series Data

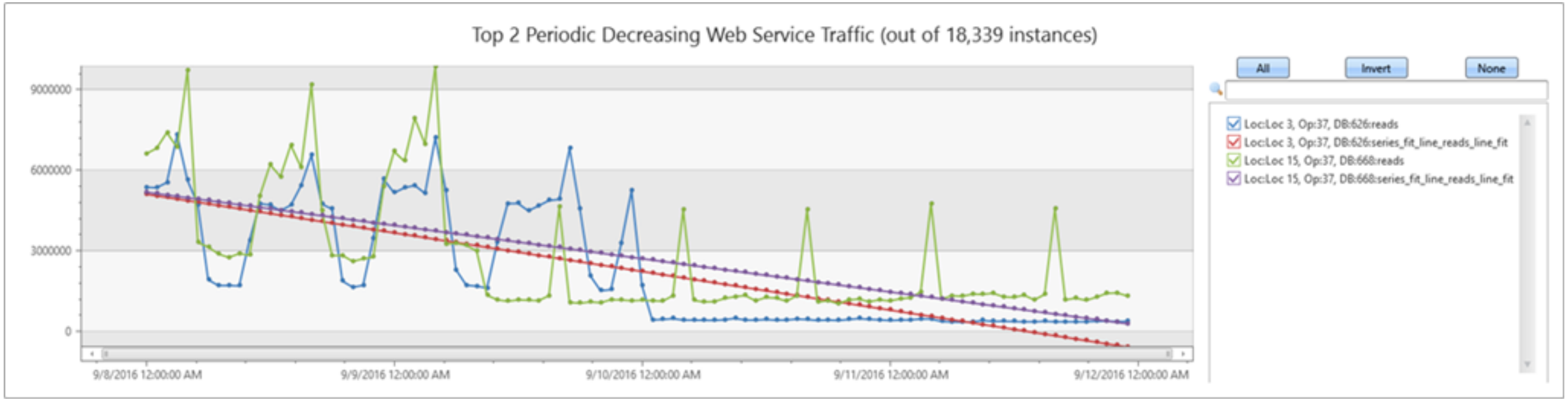
ADX Flow

- Create database & cluster
  - Dedicated cluster with 1:+ databases
  - Autoscaling supported
- Ingest data
  - Streaming / batch sources
  - Destination table & mapping
- Kusto Explorer or API
  - KQL (filtering, aggregation, grouping, joining)
- Visualization
  - Render command in KQL
  - Power BI integration
  - Python plugin / notebooks



Time series analytics in ADX

- Filtering (noise reduction, smoothing, change detection)
- Regression analysis (trend change detection)
- Seasonality support
- Element-wise functions (perform operations between time series)



Time series functions in ADX

▷ `make-series` > turns numeric values, into series by timestamp & group

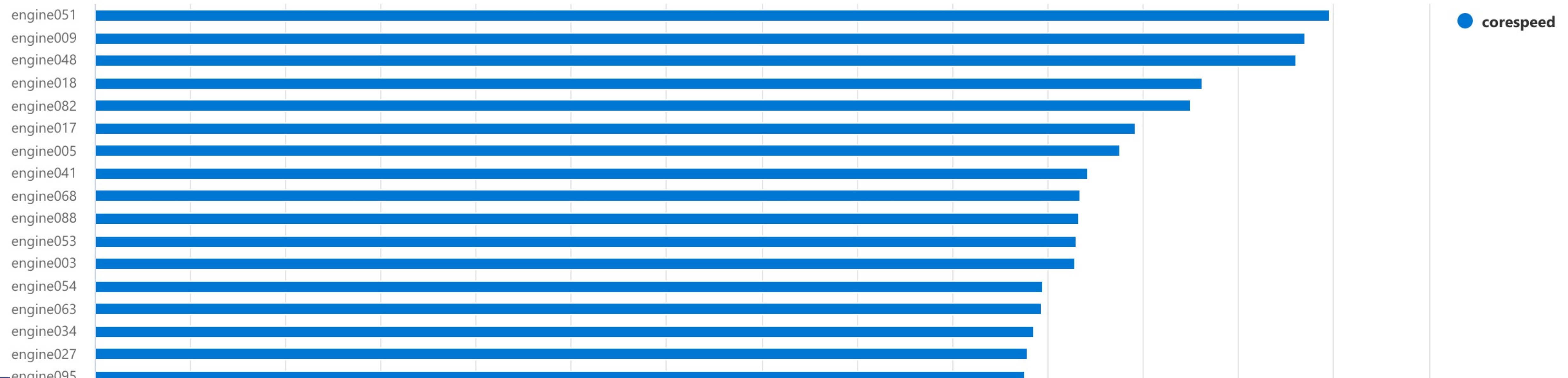
▷ `series_decompose(reads)` > detect seasonality, trend & residual
▷ `series_seasonal(reads)` > detect seasonality
▷ `series_periods_detect(reads)` > detect periodicity
▷ `series_outliers(reads)` > detect outliers

