AWS re:Invent

IOT309-R

Combining IoT and machine learning for predictive maintenance

Ryan Burke
Senior Solutions Architect
Amazon Web Services

Ashish Bajaj
Software Development Engineer
Amazon Web Services

AWS re:Invent

aws

Agenda

Introduction to predictive maintenance

Why IoT + ML is hard

Technical demonstration

Summary

Related breakouts

IOT314-R - Driving operational performance across industrial facilities with AWS

IOT339 - Transforming automotive manufacturing with Volkswagen

IOT311-R - Combining camera feeds, edge computing, and ML for remote monitoring

IOT331-R - Integrating industrial robots with the cloud

Let’s talk

Join us in the IoT Networking Lounge at The Quad (ARIA) on Wednesday, December 4, from 8 a.m. to 6 p.m.

Introduction to predictive maintenance

AWS re:Invent

aws

Definition of predictive maintenance

Monitor

- Performance and condition of equipment during operation

Predict

- Equipment remaining useful life (RUL)
- Schedule when maintenance should be performed

Alert

- When maintenance is due
- When high possibility of equipment malfunction

Let's study an example

Industrial air filter

Adjust parameters based on
air pollution levels

Daily manual measurement and
adjustment

Franco Nadalin / Shutterstock.com

Problems that we observe in last example

Highly manual

- Errors in measurement could impact RUL
- Requires human time, may conflict with other priorities

Cost inefficient

- Higher offsets could lead to increased power consumption
- Sudden changes in pollution level could lead to reduced equipment life

Scheduled maintenance

- Equipment scheduled on a clock, not when required or optimal

Value of predictive maintenance

Reduce spend on unnecessary maintenance

- Schedule maintenance when needed or most impactful

Remove dependency on manual effort

- Active monitoring of changing operating conditions
- Normalize measurement process to eliminate human error

Reduce unplanned downtime

- Automated decision-making can prevent malfunctions from unsafe operation

What we are building today

Predicting maintenance for industrial air filters

- Update filter parameters to maximize efficient use and prevent failure

Ingest data to AWS IoT Analytics

- Historical air pollution from dataset Beijing PM2.5 Data[1]

Train model with neural network

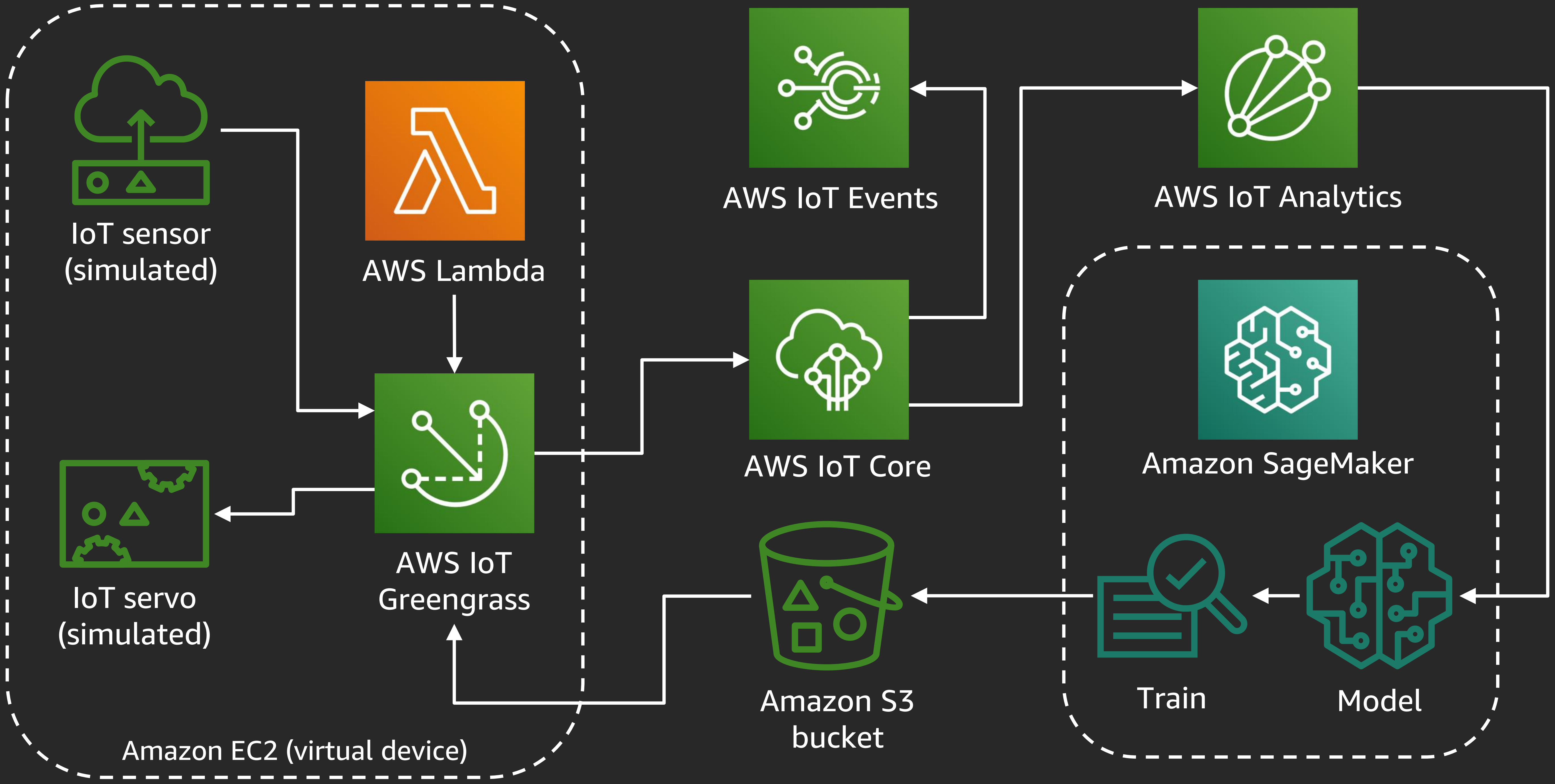
- Forecast air pollution through an LSTM model

