Machine Learning

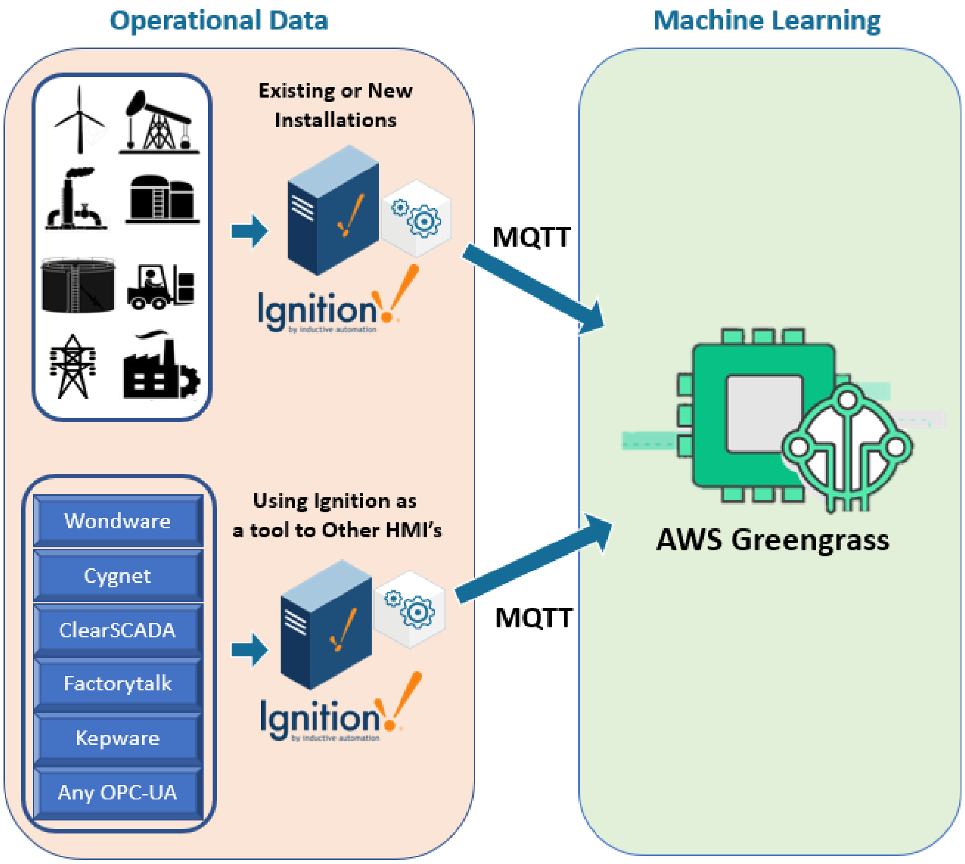
Bridging the OT IT Gap for Machine Learning with Ignition and AWS Greengrass

Cirrus Link SOLUTIONS

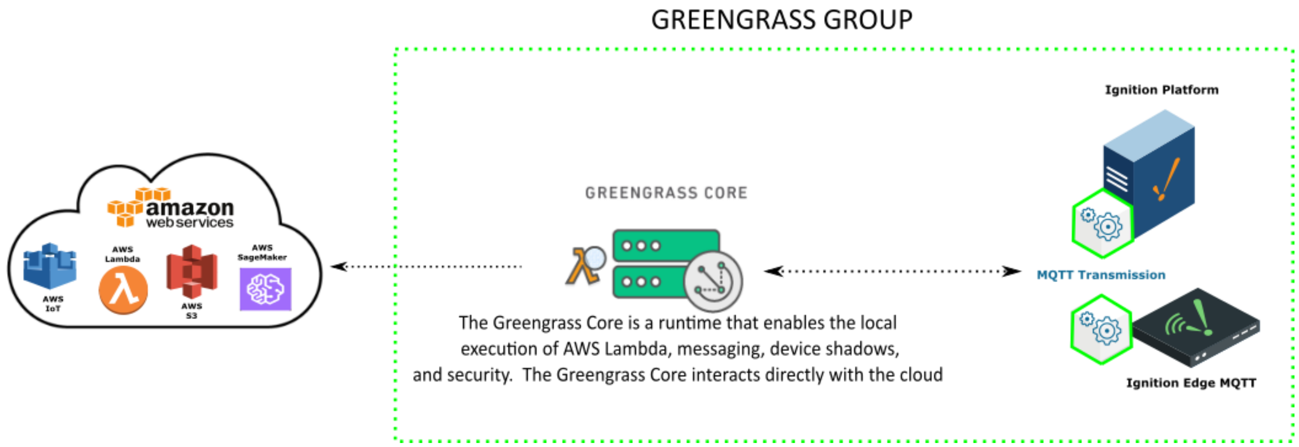
844.924.7787
www.cirrus-link.com

Simply Connect Ignition & AWS Greengrass for Machine Learning!

