Introduction to AWS Greengrass

Julien Simon
Principal Technical Evangelist
Amazon Web Services

@julsimon

Most machine data never reaches the cloud

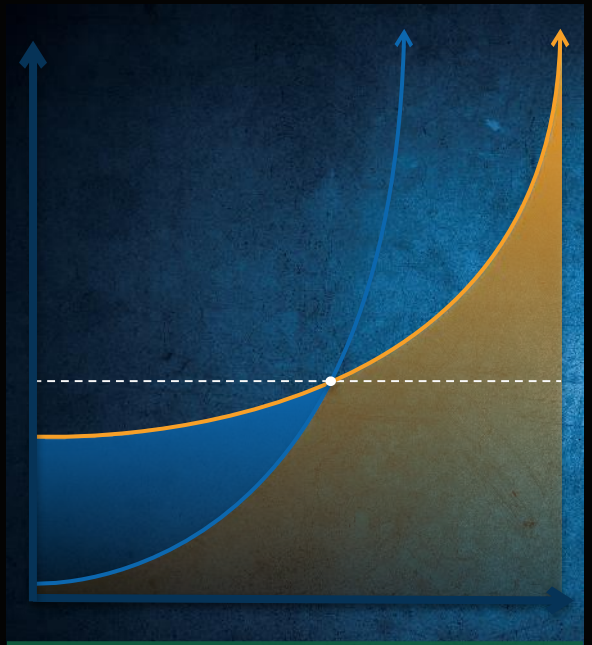
Medical equipment

Industrial machinery

Extreme environments

Why this problem isn't going away

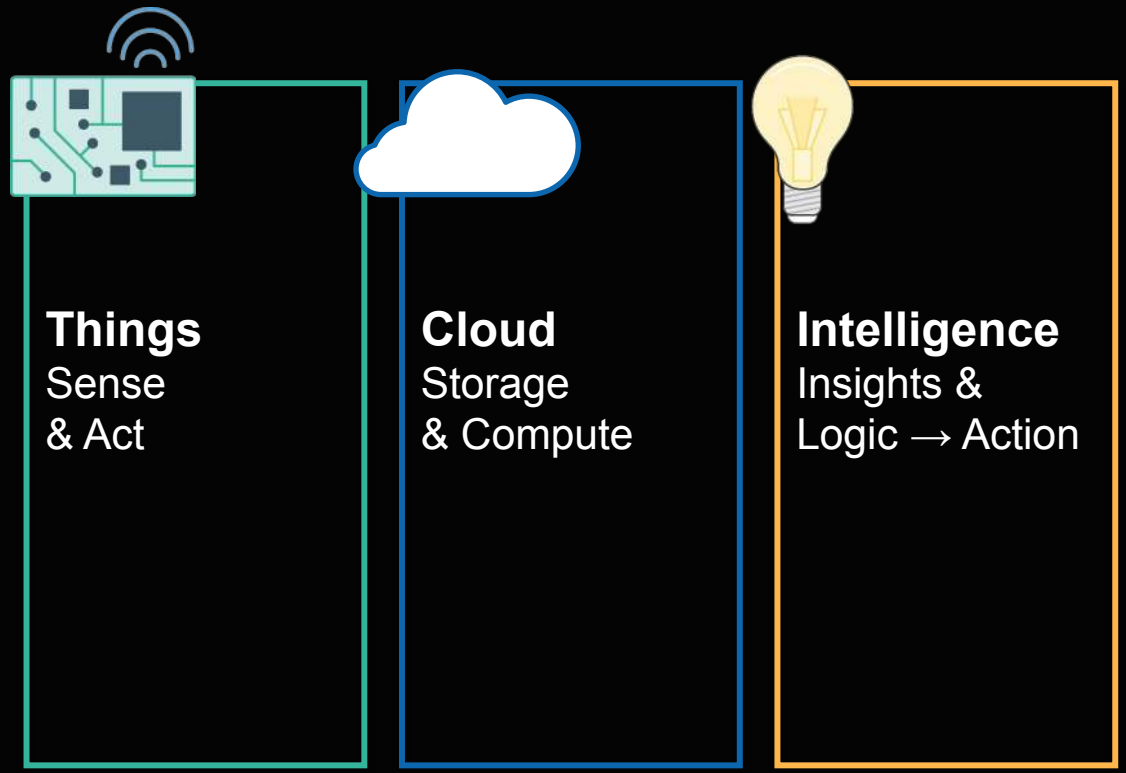
Law of physics



Law of economics

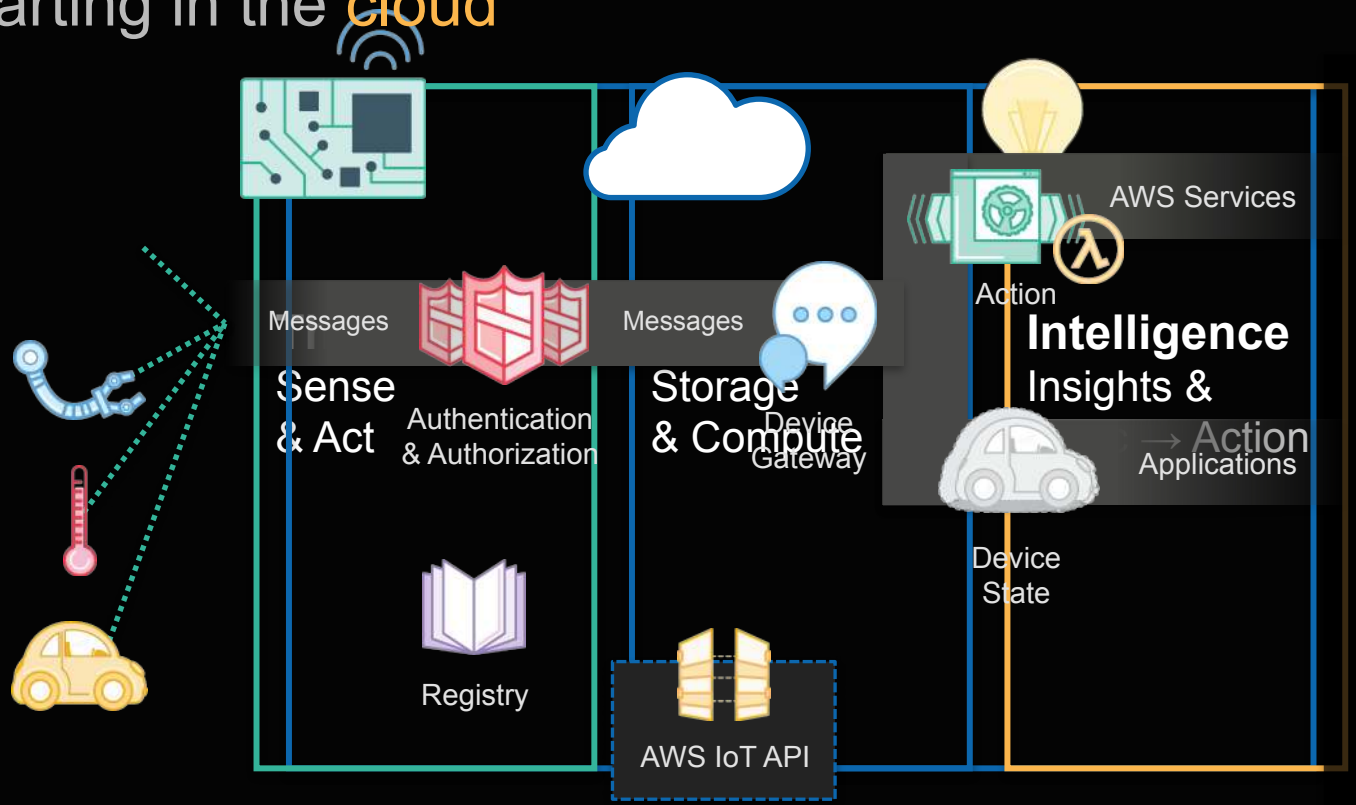
Law of the land

Three pillars of IoT



AWS IoT

Starting in the cloud



AWS IoT

Going to the edge

Introducing AWS Greengrass

*Note: Greengrass is NOT Hardware (You bring your own)

Cloud

Greengrass

AWS Greengrass extends AWS Processing Capabilities onto your devices, so you can process more of your data locally while taking advantage of the cloud

Service is available in North Virginia, Oregon, Frankfurt and Sydney.

