Xilinx Accelerates Industrial IoT Solutions with AWS IoT

Wes Skeffington (Xilinx)
Richard Elberger (AWS)
Jun 2019

XILINX.

Why & What of Cloud & Edge Collaboration

XILINX.

Why Industrial IoT?

- **Maintain system over lifecycle**
  - Upgrade/support cost optimizations
  - Deployment efficiencies & security patching
  - Prognostics to minimize system downtime

- **System cost savings via new intelligence**
  - Reduce system costs via ecosystem sensing
  - MBC/MPC control solutions
  - Real-time maintenance lifing models

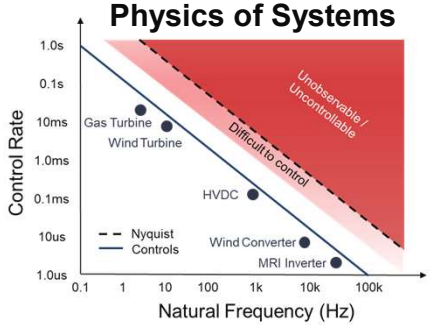
- **Monitor systems / shared dashboards**
  - Aggregated system level dashboard
  - Raise alarms & issues as occur
  - Dispatch response teams

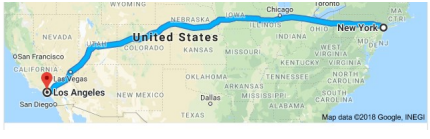
- **Monetize new services revenue**
  - Services is high profit & quick return opportunity
  - Unlock new asset utilization models
  - System & market level optimizers

3 M's of IoT
- ***Monitor***
- ***Maintain***
- ***Monetize***

XILINX.

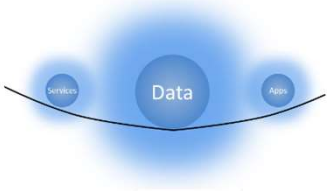
IIoT = Edge & Cloud Collaboration





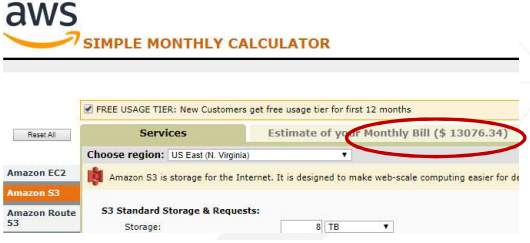
Distance NYC to LA: 2,800 miles
Speed of light: 186,000 miles/s
Round trip: 2*2800/186000 = 30ms
Required Control Rate = 10ms

"Data Gravity" –
Term coined by Dave McCrory



"Data Inertia" –
Term coined by Lew Tucker

Power Plant - Cloud Data Expense		
Raw Plant Data		
# of I/O Points	20000	#
# of Calculated Points	10000	#
Sampling Rate	4.00E-02	s
Data Size	4	bytes
Data Generation Calcs		
Data Generated per minute	180.0	MB/min
Data Generated per hour	10.8	GB/hour
Data Generated per day	259.2	GB/day
Data Generated per month	7776.0	GB/month



Power plant cloud data storage services alone: **>$13K/month**

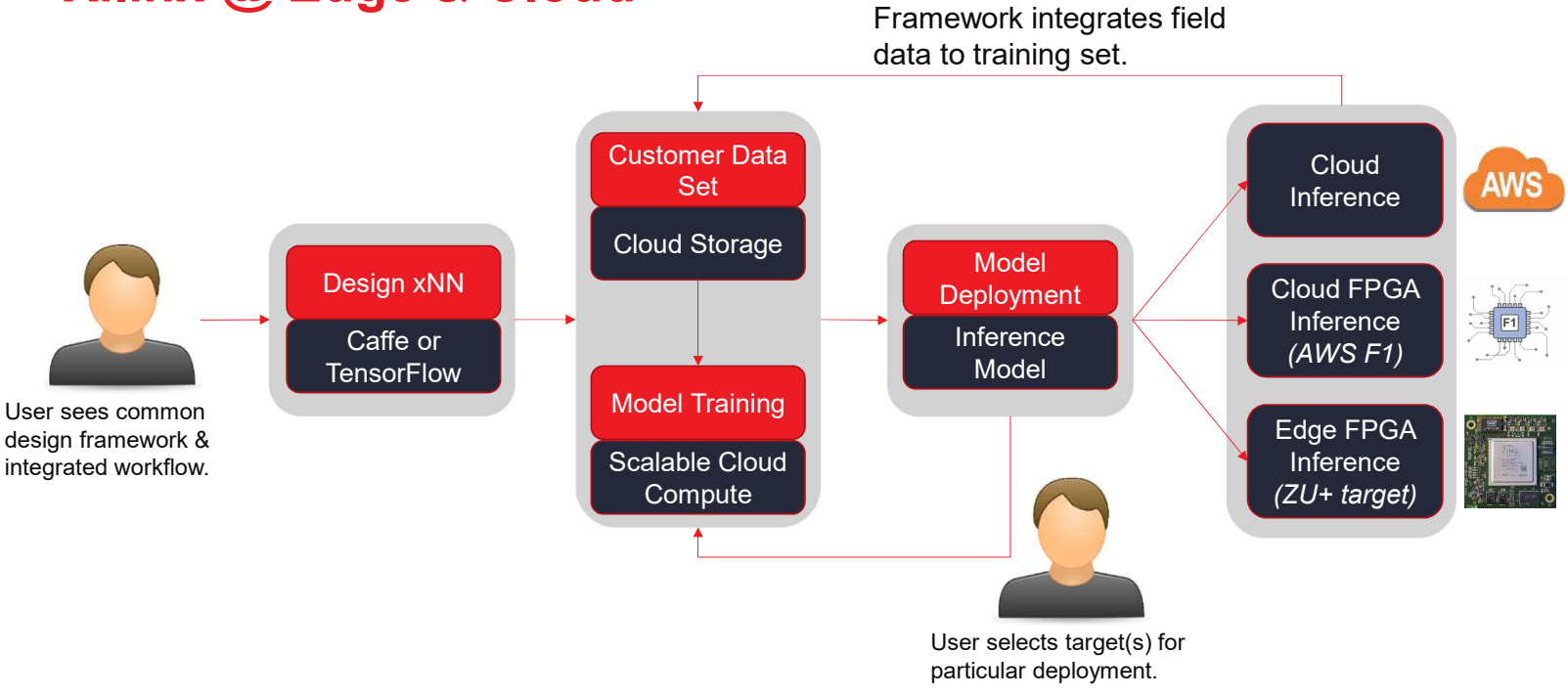
Physics dictate local app deployment

Doubled Compute Intelligence At The IIoT Edge

Data "mass" pulls computation to it

XILINX.

AWS + Xilinx - SageMaker ML Framework target Xilinx @ Edge & Cloud



XILINX.