Bridging the OT IT gaps is now easier using the tools offered by the Ignition Platform with the Cirrus Link MQTT Transmission Module and AWS's Greengrass product offerings. The Ignition software from Inductive Automation is the industry leading SCADA/HMI platform, which utilizes the MQTT Transmission Module to securely send data using MQTT. AWS Greengrass is the latest predictive modelling environment that runs on edge hardware next to the plant, this significantly reduces the cost of sending vast amounts of data to the cloud continuously to execute machine learning algorithms. These tools enable clients to quickly take advantage of machine learning and predictive maintenance to drive operational efficiencies and increase profits.



The diagram above shows how easy it is to utilize the Ignition platform and MQTT Transmission Module to connect the OT data to the IT machine learning. It connects with simple configurations to existing Ignition systems, new Ignition systems and to any existing HMI/SCADA platform that provides an OPC-UA/DA connection. The data from PLCs, RTUs, sensors or any other IOT device is securely available for machine learning requiring no code, vastly reducing time and effort. The data is published using MQTT to the AWS's Greengrass Core in a MQTT Sparkplug JSON format including tag description meta data. Using MQTT, the OT data is 100% self-discoverable including all meta data such as scaling factors, engineering units and proper tag names, saving thousands of hours manually configuring tags in multiple systems. The data also includes time stamp information of when it was measured for accurate time series analysis.



The Greengrass Core is a runtime that enables the local execution of AWS Lambda, messaging, device shadows, and security. The Greengrass Core interacts directly with the cloud

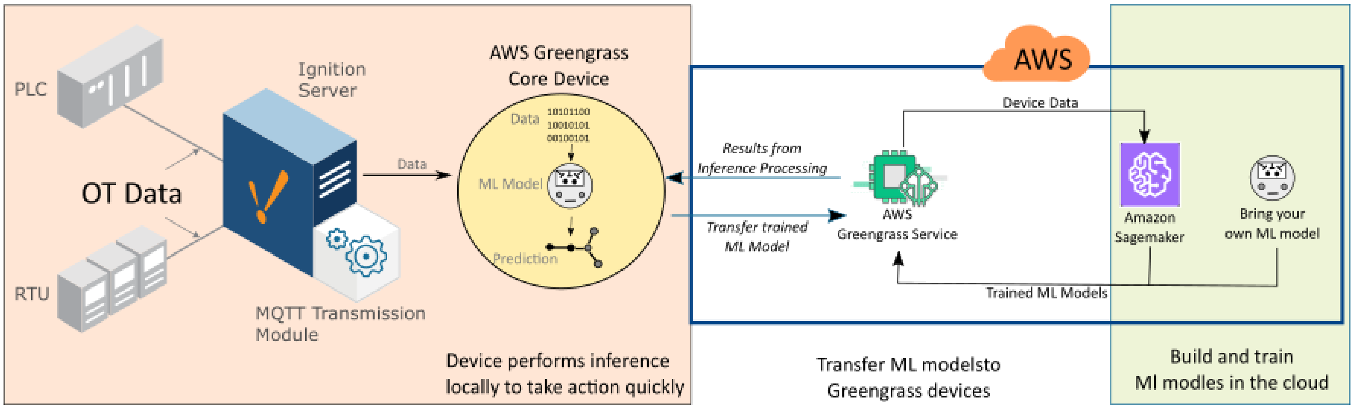
From diagram above, connectivity is accomplished by using either Ignition or Ignition Edge to the Greengrass core. One key factor is that the Greengrass core can run on the same Edge hardware as Ignition or on a separate Edge device providing local algorithms and models for real-time analytics. This minimizes data traffic going to the cloud and reduces cost while increasing system response times. AWS Greengrass is software that lets you run local compute, messaging, data caching, and sync capabilities for connected devices in a secure way. With AWS Greengrass, connected devices can run AWS Lambda functions, keep device data in sync, and communicate with other devices securely – even when not connected to the Internet.

Now, with the Greengrass Machine Learning (ML) Inference capability, you can also easily perform ML inference locally on connected devices. Machine Learning works by using powerful algorithms to discover patterns in data and construct complex mathematical models using these patterns. Once the model is built, you perform inference by applying new data to the trained model to make predictions for your application. Building and training ML models requires massive computing resources, so it is a natural fit for the cloud. But, inference takes a lot less compute power and is typically done in real-time when new data is available, so getting inference results with very low latency is important to making sure your applications can respond quickly to local events.

AWS Greengrass ML inference gives you the best of both worlds. You use ML models that are built and trained in the cloud and you deploy and run ML inference locally on connected devices. For example, you can build a predictive model in the cloud about the diamond heads of boring equipment and then run it underground where there is no cloud connectivity to predict the wear and usage of the diamond.

Using Amazon SageMaker with Greengrass

Amazon SageMaker is a fully managed end-to-end machine learning service that enables data scientists, developers, and machine learning experts to quickly build, train, and host machine learning models at scale. This drastically accelerates all of your machine learning efforts and allows you to add machine learning to your production applications quickly. The diagram below shows the end-to-end architecture of the solution when using Amazon SageMaker.