Deploy model to edge with AWS IoT Greengrass

- Make inferences locally to minimize bandwidth and latency

1. https://archive.ics.uci.edu/ml/datasets/Beijing+PM2.5+Data

Architecture



Why IoT + ML is hard

Why IoT + ML is hard

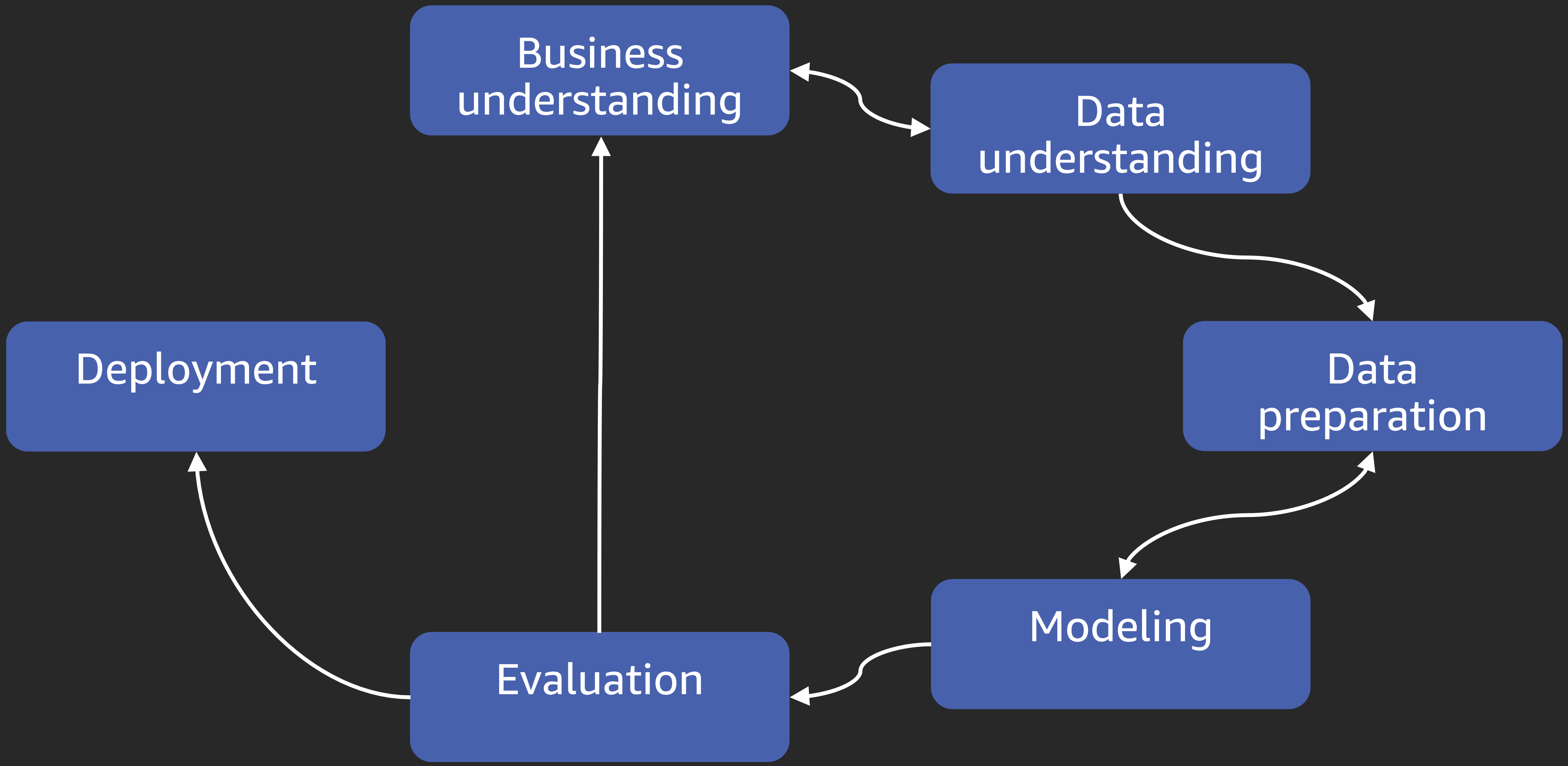
Level of effort

- Less “IoT + ML” and more “IoT x ML”
- Iterative process to reach goal, difficult to predict project duration