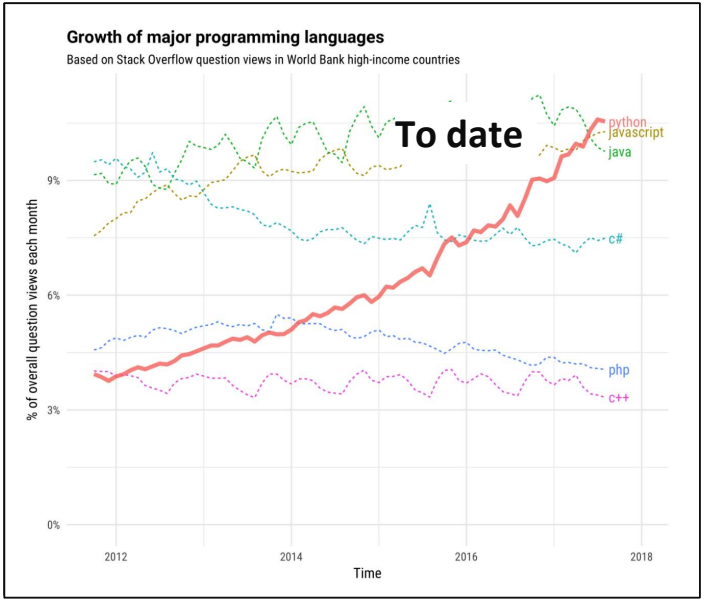
Python is increasingly the Language of Choice

Top Programming Languages,
IEEE Spectrum, July'18

Language Rank	Types	Spectrum Ranking
1. Python		100.0
2. C++		98.4
3. C		98.2
4. Java		97.5
5. C#		89.8
6. PHP		85.4
7. R		83.3
8. JavaScript		82.8
9. Go		76.7
10. Assembly		74.5

Python is listed as an embedded language for the first time

https://spectrum.ieee.org/at-work/innovation/the-2018-top-programming-languages



https://stackoverflow.blog/2017/09/06/incredible-growth-python/

Python is the fastest growing language: driven by data science, AI, ML and academia

Python Productivity on Zynq

Domain Experts

Embedded software Engineers

Hardware Engineers

Targeting the data center artificial intelligence, machine learning, data science

New users are not hardware designers, or embedded systems designers

PYNQ

Enable more people to program Xilinx processing platforms, more productively

XILINX

AWS IoT Solutions

XILINX.

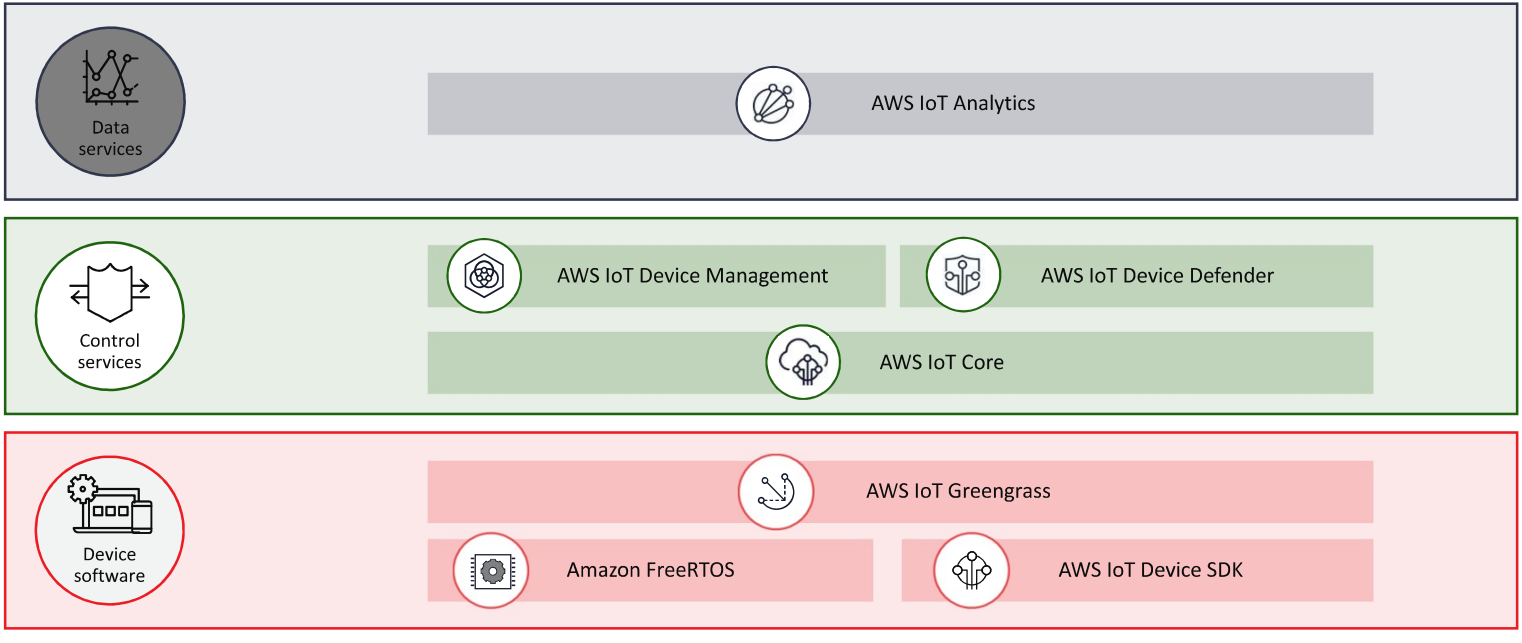
AWS IoT architecture

How do I extract value from my IoT data?

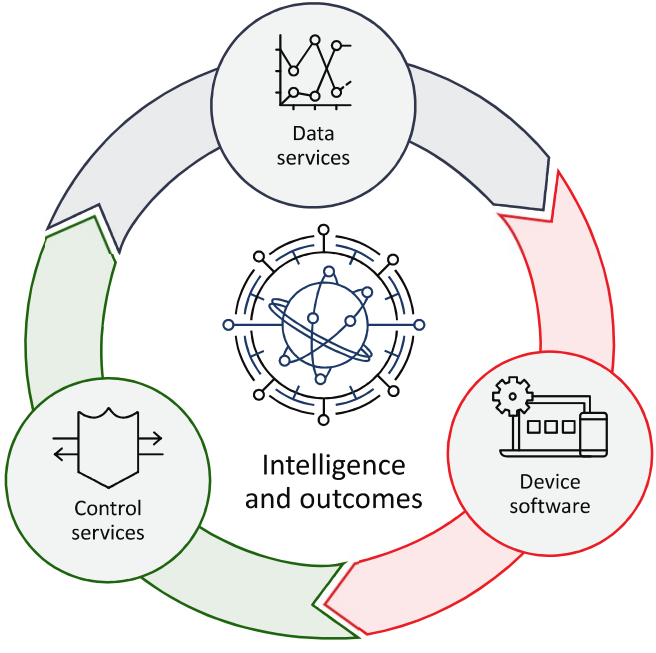
How can I control, manage, and secure my devices?

How can I connect my devices and operate at the edge?

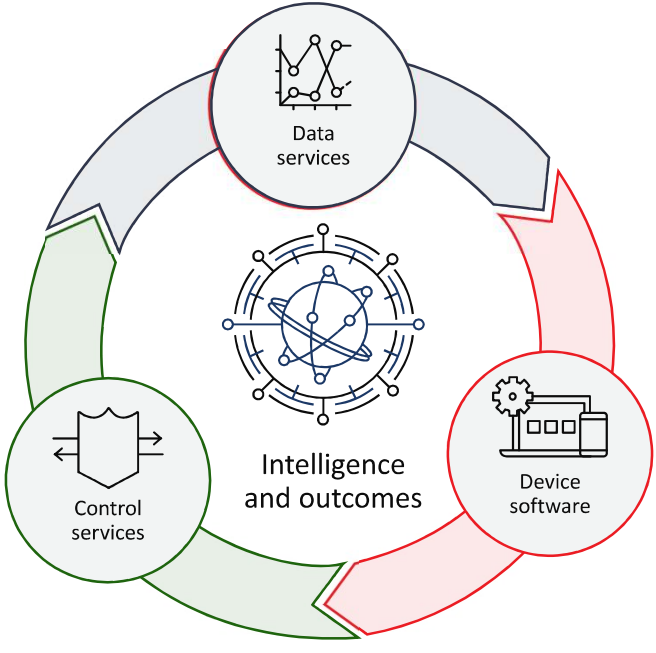
AWS IoT architecture



IoT virtuous cycle

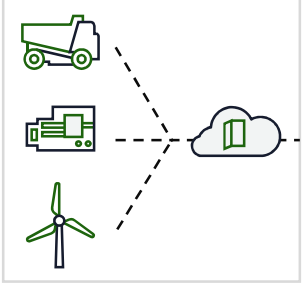


IoT virtuous cycle

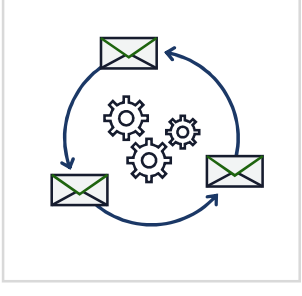


AWS IoT Core

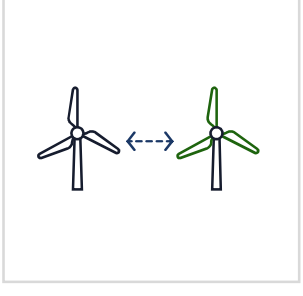
AWS IoT Core is a managed service that lets connected devices easily and securely interact with cloud applications and other devices.