▷ `series_decompose_forecast(reads)` > forecast time values

[Time Series Analysis - Azure Data Explorer | Microsoft Docs]()

Basic Query

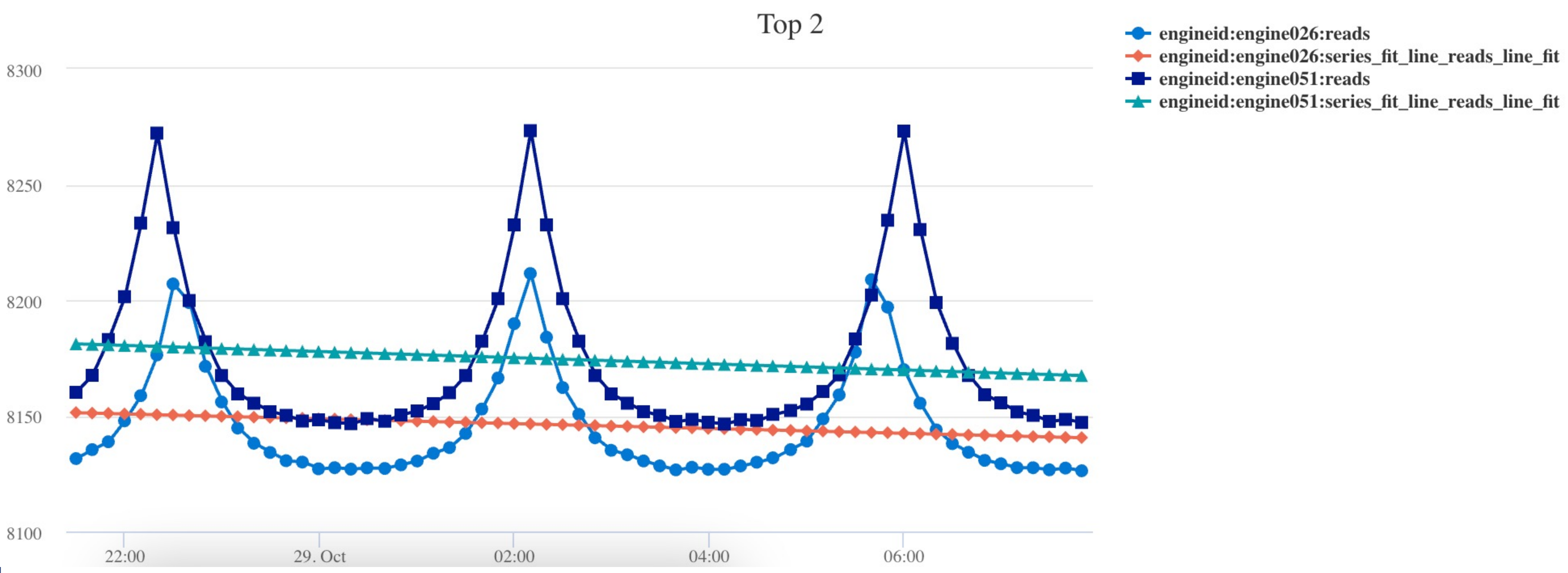
```
telemetry
| summarize corespeed=avg(physical_core_speed)by engineid
| top 25 by corespeed
| render barchart
```



Time decomposition

```
let min_t = ago(90m);
let max_t = ago(10m);
telemetry
| make-series reads=avg(corrected_core_speed) on eventtime in range(min_t, max_t, 10m) by engineid
| extend (p, ps)=series_periods_detect(reads, 0, 24, 1)
| mvexpand p to typeof(double), ps to typeof(double)
| extend series_fit_line(reads)
| top 2 by series_fit_line_reads_slope asc
| render timechart with(title='Top 2')
```