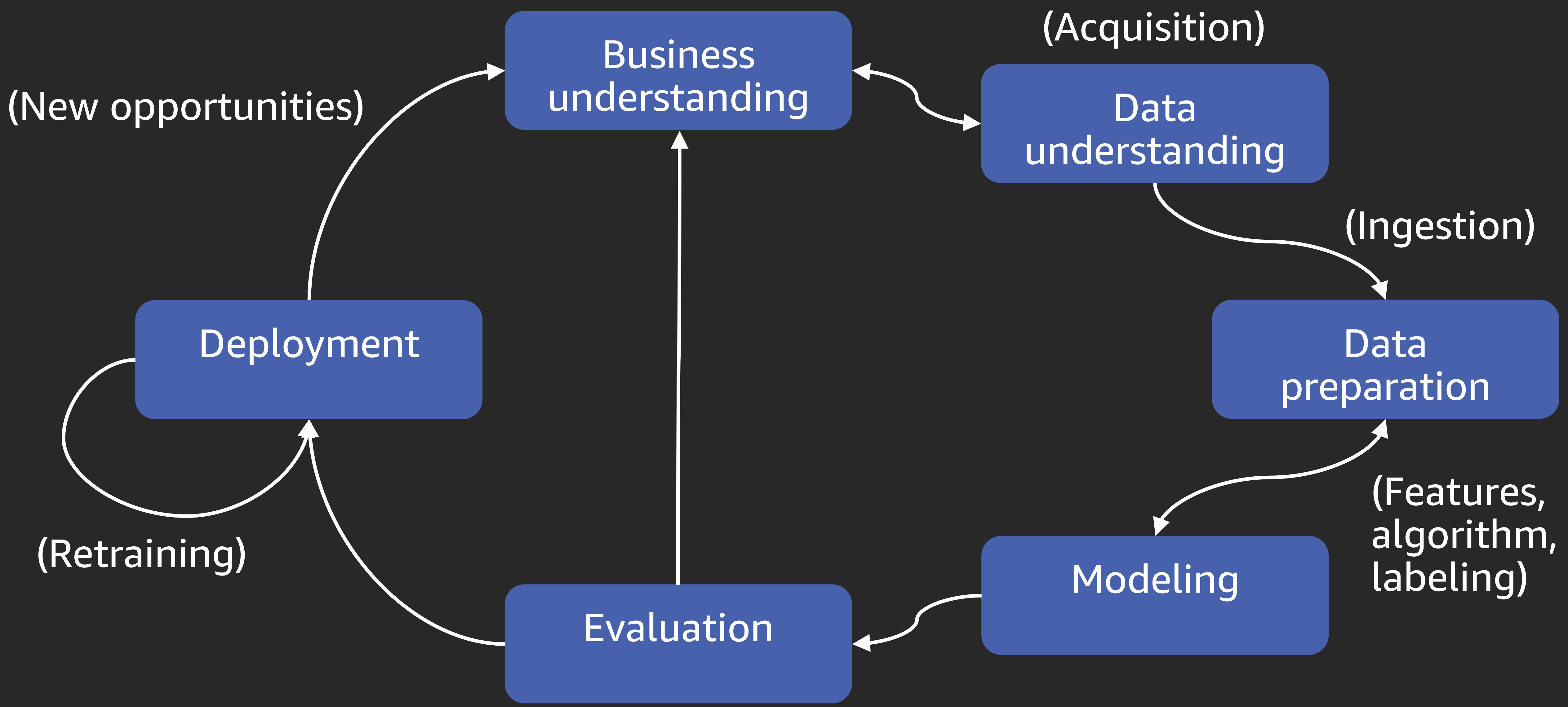
Personas involved

- Stakeholders required across IT, OT, business leadership
- Data engineer, process engineer, data scientist, domain expert, etc.
- If you can solo an IoT + ML project, you’re underpaid or doing it wrong

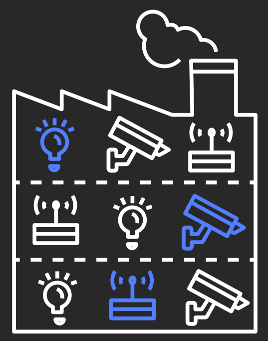
Cross-industry standard process for data mining



IoT + ML flywheel



Challenges of IoT

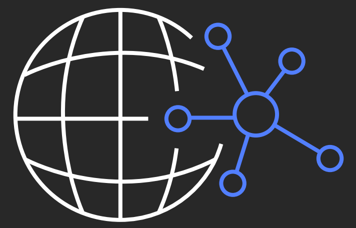


Data acquisition

Where is the data and how to collect it

Data ingestion

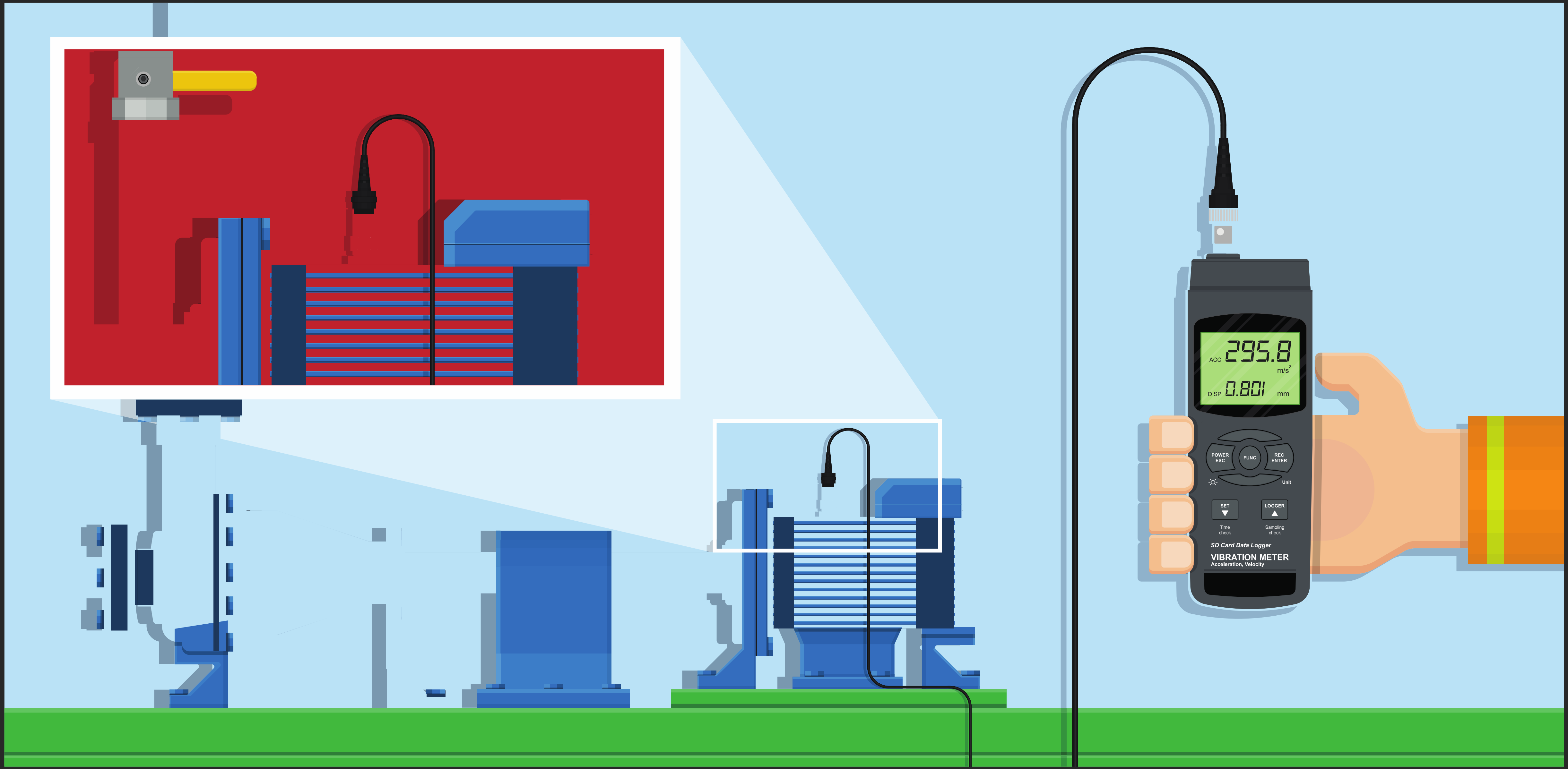
How to move industrial data to the cloud