To securely connect devices to the AWS cloud & other devices at scale

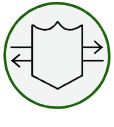


To route, process, and act upon data from connected devices



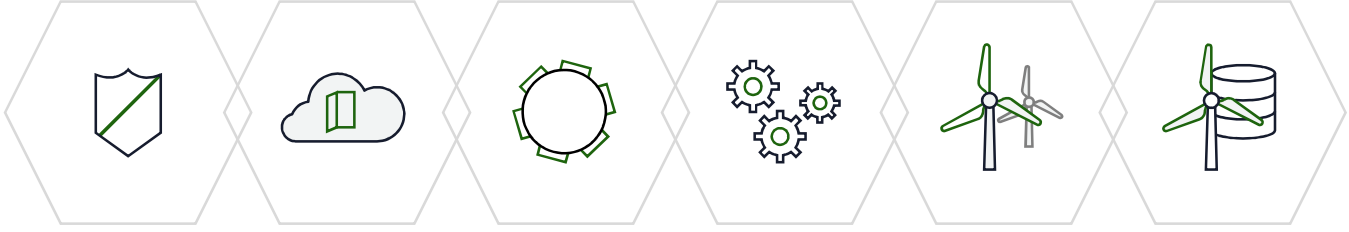
To enable applications to interact with devices even when they are offline

To fully integrate with other AWS services to reason on top of the data
(Analytics, Databases, AI, etc.)



Control services

AWS IoT Core



Identity Service

Manage authorization of devices & provision unique identities at scale

Device Gateway

Fully manage connectivity optimized for IoT workloads

Message Broker

Reliable and fast communication across your IoT fleet

Rules Engine

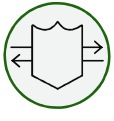
Ingest large amounts of IoT data at a low cost, pre-process it, & make it available to 10+ services for analytics, reporting, & visualization

Device Shadow

Understand and control the status of your device at any time

Registry

Define and catalog device for easy use by AWS services



Control services

AWS IoT Greengrass

Local Messages and Triggers

Enable device communication without a cloud connection

Local Actions

Simplify device programming with AWS Lambda

Data and State Sync

Operate devices offline & synchronize data when reconnected

Security

Mutual authentication & authorization between cloud and devices

Local Resource Access

AWS Lambda functions can access & use local resources of a given device

Machine Learning Inference

Perform ML Inference locally

Connectors

Extend edge devices with connections to external services

Secrets Manager

Deploy secrets to edge devices

Over the Air Updates

Easily update AWS IoT Greengrass Core

Device software

XILINX.

Local Messages and Triggers

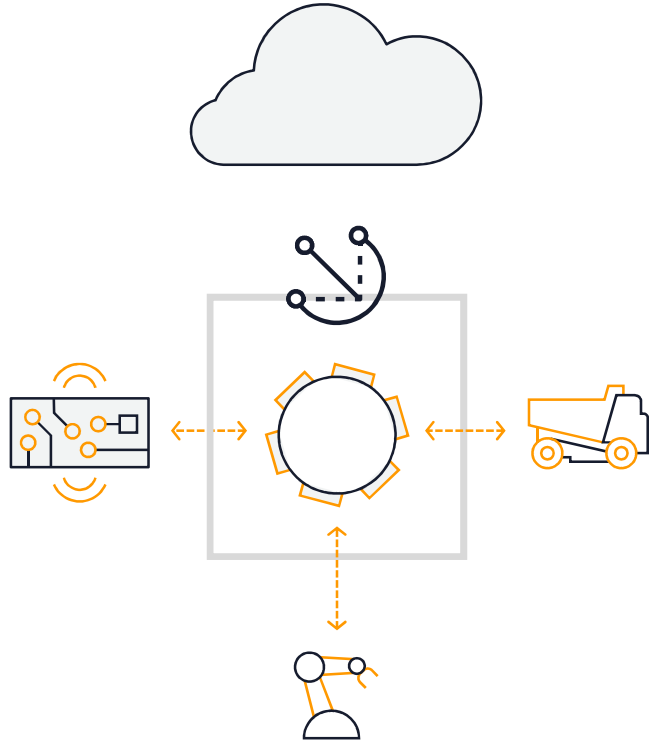
Enables messaging between devices on a local network so they can communicate without a cloud connection

Extends the AWS IoT MQTT pub/sub messaging paradigm locally to the edge

Allows AWS Lambda functions written in the cloud and deployed locally on the AWS IoT Greengrass Core to trigger and respond to events

Enables offline command and control operations from the AWS IoT Greengrass Core and other devices that use the AWS IoT Device SDK

For example, the AWS IoT Greengrass core can detect low moisture in the soil and in response, trigger an action to spray more water in smart greenhouse, without a connection to the cloud



Local Actions

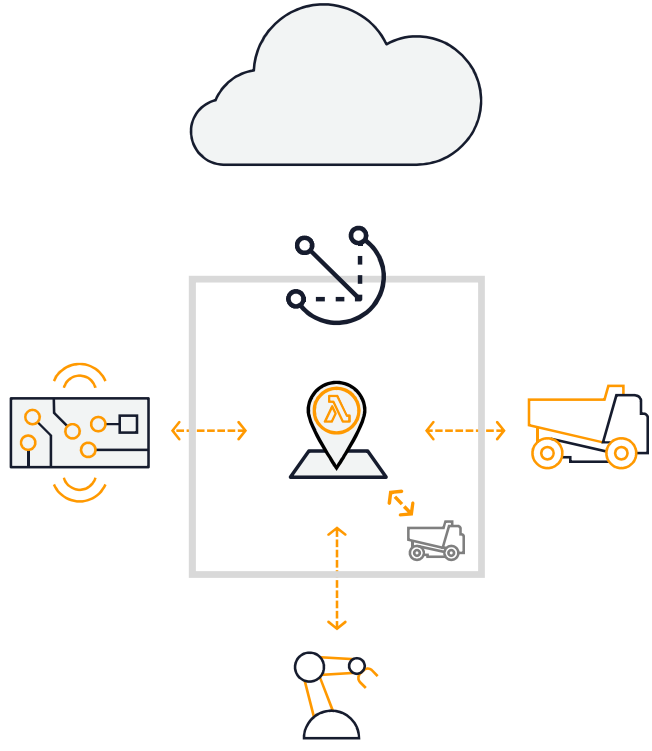
Simplify embedded software development with local AWS Lambda functions

Write event-driven AWS Lambda functions in the cloud and deploy them to devices

Run AWS Lambda functions written in Python 2.7, Node.js or Java

Invoke AWS Lambda functions with messaging and shadow updates

Offline actions and triggers for example, detecting low moisture in the soil and then triggering controls to spray more water inside a smart greenhouse