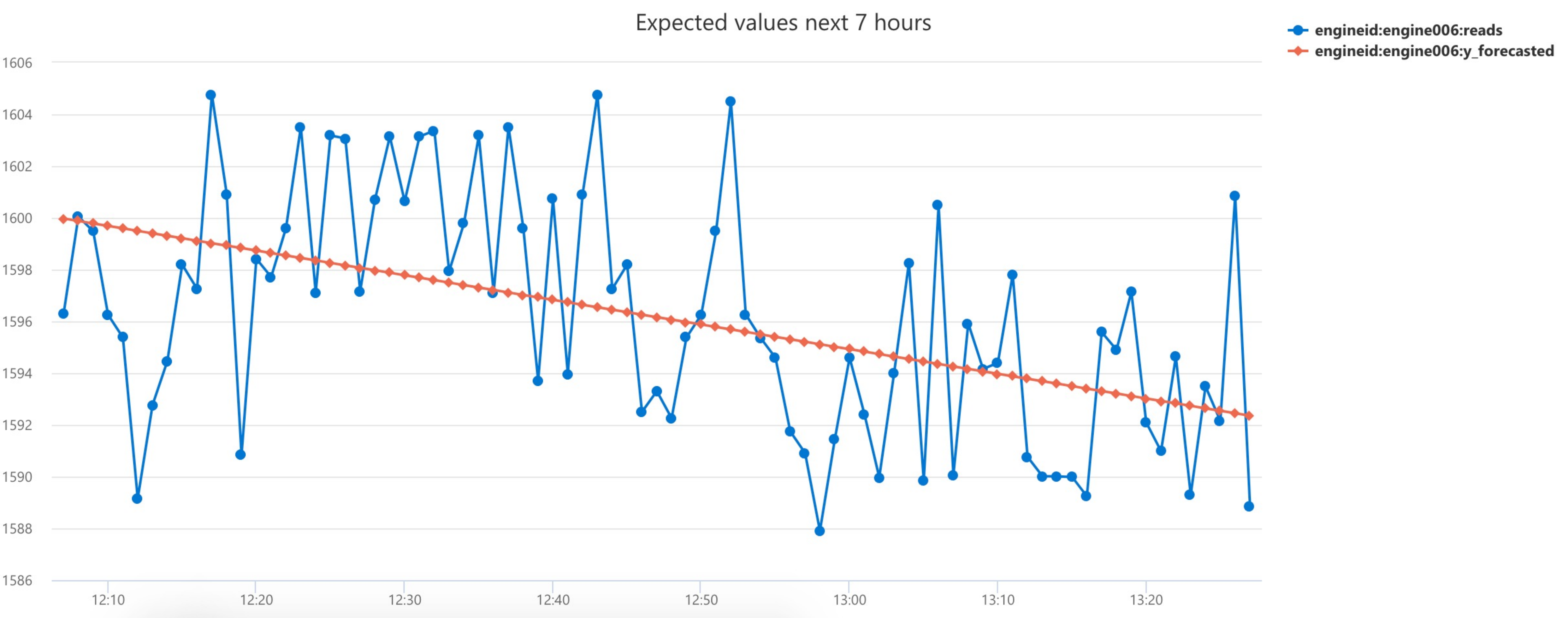
Time decomposition



Time forecasting

```
let min_t = ago(90m);
let max_t = ago(10m);
telemetry
| where engineid == 'engine006'
| make-series reads=avg(temp_hpc_outlet) on eventtime in range(min_t, max_t, 1m) by engineid
| extend y_forecasted = series_decompose_forecast(reads, 7) // forecast 7 hours forward
| render timechart with(title='Expected values next 7 hours')
```

Time forecasting



Azure Data Explorer

KQL In Action

C:\Users\samva>demo.exe

All samples available on:

▷ https://github.com/SamVanhoutte/azure-time-travel

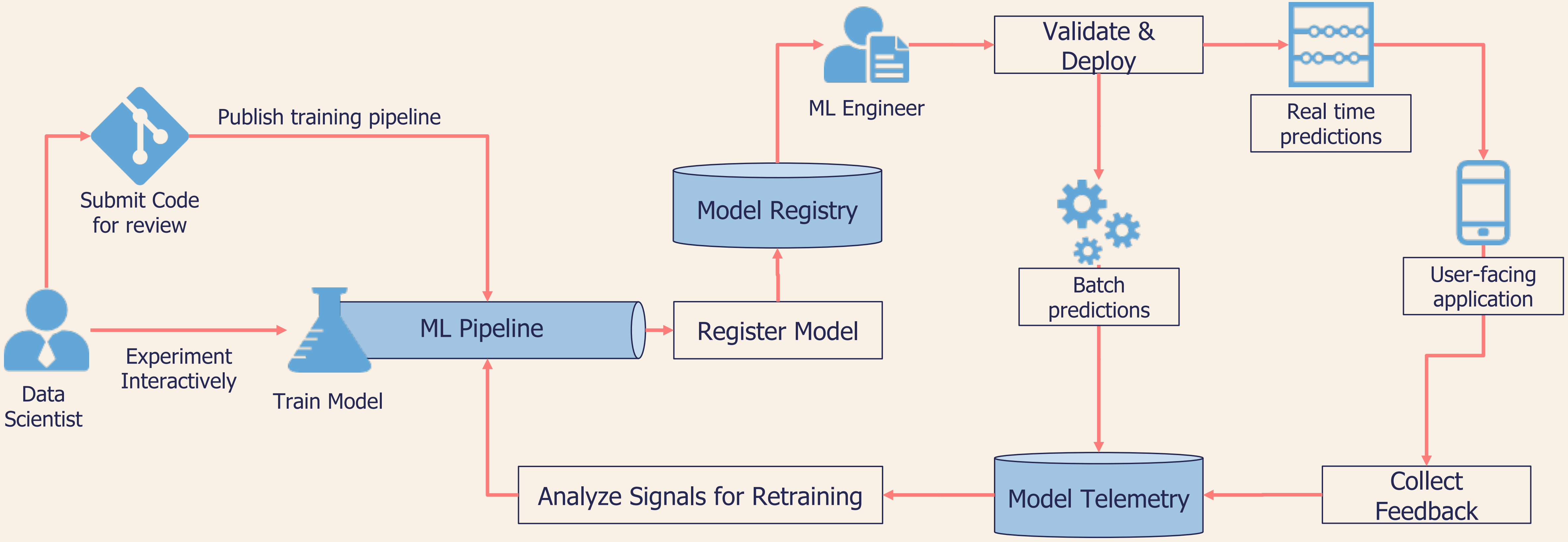
Predictions through Machine Learning

Model training with Azure ML

Time Series Predictions

- Not always need for Machine/Deep Learning
- Statistical algorithms available:
  - Moving Average: flatten out peaks and take average of last n days
  - Auto Regression: take n previous values and apply same linear function on it
  - ARIMA: Autoregressive Integrated Moving Average (combines above)
  - SARIMA: Seasonal ARIMA, considers seasonality
- Machine learning / Deep learning algorithms
  - Prophet: Facebook algorithm, targeted to seasonality
  - LSTM: Recurring Neural Network with memory gate (see further)

The MLOps process



Reusability & standardization

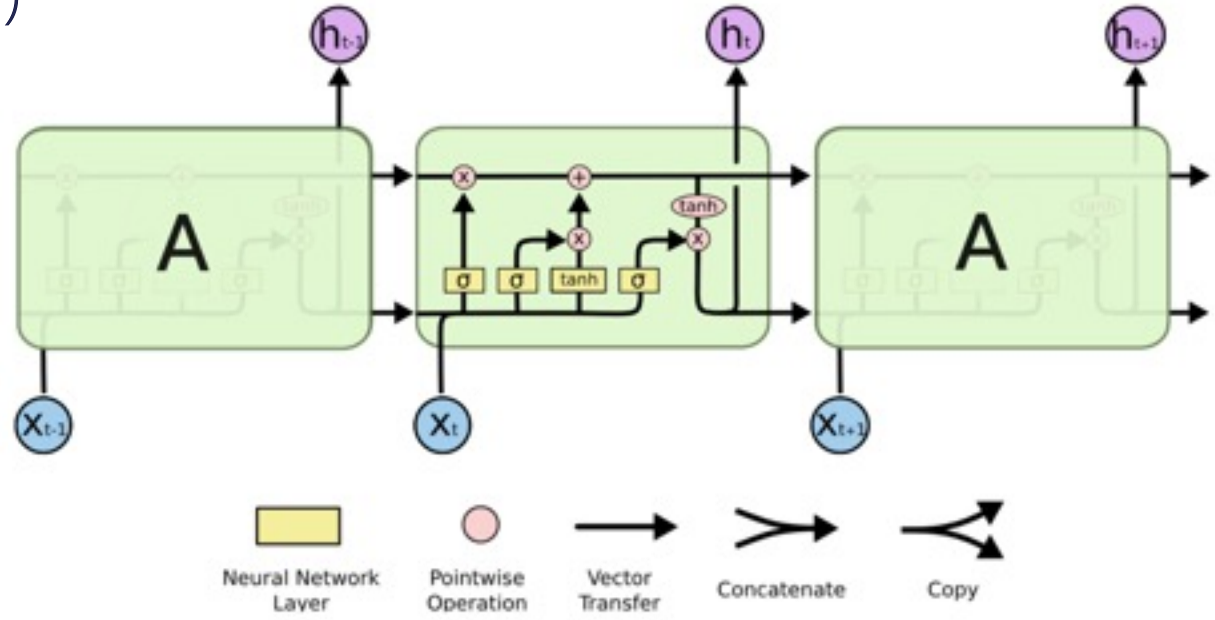
- Open source effort for the community
  - Arcus for Machine Learning
    - Common operations for DataFrames
    - Images
      - IO: load from dataframe, Load from Url, Slice Images, Caching, Transform
      - Explorer: Visualize Images, Compare images (expected/In/out)
    - Time Series: Time Windowing
  - Arcus for Azure Machine Learning
    - Interactive experimenting (notebook), with centralized logging & snapshotting
    - Cloud training & experiments, with standard templates, logging and evaluations