Model deployment

Finding the optimal host for model inference

Data acquisition



rumruay / Shutterstock.com

Data acquisition in this solution

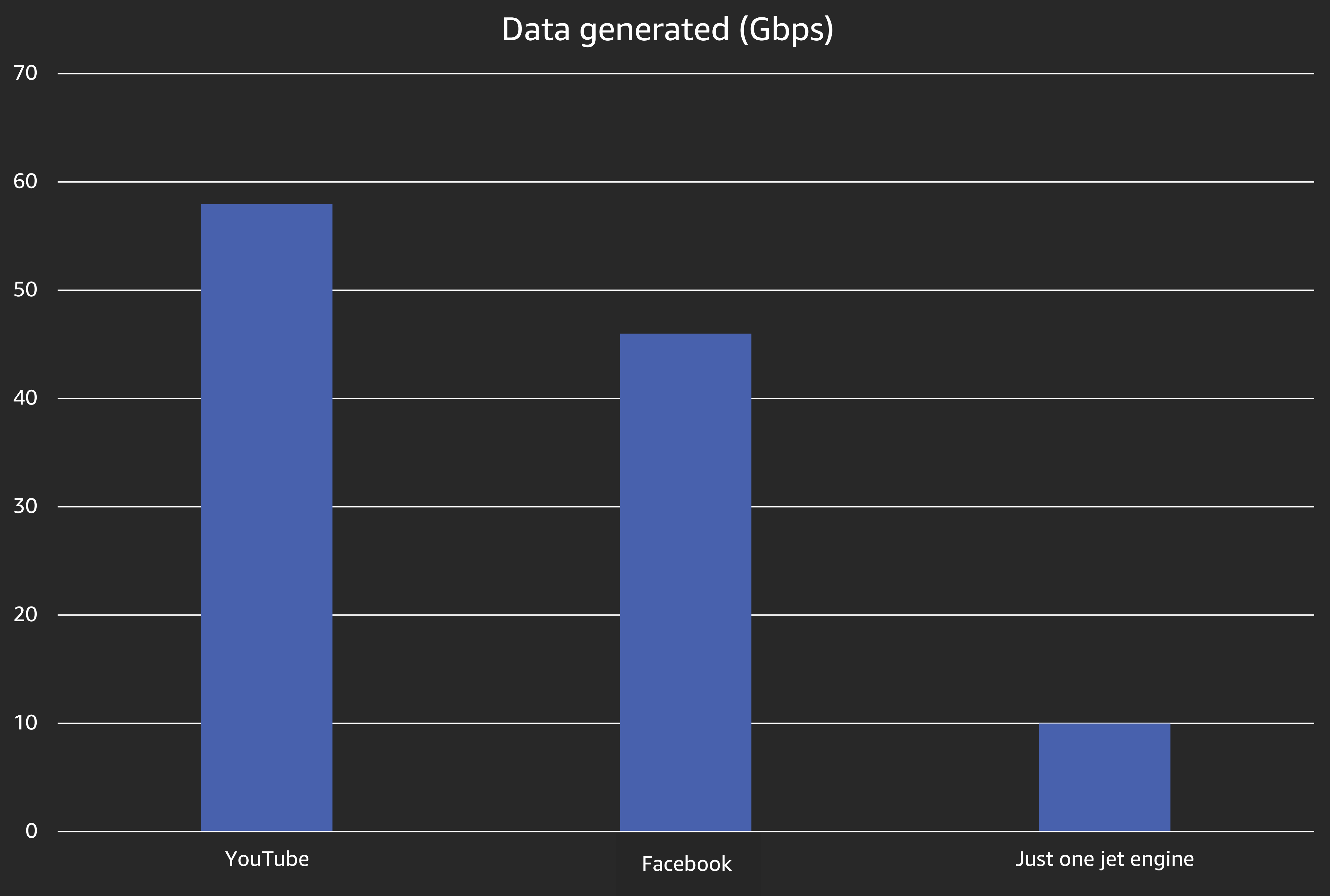
Air pollution measured at US embassy in Beijing