Local Resource Access

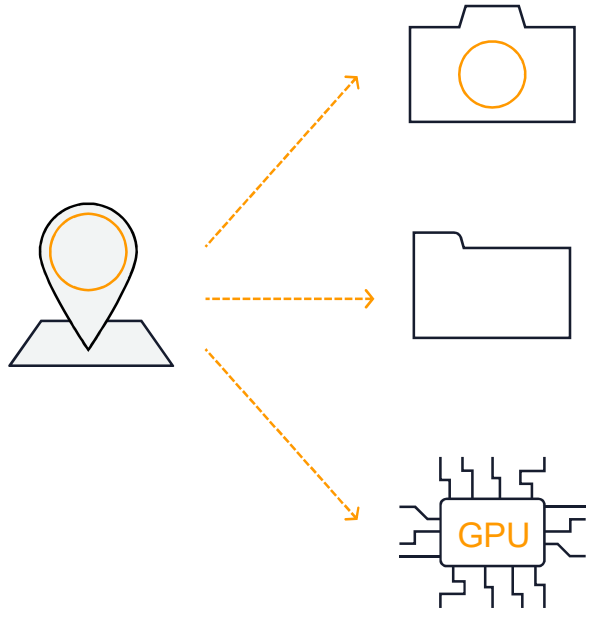
AWS Lambda functions can access & use local resources of a given device

Allows Lambdas to access local resources on a device

GPIO can be accessed to process sensor and actuator data

Lambdas can take advantage of the local file system on your operating system

Lambdas can use GPUs for hardware acceleration for machine learning



Machine Learning Inference

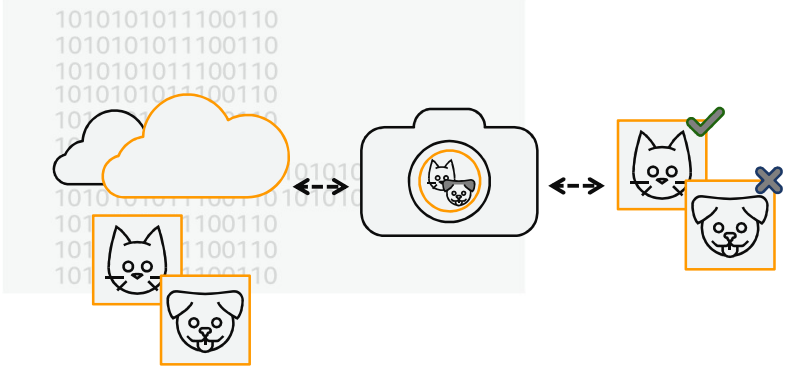
Perform ML Inference locally without data transfer costs or increased latency

Train models in the cloud using Amazon SageMaker or another service using EC2

ML Inference works with Apache MXNet and TensorFlow

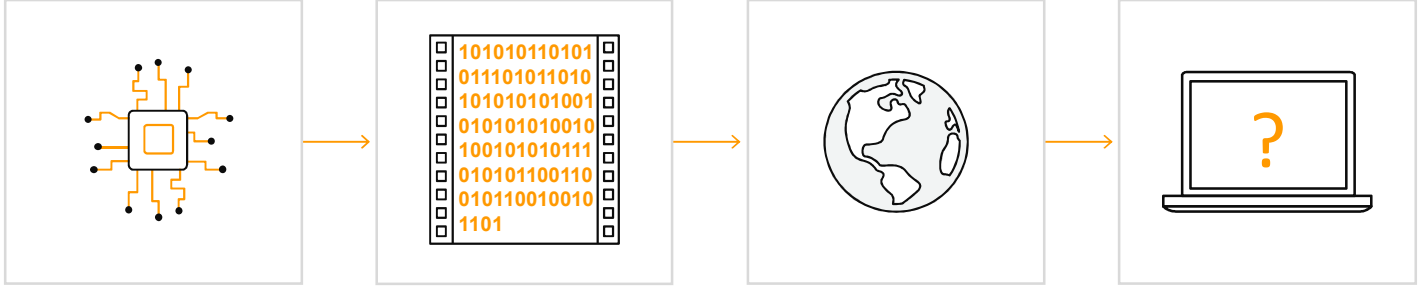
Transfer your trained models onto your device and also send data back to the cloud to improve model accuracy

Integration with Amazon SageMaker reduces model runtime footprint 100x and improves inference performance 2x



Amazon FreeRTOS

Amazon FreeRTOS is a microcontroller operating system that makes small, low powered edge devices easy to program, deploy, secure, connect, and maintain.



Will it work on my chip?

Does it have the functionality I need?

Where do I get it?

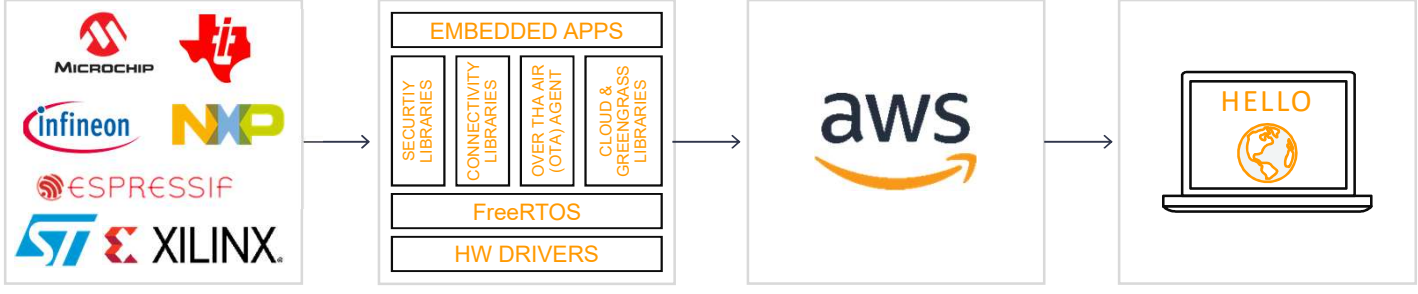
How do I start?

Device software

XILINX.

Amazon FreeRTOS

Amazon FreeRTOS is a microcontroller operating system that extends the popular FreeRTOS kernel to enable devices to securely connect to AWS, while making them easy to manage, deploy, and update.



Will it work on my chip?

Does it have the functionality I need?

Where do I get it?

How do I start?