Training approach

- Experiment locally (Jupyter Notebook)
  - Smaller datasets
  - Focus on syntax and feasibility
  - Immediate feedback, short training cycles
- Script generation
- Schedule trainings in the cloud
  - Hyper parameters as script arguments
  - Multiple runs in parallel, through scheduling queue by AzureML
  - Docker image generation, based on required packages and sizing needs
  - Every experiment gets logged & tracked

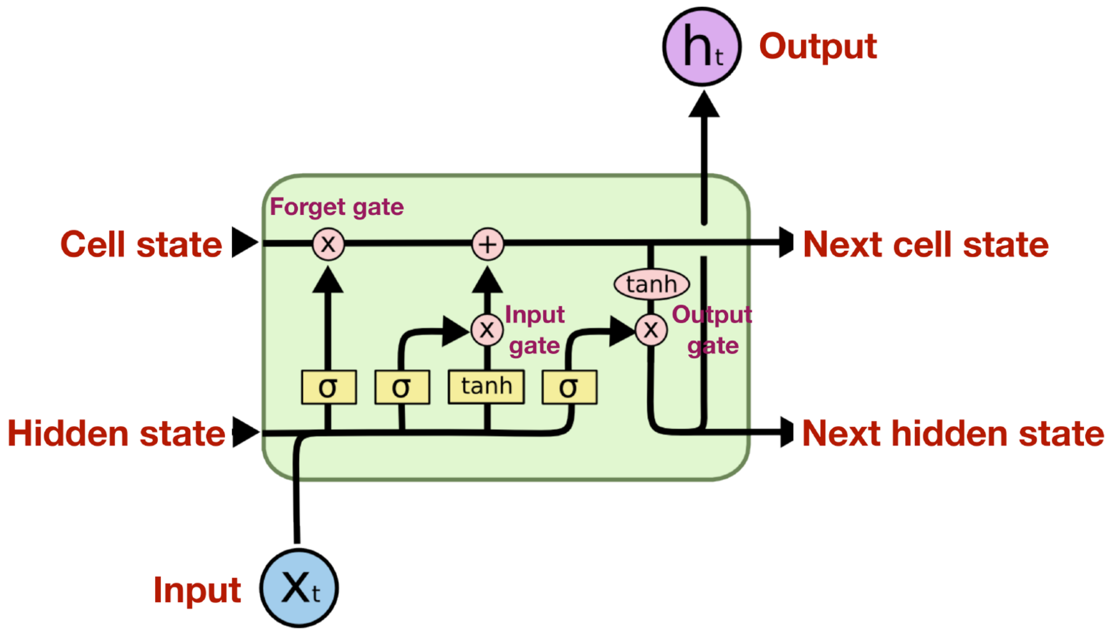
Long Short Term Memory network (LSTM)

- An advanced type of RNN (Recurrent Neural Network)
- Common scenarios
  - Text prediction
  - NLP (Natural Language processing)
  - Music generation
  - Time series prediction
- GRU (Gated Recurrent Unit):
  - Lightweight version