- Particulate matter 2.5 microns in diameter

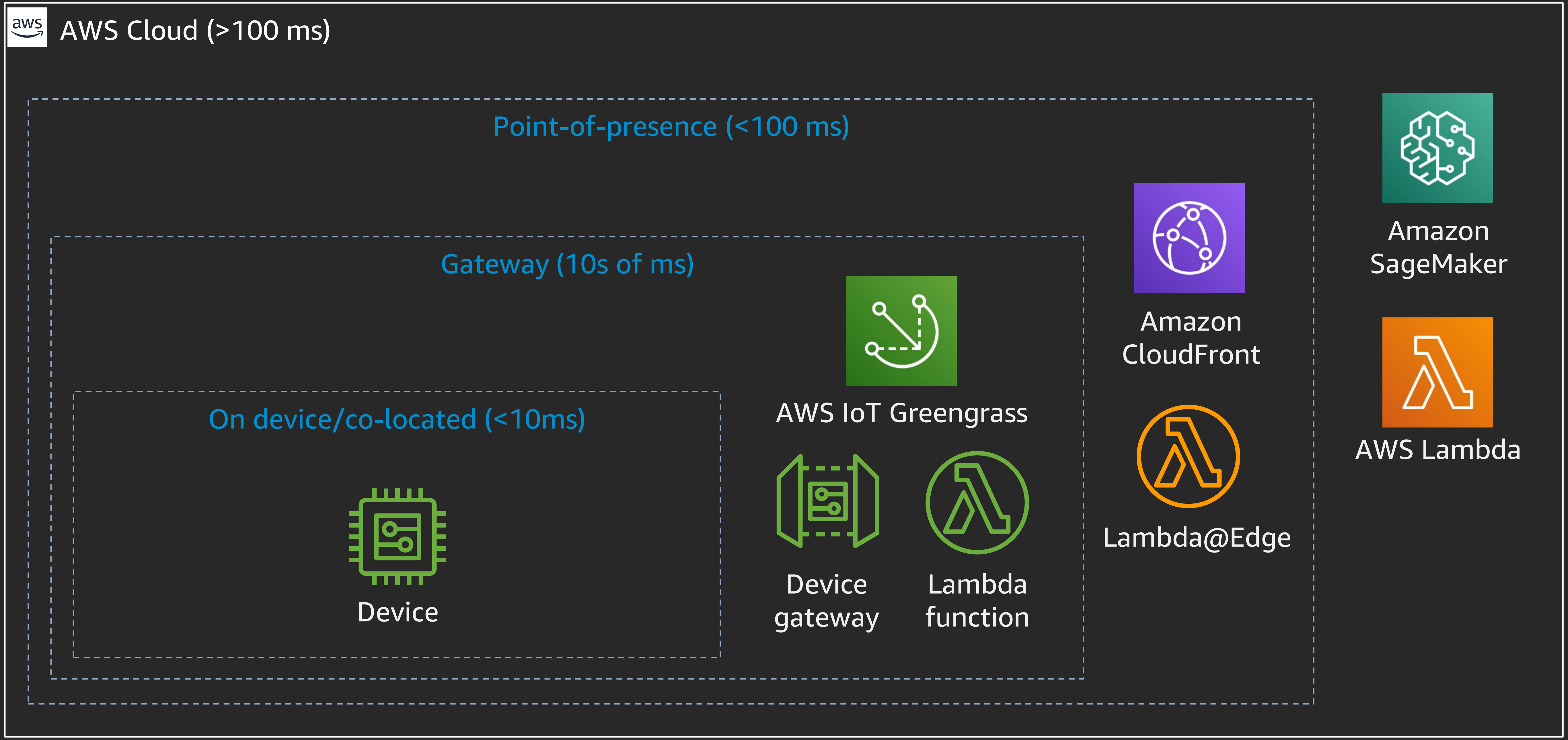
Sample data

Date	Pollution	Dew	Temp	Press	Wnd_dir	Wnd_spd	Snow	Rain
2010-01-02 00:00:00	129.0	-16	-4.0	1020.0	SE	1.79	0	0
2010-01-02 01:00:00	148.0	-15	-4.0	1020.0	SE	2.68	0	0
2010-01-02 02:00:00	159.0	-11	-5.0	1021.0	SE	3.57	0	0
2010-01-02 03:00:00	181.0	-7	-5.0	1022.0	SE	5.36	1	0
2010-01-02 04:00:00	138.0	-7	-5.0	1022.0	SE	6.25	2	0