Device software

Amazon FreeRTOS

Local Connectivity Libraries

Communicate with AWS IoT Greengrass devices without a cloud connection

Cloud Connectivity Libraries

Easily collect data and take actions on microcontroller-based devices

Security Connectivity Libraries

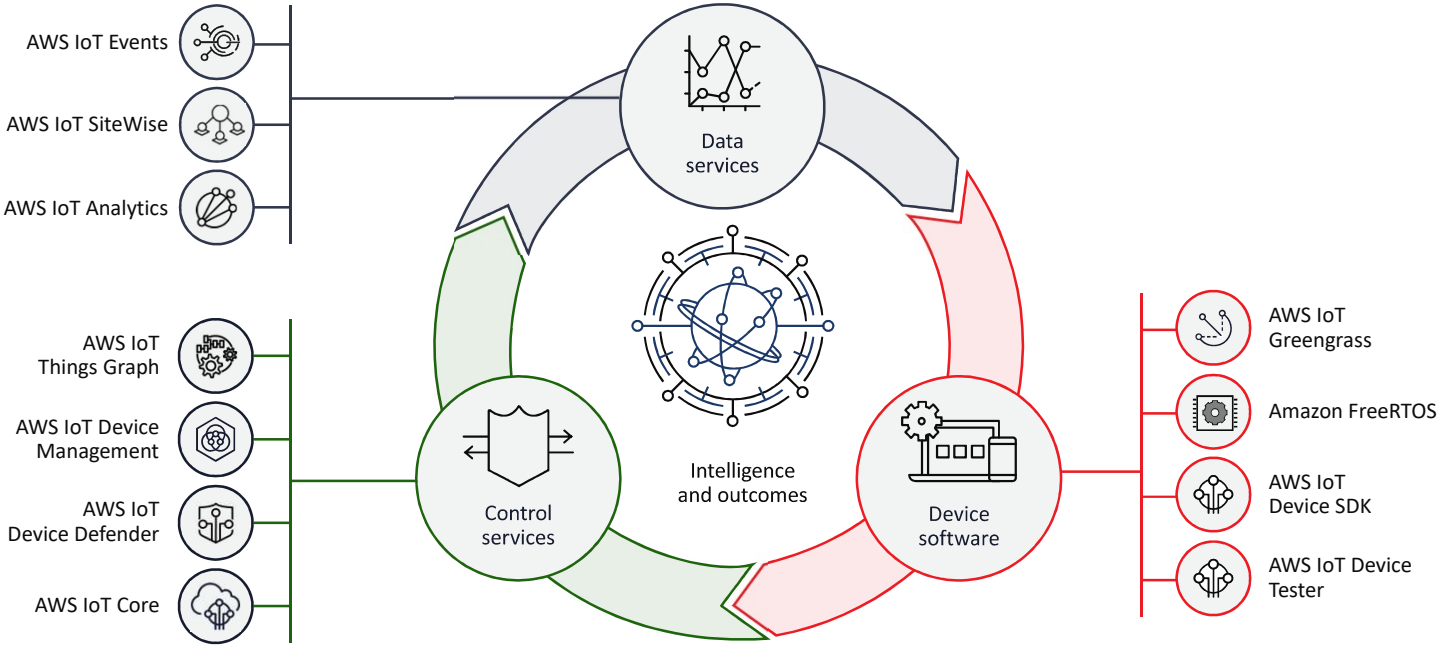
Secure device data and connections

OTA & Code Signing

Deploy security updates, bug fixes, and firmware updates to devices in the field

Device software

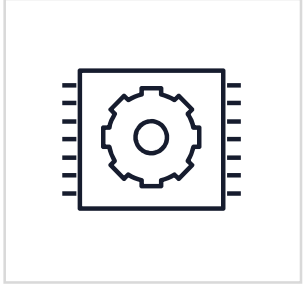
IoT virtuous cycle



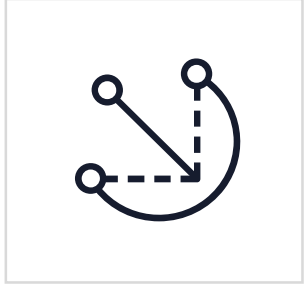
XILINX.

AWS IoT Device Tester

AWS IoT Device Tester is a test automation tool that lets you test Amazon FreeRTOS or AWS IoT Greengrass on your choice of devices.



AWS IoT Device Tester for Amazon FreeRTOS
Tests if the Amazon FreeRTOS cloud connectivity, OTA, and security libraries function correctly on top of microcontroller board device drivers



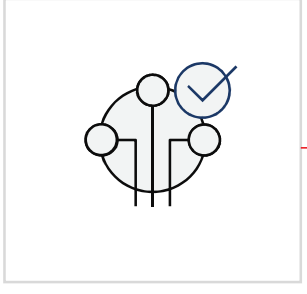
AWS IoT Device Tester for AWS IoT Greengrass
Tests if the combination of device's CPU architecture, Linux kernel configuration, and drivers work with AWS IoT Greengrass

Download AWS IoT Device Tester

from AWS IoT Greengrass and Amazon FreeRTOS product pages

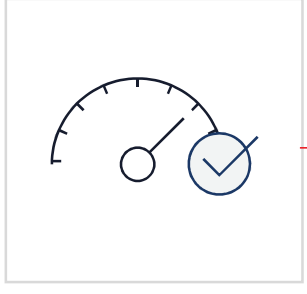
AWS IoT Device Tester

Gain Confidence



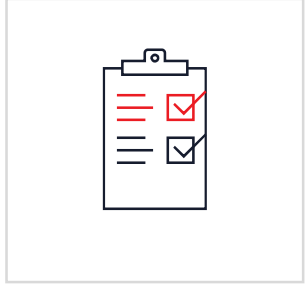
Authenticate and interoperate with AWS IoT services

Make Testing Easy



Sets up required cloud resources, automates compiling and flashing of test cases on the target device for execution

Get Listed



Submit AWS IoT Device Tester results to get listed in AWS Device Catalog*

Device software

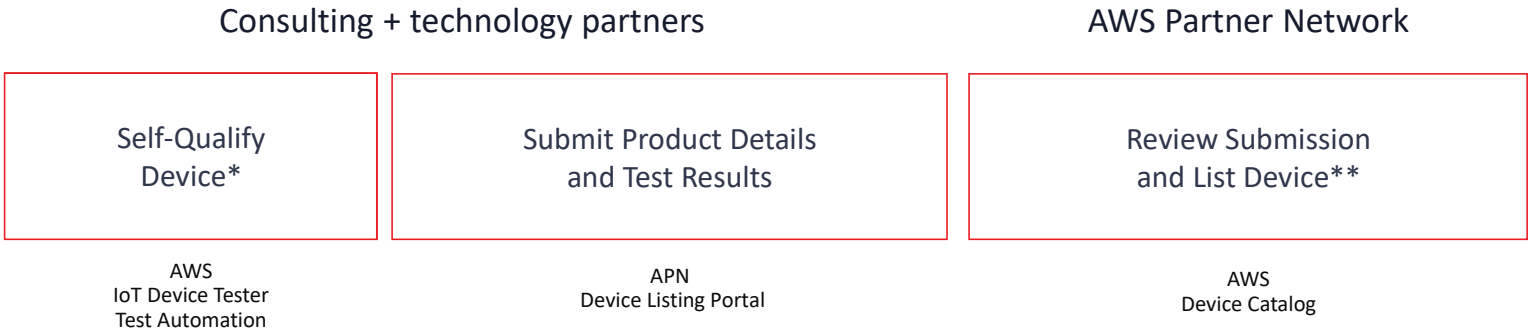
*Available as part of the Device Qualification Program

XILINX.

AWS Device Qualification Program

The AWS Device Qualification Program is a hardware qualification and incentive program that is open to all APN partners who want to qualify their device hardware by validating that they work with AWS IoT Greengrass and Amazon FreeRTOS.

Qualified devices are eligible for listing in the AWS Device Catalog, which allows customers to discover IoT devices that work with AWS.



*Supported for Amazon FreeRTOS and AWS IoT Greengrass

**APN may request partner-submitted hardware for technical validation

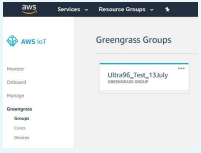
XILINX.

Workshop Use Case

Workshop Goals

- Connect Intelligent I/O module sensor data to AWS Cloud via Amazon FreeRTOS
- Connect & execute edge health check application on Unit Controller through AWS Greengrass Device Shadow
- Deploy edge based machine learning (ML) inference in FPGA/programmable logic through AWS Greengrass
- Bring ML inference response to AWS Cloud through AWS Greengrass

Xilinx + AWS IoT Demonstrator

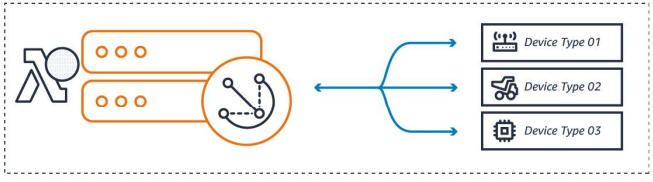


AWS IoT Core

Cloud

Greengrass Group

A defined group of Greengrass Cores and other devices that are configured to communicate with one another. A Greengrass Group may represent one floor of a building, one truck, or one home.