LSTM Cell

- Cell state: long term memory
- Forget gate: define what to erase from memory
- Input gate: define which new information should be added to cell state
- Output gate: define which data will be passed on as input to the next cell



Time series for predictive maintenance

- Signals can be treated independently
  - Missing history and lead up
  - Logistic regression 92.56% (gridsearched)
  - Random Forest Classifiers 92.98% (gridsearched)
  - Vanilla Neural Network 91.20% (not tuned)
- Signals can be grouped in time windows
  - History and trends can be taken into account
  - LSTM 93.12% (with window length of 30)
  - CNN 92.78%

Impact on window length

- The larger the window length (dimensions for network input), the higher the accuracy:
  - Window length 15: 86.3%
  - Window length 30: 92.96%
  - Window length 50: 95.10%

- In reality, the window length will be defined based on
  - The ability to keep previous cycles available for model input
  - The overfitting risk

Azure Machine Learning

Predictive Modeling

```
C:\Users\samva>demo.exe_
```

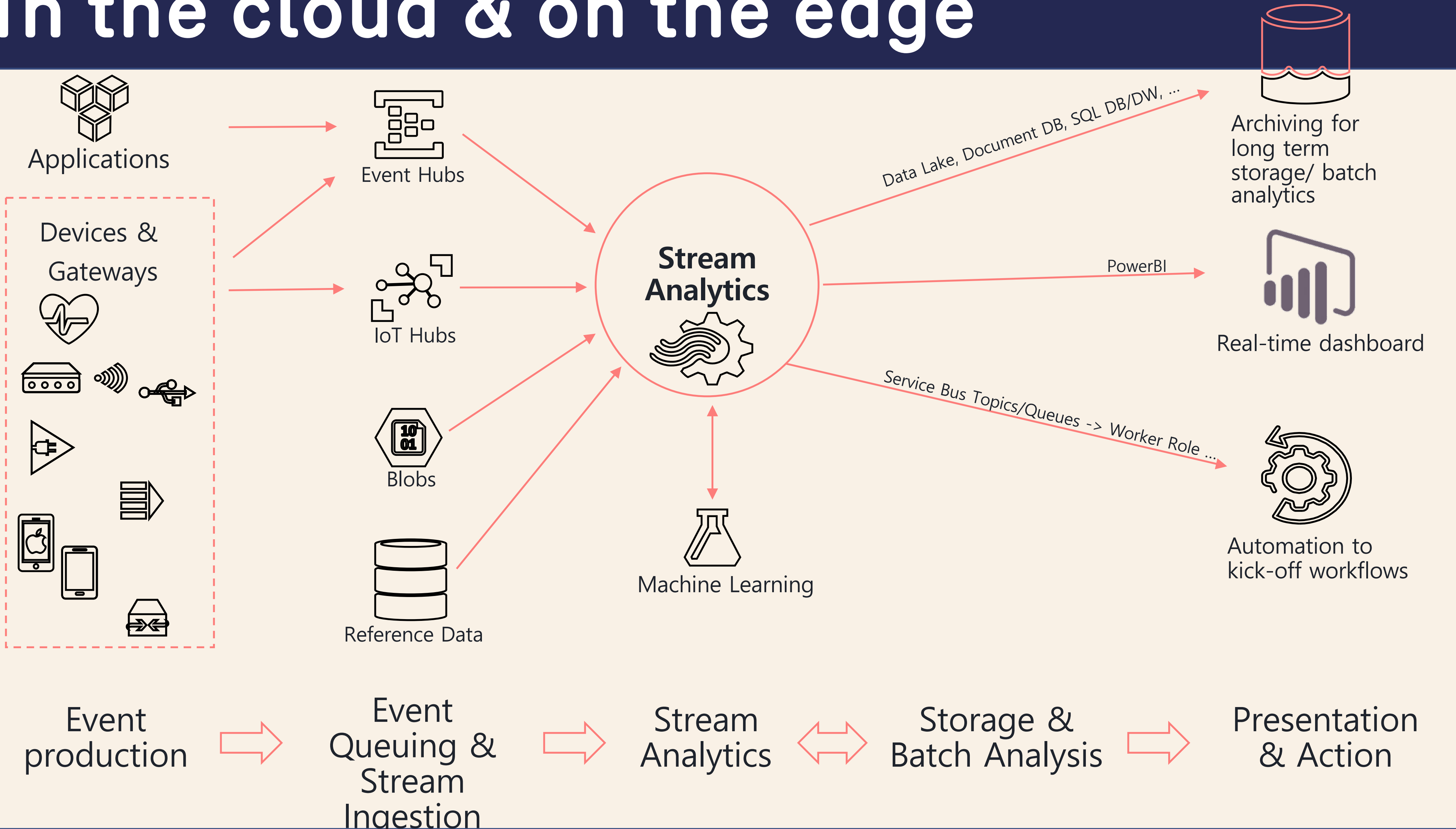
All samples available on:

▷ https://github.com/SamVanhoutte/azure-time-travel

Azure Stream Analytics

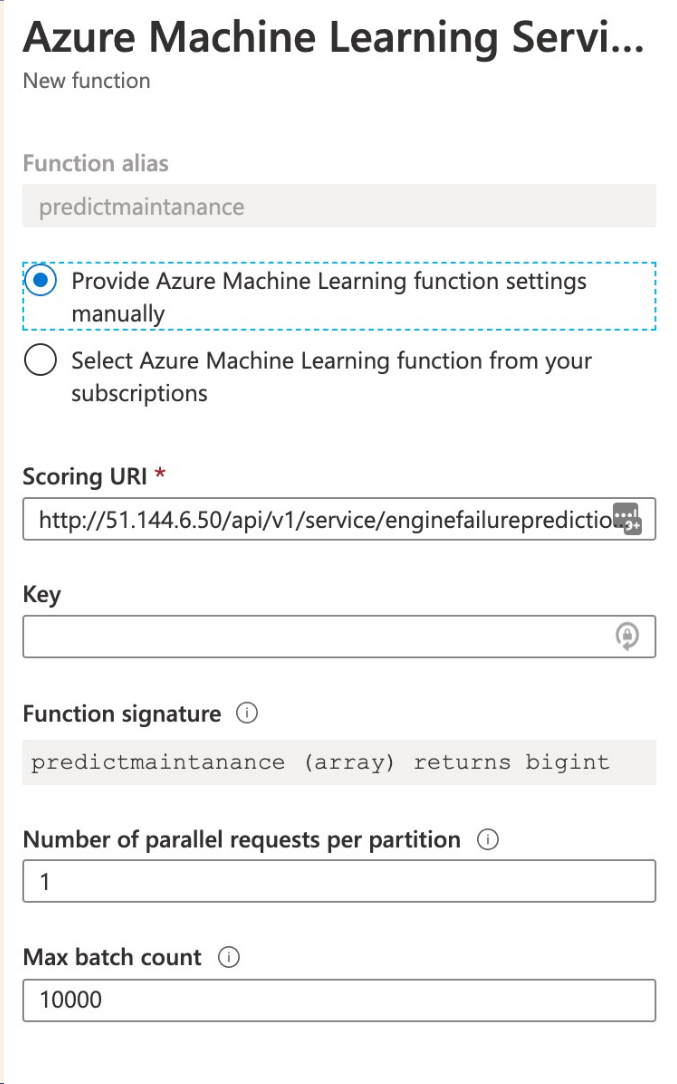
Leverage time windowing in the telemetry stream

Azure Stream Analytics
In the cloud & on the edge



Calling AzureML models from A.S.A.

- Creation of a new UDF (user defined function)
- AzureML endpoint requirements (example)
  - Swagger metadata should be exposed
  - Input data types supported
    - int, str, numpy, pandas, pyspark



Specifics to call the ML Model from A.S.A.

- Taking last 30 records for every engine (window length)

```
SELECT engineid,
CollectTop(30) OVER (ORDER BY eventtime DESC) as Windows
```

Specifics to call the ML Model from A.S.A.