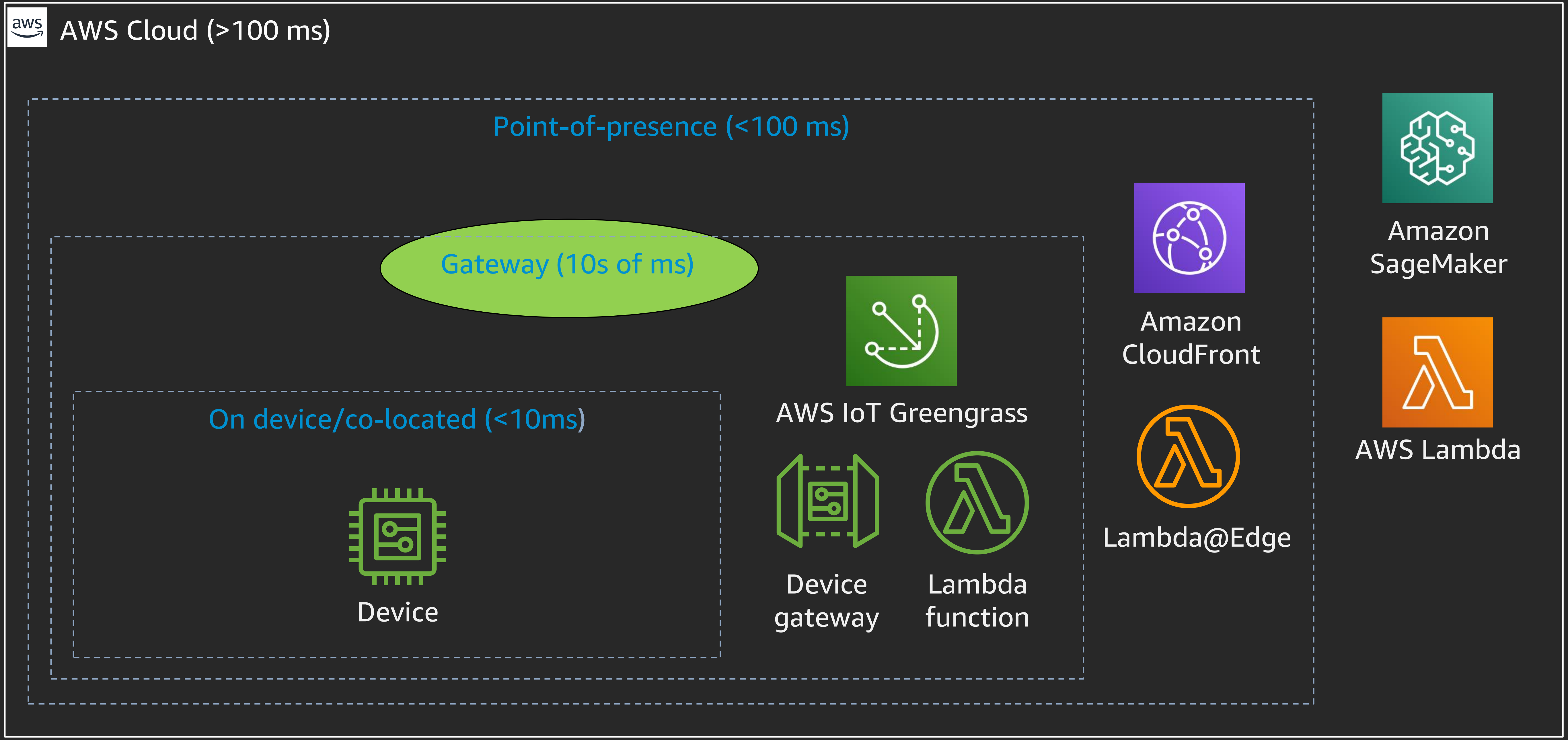
Data ingestion



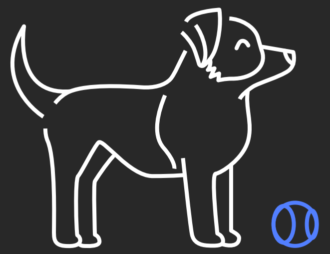
Model deployment



Model deployment in this solution

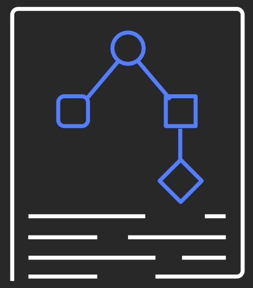


Challenges of ML



Defining features

Selecting intelligence to observe for patterns

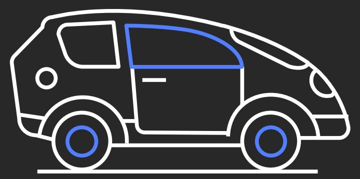


Choosing algorithm

How to best fit the data to the problem

Labeling data

Teaching the computer what to recognize



Model maintenance

What to do after the solution is deployed

Defining features

Be specific

- What precise problem are you trying to solve?
- What constitutes project success?
- How would you explain it to a team of five-year-olds?

Translate domain expertise to 1's and 0's

- What constitutes a machine or process failure?
- How will the model determine that from raw input?

Model-to-device ratio

- Is every device truly unique?

Casezy / Shutterstock.com

Choosing the right algorithm

Categorize the problem

- Supervised vs. unsupervised
- Regression vs. classification

Understand your data

- Analyze the data to understand trends with descriptive statistics
- Transform the data to represent the underlying features