Greengrass Core

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

Device

Any device using Amazon FreeRTOS or IoT Device SDK can be configured to interact with Greengrass Core via the local network

AWS Cloud

Ultra96

MicroZed

XILINX.

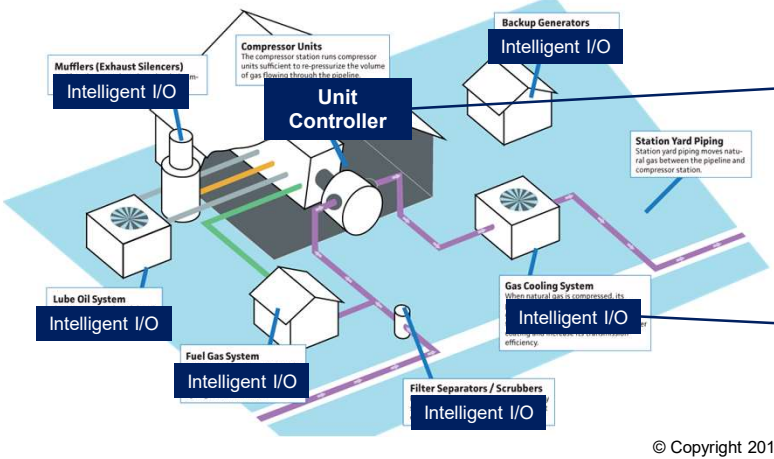
Industrial Controls IIoT Use Case

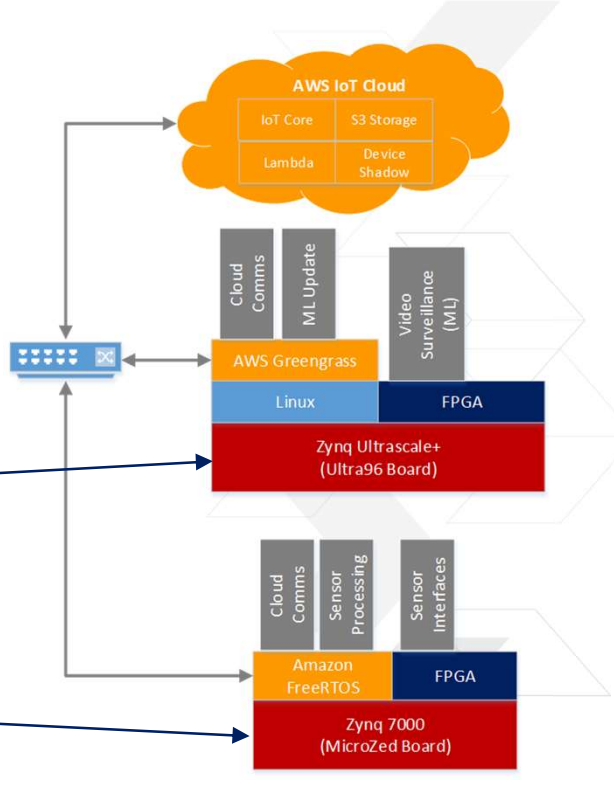