Benefits of AWS Greengrass

Respond to local events quickly

Operate **offline**

Simplified device programming

Reduce the cost of IoT applications

Rio Tinto

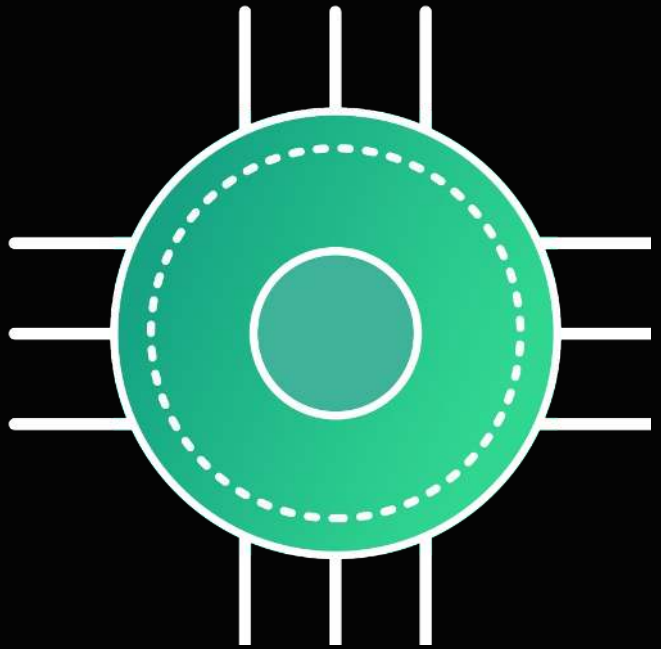
Boron, California


StanleyBlack&Decker


technicolor

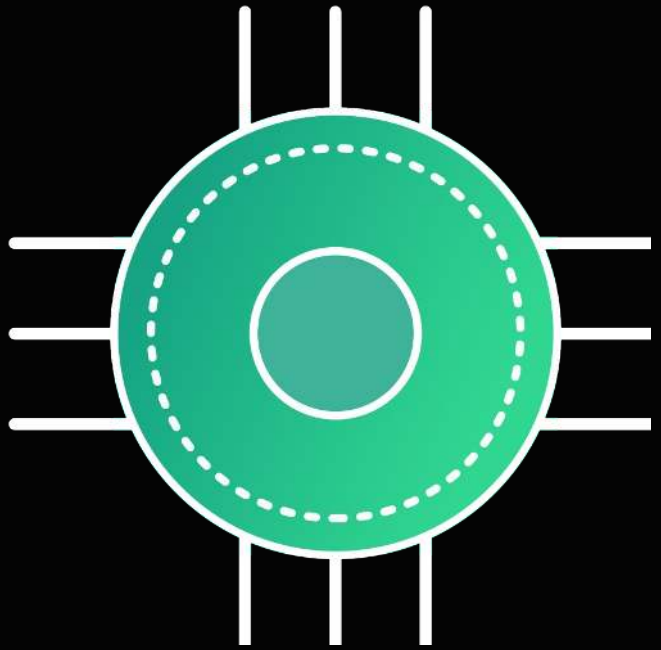
Components of AWS Greengrass

Greengrass Core (GGC)



The runtime responsible for Lambda execution, messaging, device shadows, security, and for interacting directly with the cloud

Greengrass Core (GGC)



GGC takes advantage of your device's compute, memory, storage, and peripherals

- Min Single-Core 1GHz
- Min 128MB RAM
- x86 and ARM
- Linux (Ubuntu or Amazon)