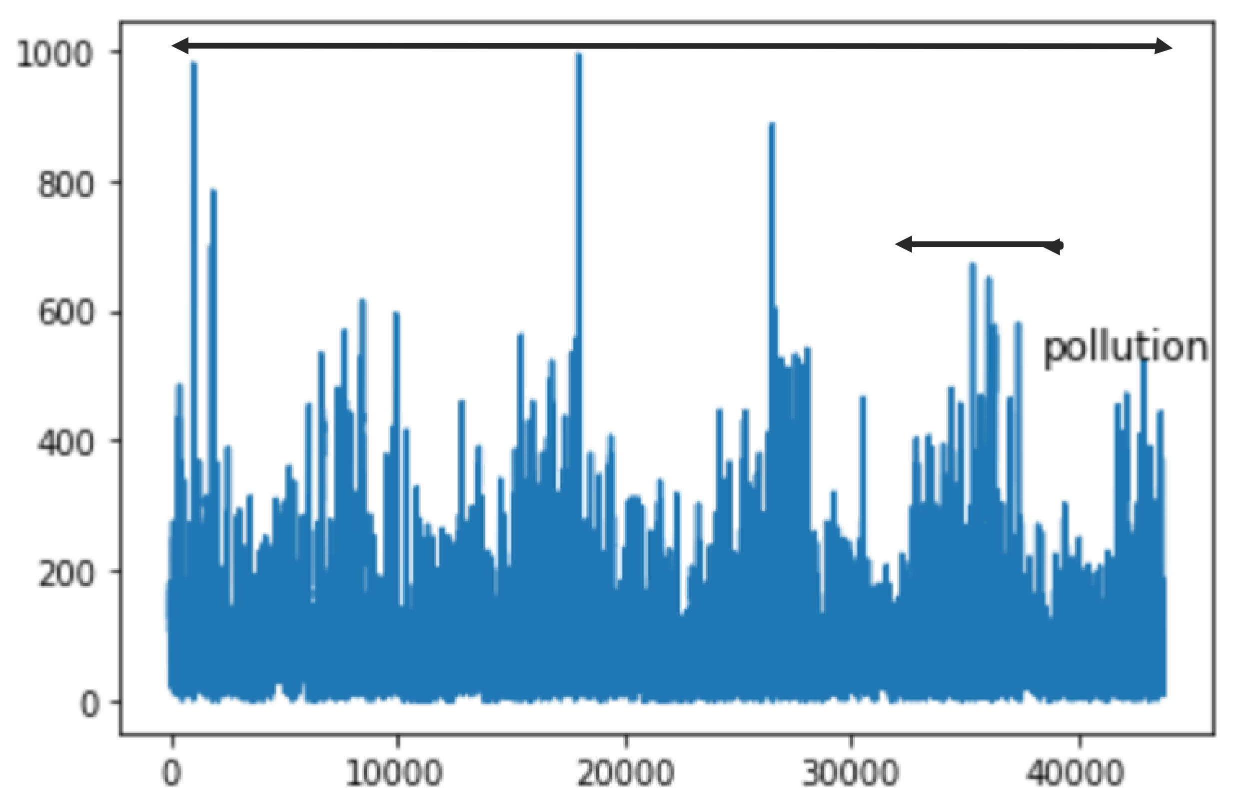
Model the algorithm

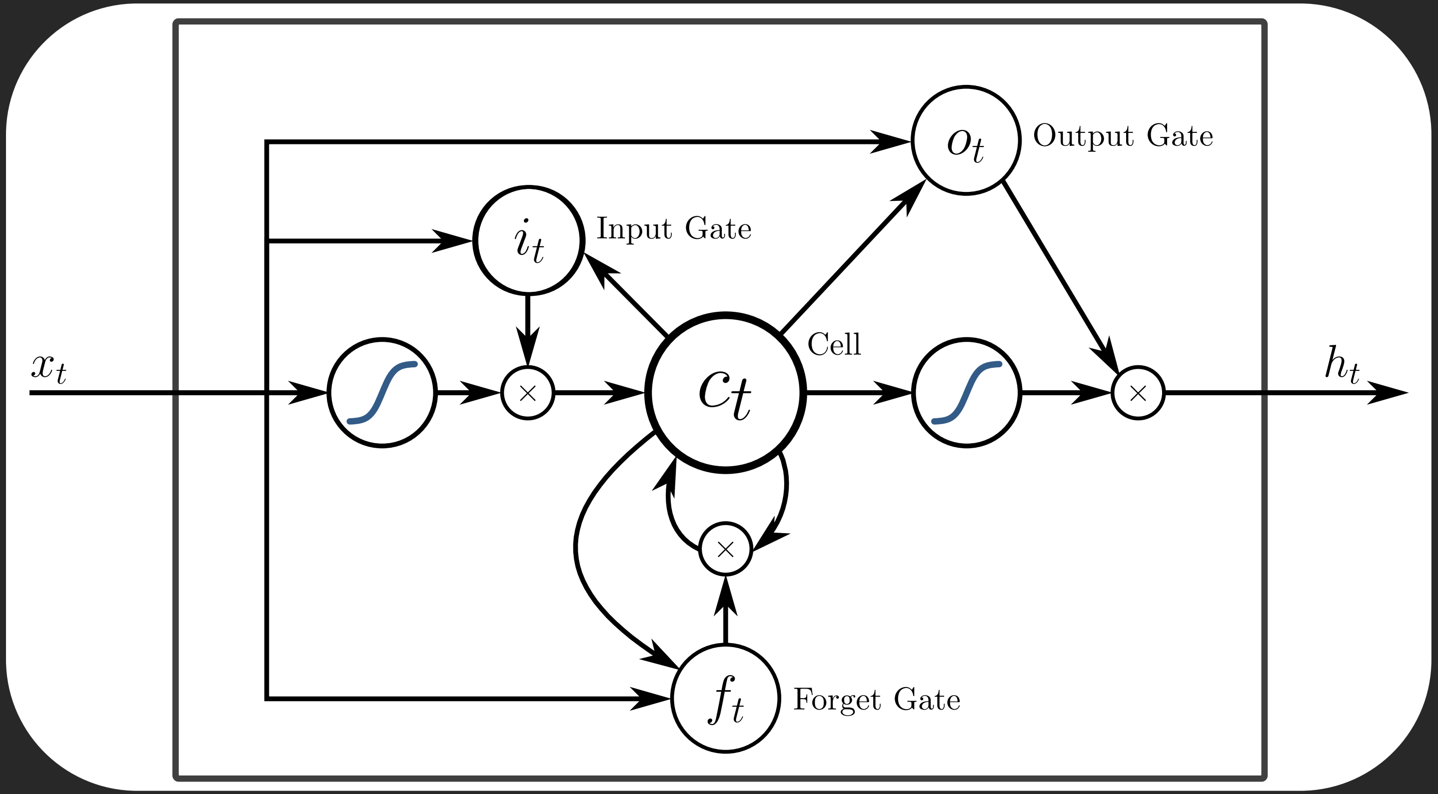
- Define accuracy, interpretability, and scaling for the model
- Test different models and scenarios
- Optimize hyperparameters

Algorithm in this solution

Long Short-Term Memory network (LSTM)

- Part of the Recurrent Neural Network family (RNN)
- Uses internal state to process sequence of inputs
- Capable of learning long term dependencies





Labeling data

Training data is unavailable

- Do you have the right domain experts available to label data?
- Can training data be programmatically generated from unlabeled data?

Programmatically generate labeled data

- Is there a related training data to start from?
- Can you jumpstart with mix of human labeling and ML reinforcement?

Imbalanced dataset

- Are dataset classes represented equally?
- Can dataset be resampled or augmented with synthetic data?

Model maintenance

How to evolve model with feedback?