Zynq 7000 w/ Amazon FreeRTOS

- Targets limited resource devices
- Bridge physical & digital domains

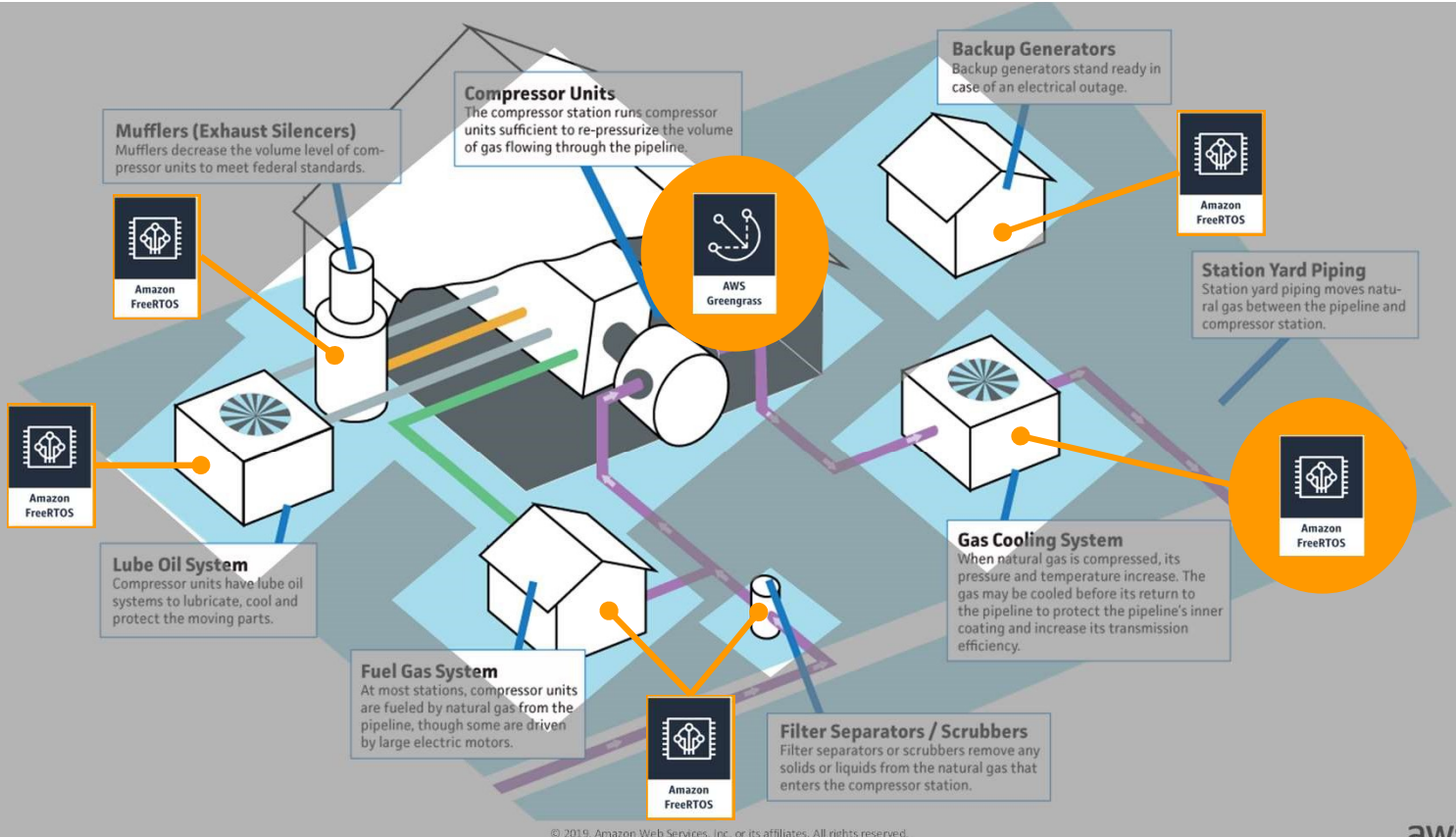
Zynq US+ w/ Linux & AWS IoT Greengrass

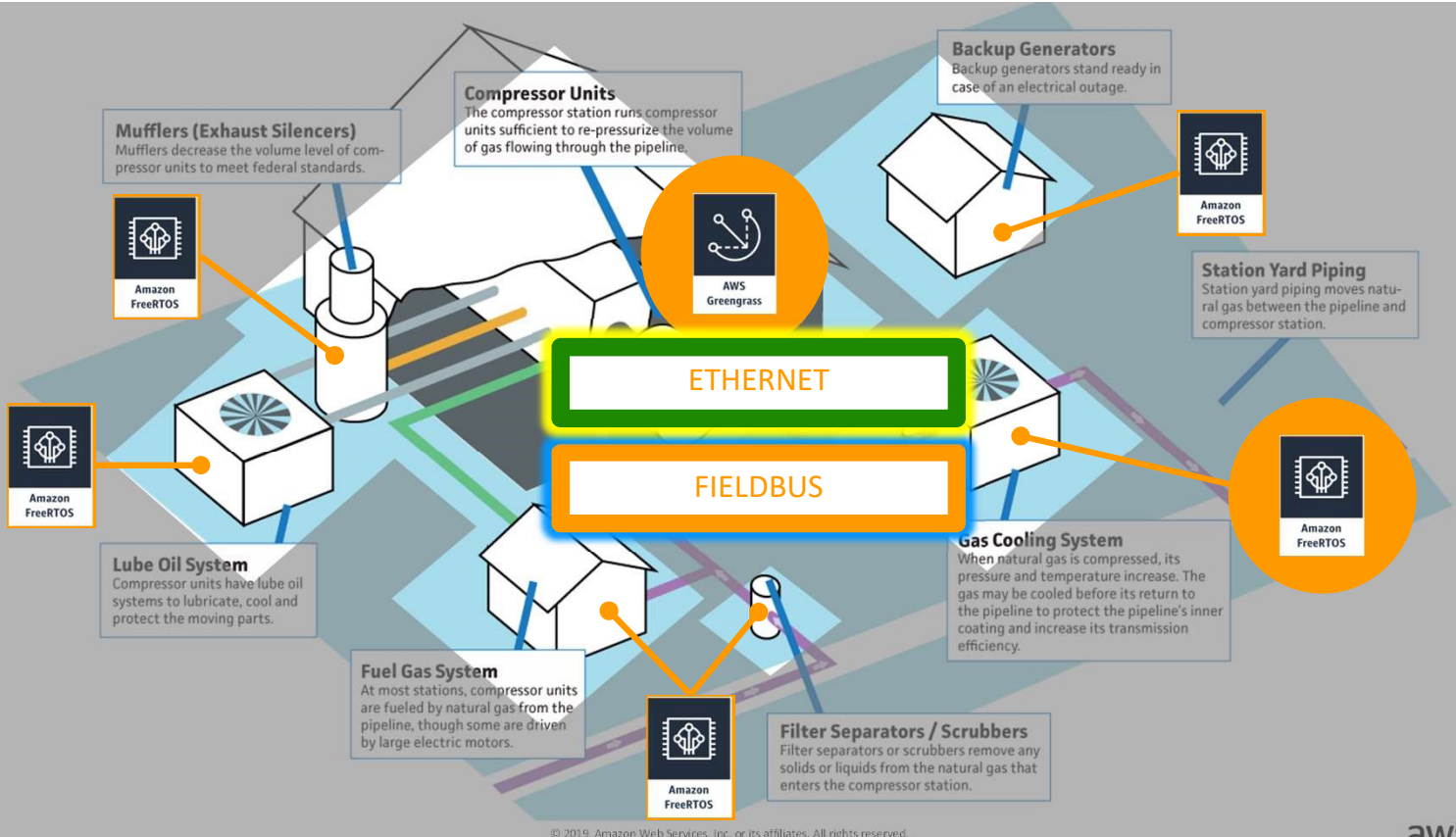
- Targets more capable embedded devices
- Brings together control action & local analytics



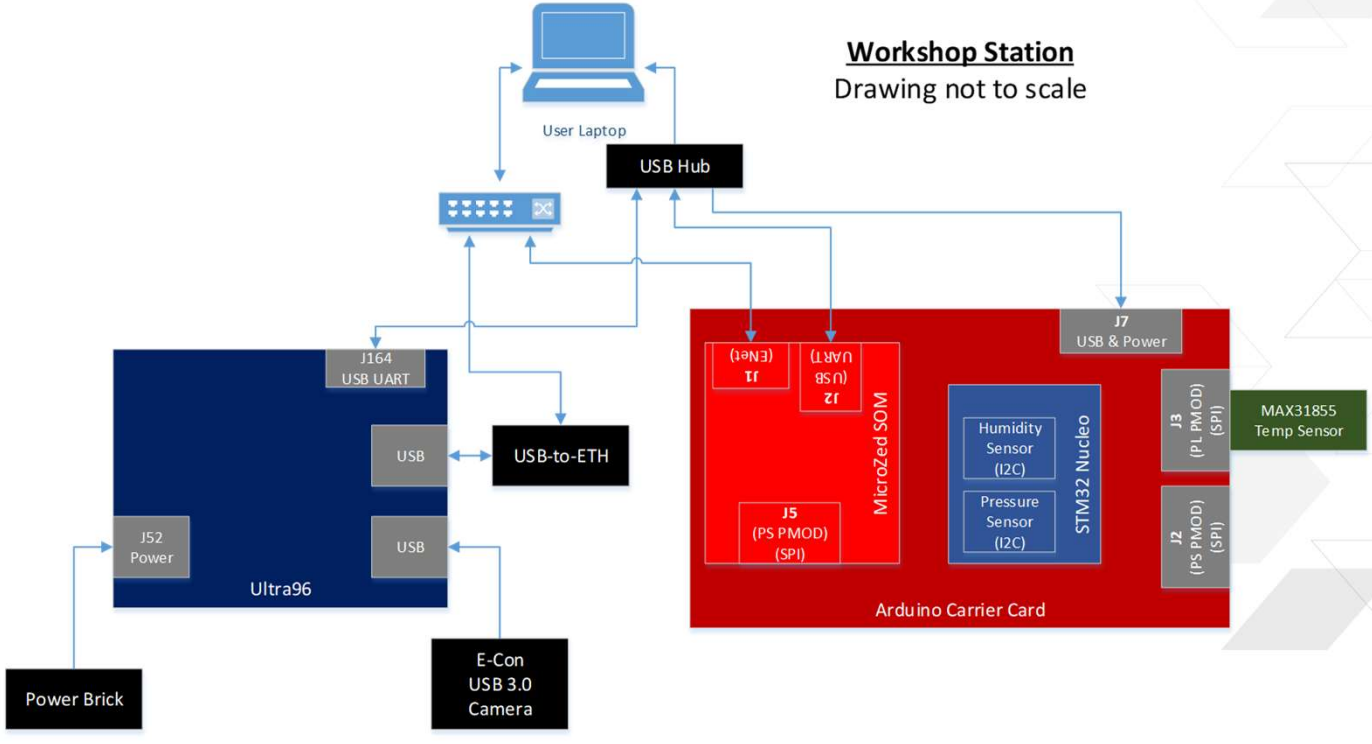


XILINX.