- Transforming to multi dimensional array (Javascript U.D.F.)

```
// creates a N x 18 size array
function createTimeWindows(engine_windows) {
        'use strict';
        var result = [];
        var output = [];
        for(var window_id in engine_windows){
                var array = [];
                array.push(engine_windows[window_id].value.temp_fan_inlet);
                // Add every relevant field
                output.push(array);
        }
        result.push(output);
        return result;
}
```

Azure Stream Analytics

Streaming processing & predictions

```
C:\Users\samva>demo.exe_
```

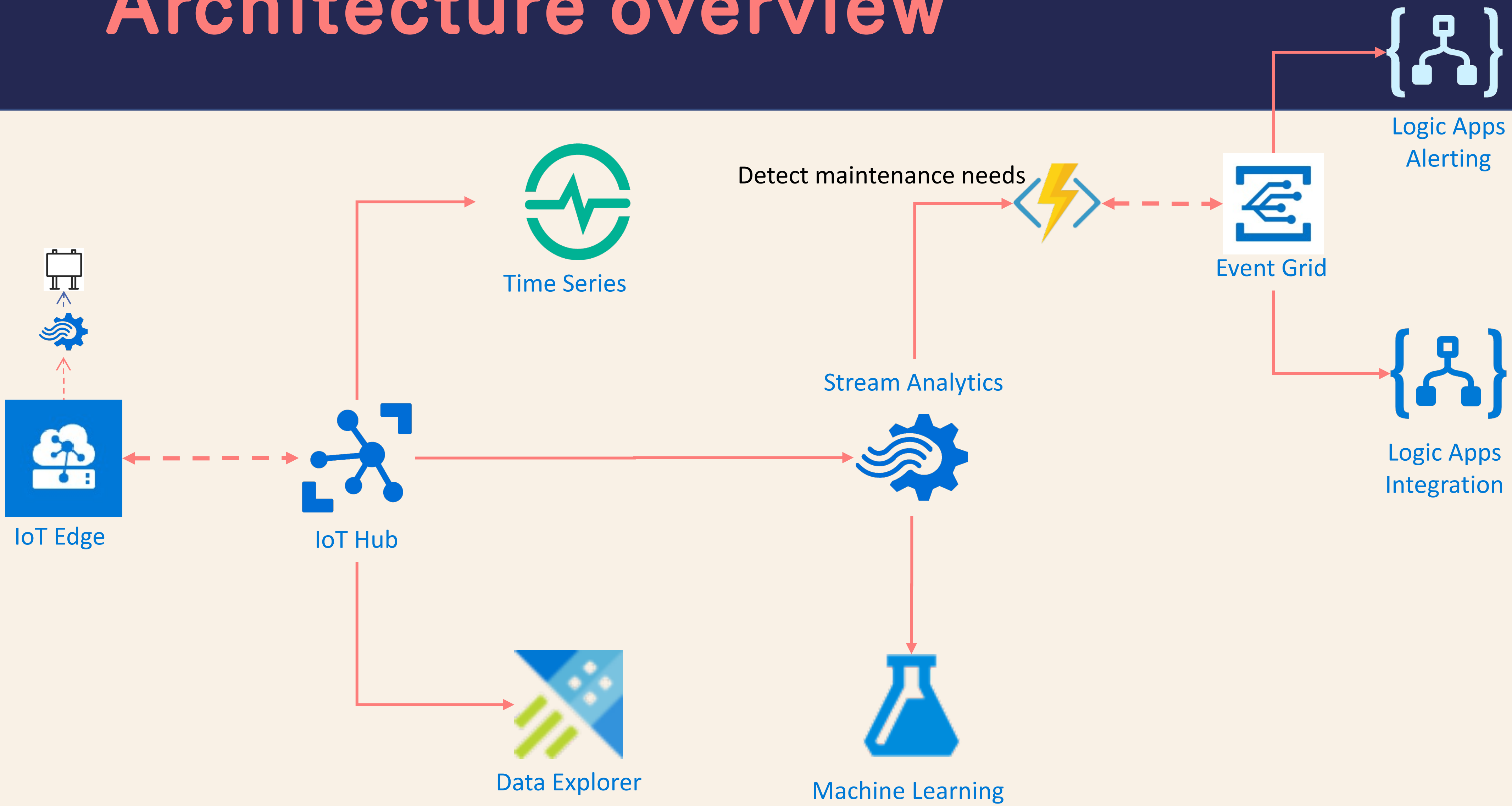
All samples available on:

▷ https://github.com/SamVanhoutte/azure-time-travel

The integration of everything

Stitching things together

Architecture overview



Takeaways

Azure offers plenty options for Time Series processing

- Ingest data into Time Series Insights
- Enable Data exploration, querying and visualization
- Extend to Machine Learning, Data Science and Front End applications
- Out of the box integration with Data Lake, Power BI, etc

Thank you
let’s connect

twitter.com/SamVanhoutte

github.com/SamVanhoutte

All samples available on:

▷ https://github.com/SamVanhoutte/azure-time-travel