- Retrain the model as the new data flows in
- Keep track if the predictions are incorrect

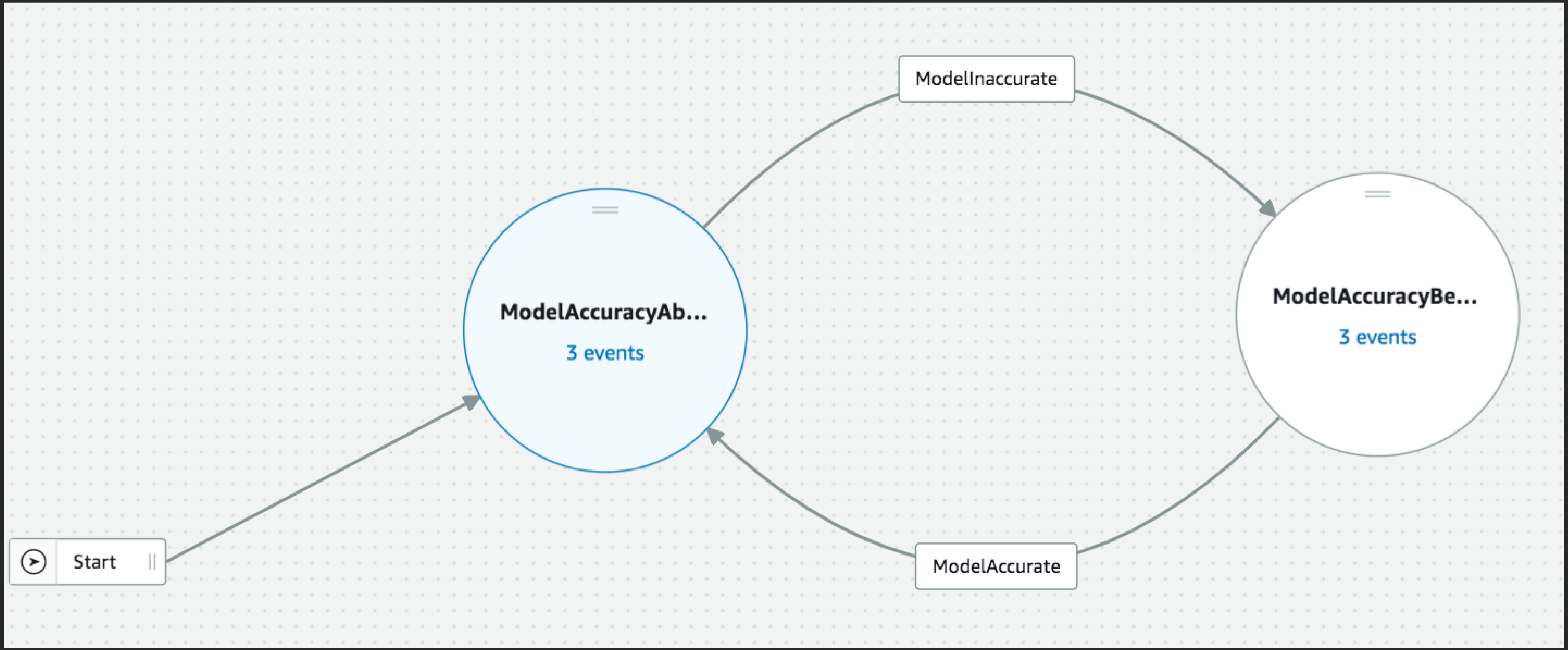
How does the model handle drift?

- Reinforcement learning could help correct model if it drifts
- Alerting mechanism in place to monitor drift

Model maintenance in this solution

Scheduled retraining of model as new data flows in

Trigger alerts using AWS IoT Events to know if predictions are incorrect



Demo

AWS re:Invent

Summary

Kick-off questions

Business readiness

- Do you have a S.M.A.R.T. problem statement?
- What are your success criteria?
- Do you have all the stakeholders?
- Who will operate and maintain the solution once deployed?

Build versus buy

- Does your equipment vendor offer predictive maintenance?
- Does a third-party vendor[1] sell a compatible brownfield solution?
- Does the project timeline afford an internal build?

1. https://aws.amazon.com/partners/find/

Summary

How to get this demo solution

- Download template https://github.com/aws-samples/amazon-sagemaker-aws-greengrass-custom-timeseries-forecasting
- Deploy with AWS CloudFormation

Further documentation and learnings

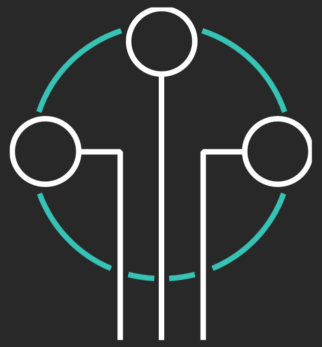
- AWS IoT Analytics user guide https://docs.aws.amazon.com/iotanalytics/latest/userguide/welcome.html
- Blog: https://aws.amazon.com/blogs/iot/using-aws-iot-for-predictive-maintenance/

Another introductory solution

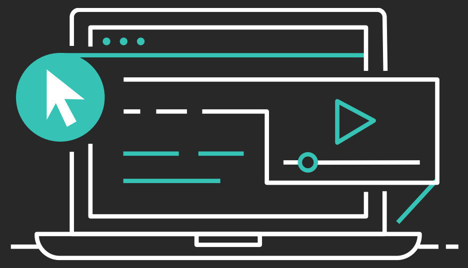
- Solution home page https://aws.amazon.com/solutions/predictive-maintenance-using-machine-learning/
- Focuses on Amazon SageMaker for training and inference

Learn IoT with AWS Training and Certification

Resources created by the experts at AWS to help you build IoT skills



Take the free digital curriculum, Internet of Things (IoT) Foundation Series, to build IoT skills and work through common scenarios



25+ additional free digital courses cover topics related to IoT, including:

- AWS IoT Core
- AWS IoT Greengrass
- AWS IoT Analytics
- AWS IoT Device Management
- AWS IoT Events

Visit the Learning Library at https://aws.training

aws training and certification

Thank you!

AWS re:Invent

Please complete the session survey in the mobile app.