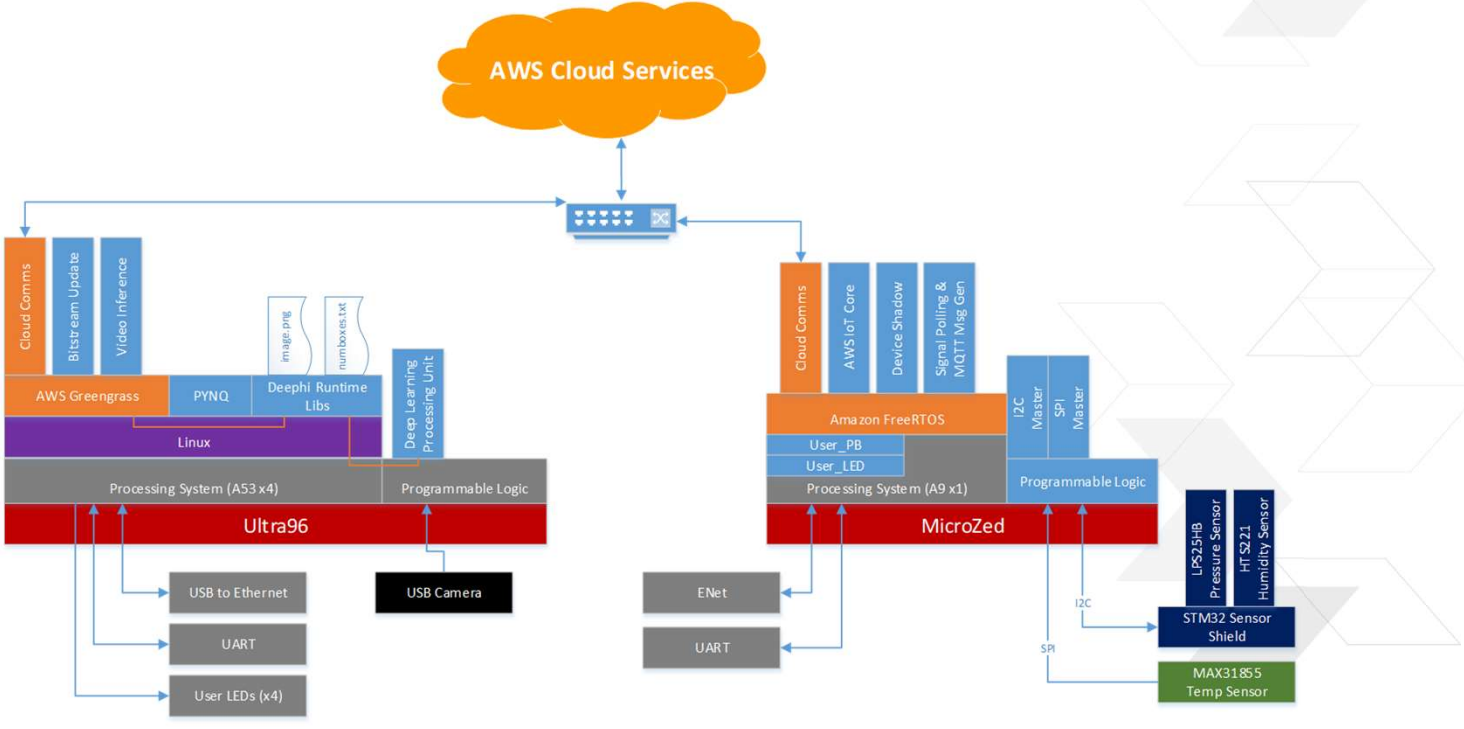


Attendee Workstation



XILINX.

Xilinx & AWS IoT Workshop



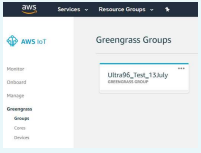
Details at https://github.com/rpcme/aws-cloud-and-xilinx-workshop

XILINX.

Summary & Next Steps

XILINX.

Xilinx AWS IoT Demonstrator

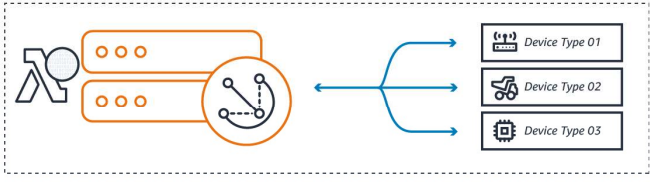


AWS IoT Core

Cloud

Greengrass Group

A defined group of Greengrass Cores and other devices that are configured to communicate with one another. A Greengrass Group may represent one floor of a building, one truck, or one home.



Greengrass Core

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

Device

Any device using Amazon FreeRTOS or IoT Device SDK can be configured to interact with Greengrass Core via the local network

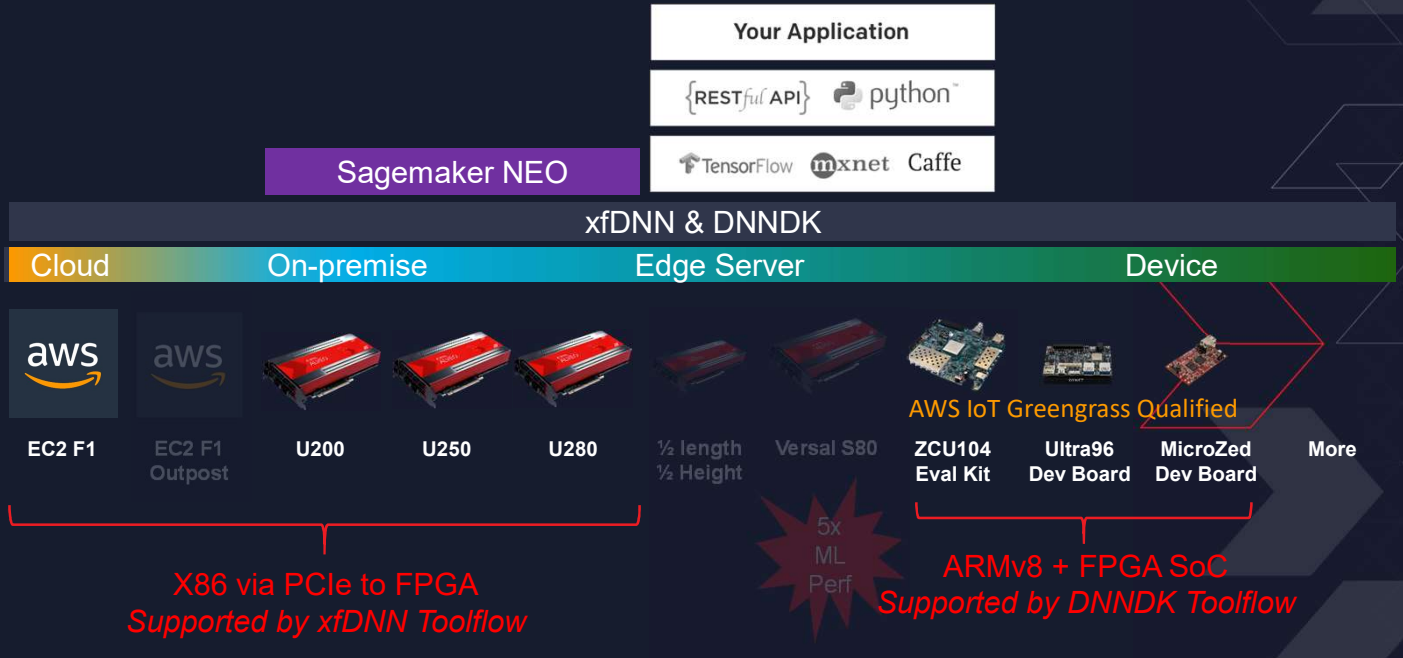
AWS Cloud

Ultra96

MicroZed

XILINX.

Xilinx Edge & Cloud – ML Inference Acceleration



Xilinx AWS IoT Qualified Devices

Amazon FreeRTOS MicroZed Starter Kit

- MicroZed HW platform from Avnet
- SW & application “getting started” at a:FreeRTOS

aws qualified device Amazon FreeRTOS

AWS IoT Greengrass on Ultra96

- Ultra96 board from Avnet
- SW & application “getting started” at AWS IoT Core

aws qualified device internet of things

AWS IoT Greengrass on ZCU104

- ZCU104 board from Xilinx
- SW & application “getting started” at AWS IoT Core

aws qualified device internet of things

Work through reference at https://github.com/rpcme/aws-cloud-and-xilinx-workshop

XILINX.

Adaptable.
Intelligent.

XILINX.