PLC
RTU
OT Data
Ignition Server
MQTT Transmission Module
Data
AWS Greengrass Core Device
Data 10101100 10010101 00100101
ML Model
Prediction
Device performs inference locally to take action quickly
AWS
Results from Inference Processing
Transfer trained ML Model
AWS Greengrass Service
Device Data
Amazon Sagemaker
Bring your own ML model
Trained ML Models
Transfer ML modelsto Greengrass devices
Build and train Ml modles in the cloud

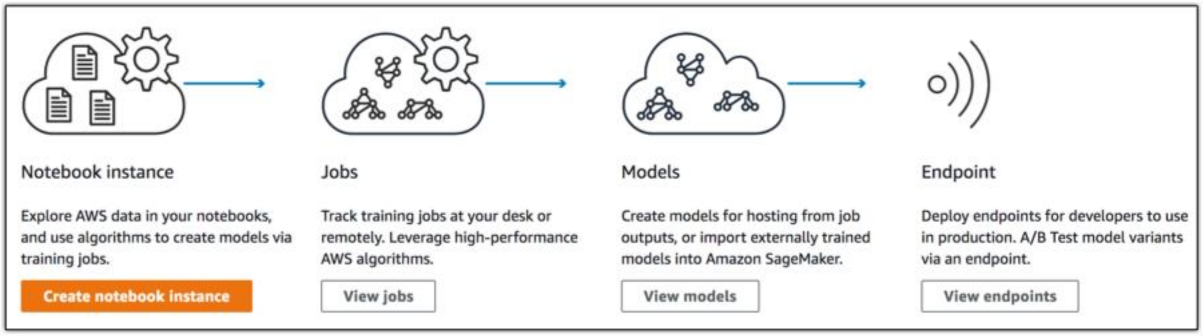
There are 3 main components of Amazon SageMaker:

- Authoring: Zero-setup hosted Jupyter notebook IDEs for data exploration, cleaning, and preprocessing. You can run these on general instance types or GPU powered instances.

- Model Training: A distributed model building, training, and validation service. You can use built-in common supervised and unsupervised learning algorithms and frameworks or create your own training with Docker containers. The training can scale to tens of instances to support faster model building. Training data is read from S3 and model artifacts are put into S3. The model artifacts are the data dependent model parameters, not the code that allows you to make inferences from your model. This separation of concerns makes it easy to deploy Amazon SageMaker trained models to other platforms like IoT devices.

- Model Hosting: A model hosting service with HTTPs endpoints for invoking your models to get real-time inferences. These endpoints can scale to support traffic and allow you to A/B test multiple models simultaneously. Again, you can construct these endpoints using the built-in SDK or provide your own configurations with Docker images.

Each of these components can be used in isolation which makes it easy to adopt Amazon SageMaker to fill in the gaps in your existing pipelines. That said, there are some powerful things that are enabled when you use the service end-to-end. The diagram below shows the process in modeling with SageMaker.



Notebook instance

Explore AWS data in your notebooks, and use algorithms to create models via training jobs.

Jobs

Track training jobs at your desk or remotely. Leverage high-performance AWS algorithms.

Models

Create models for hosting from job outputs, or import externally trained models into Amazon SageMaker.

Endpoint

Deploy endpoints for developers to use in production. A/B Test model variants via an endpoint.

Benefits

Bridges the OT IT Gap

With the tools Ignition and MQTT Transmission offer, you can quickly configure and publish data to AWS Greengrass bridging the OT IT gap either at a control center or at the Edge. No code is needed to write to gain access to the data. The data is pushed securely and not requested from IT. This eliminates the need to open up TCP/IP ports for IT and provides a secure process to go through the DMZ zone of the layer 3 to layer 4 of the ISA 95 Purdue model.

Easily run ML Inference on Connected Devices

Performing inference locally on connected devices reduces the latency and cost of sending device data to the cloud to make a prediction. Rather than sending all data to the cloud for ML inference, ML inference is performed right on devices and data is sent to the cloud only when it requires more processing.

Flexible

You can pick an ML model for each of your connected devices built and trained using the Amazon SageMaker service or stored in Amazon EC2. AWS Greengrass ML Inference works with Apache MXNet, TensorFlow, Caffe2, and CNTK. For NVIDIA Jetson, Intel Apollo Lake devices, and Raspberry Pi, AWS Greengrass ML Inference includes a pre-built Apache MXNet package, so you don’t have to build or configure the ML framework for your device from scratch.

Transfer Models to Your Connected Device with a Few Clicks

AWS Greengrass ML Inference makes it easy to transfer your machine learning model from the cloud to your devices with just a few clicks in the Greengrass console. From the Greengrass console, you can select the desired machine learning model along with AWS Lambda code and it will then be deployed and run on connected devices.

References

Amazon Website: https://aws.amazon.com/sagemaker/

Blog Post - Amazon SageMaker – Accelerating Machine Learning by Randall Hunt – 20 Nov 2017 - *https://aws.amazon.com/blogs/aws/sagemaker/*

Amazon Website - *https://aws.amazon.com/greengrass/ml/*