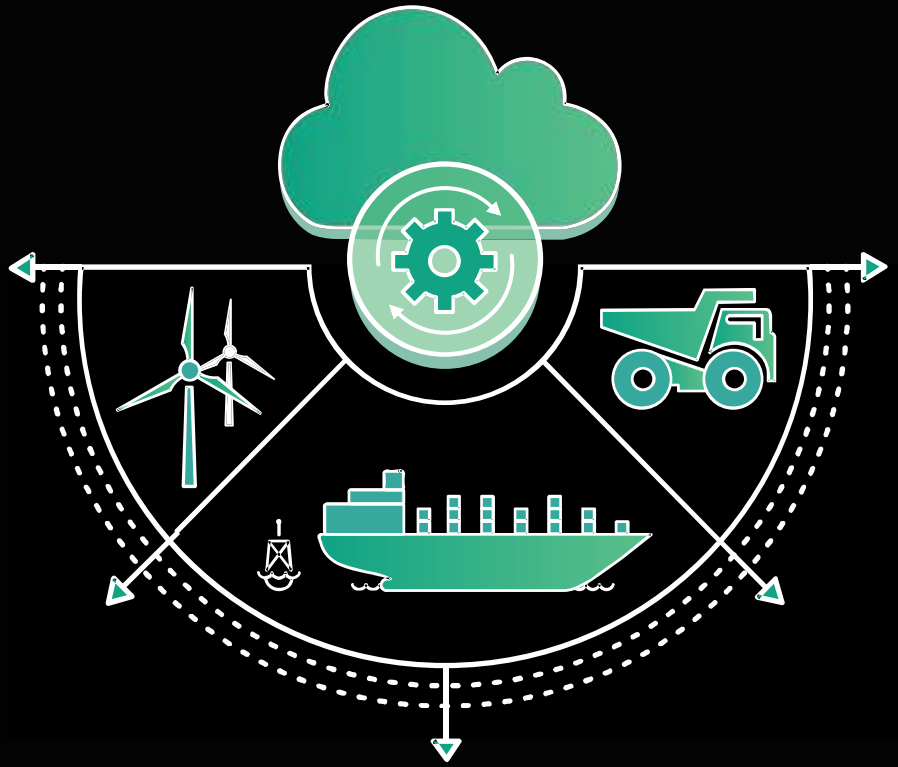
IoT Device SDK

Any device (big or small) that uses the IoT Device SDK can be configured to interact with Greengrass Core via the local network

---

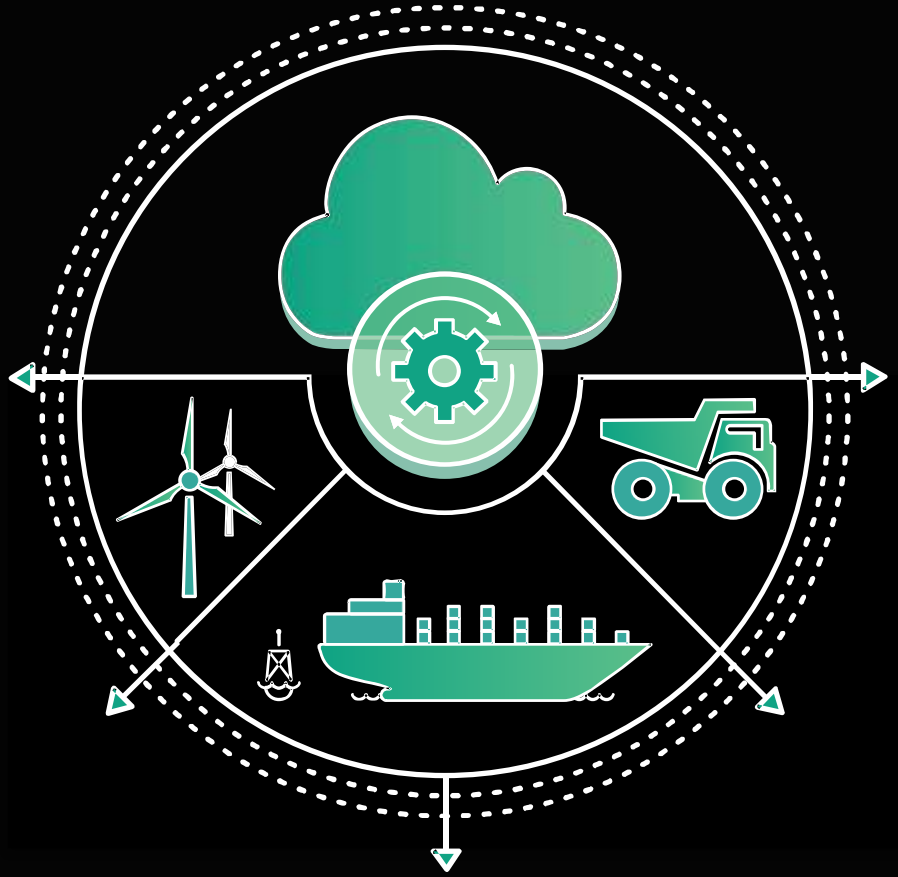
Starts with the IoT Device SDK for C++, more coming soon

Devices work together locally



A Greengrass Group is a set of Cores and other devices configured to communicate with one another

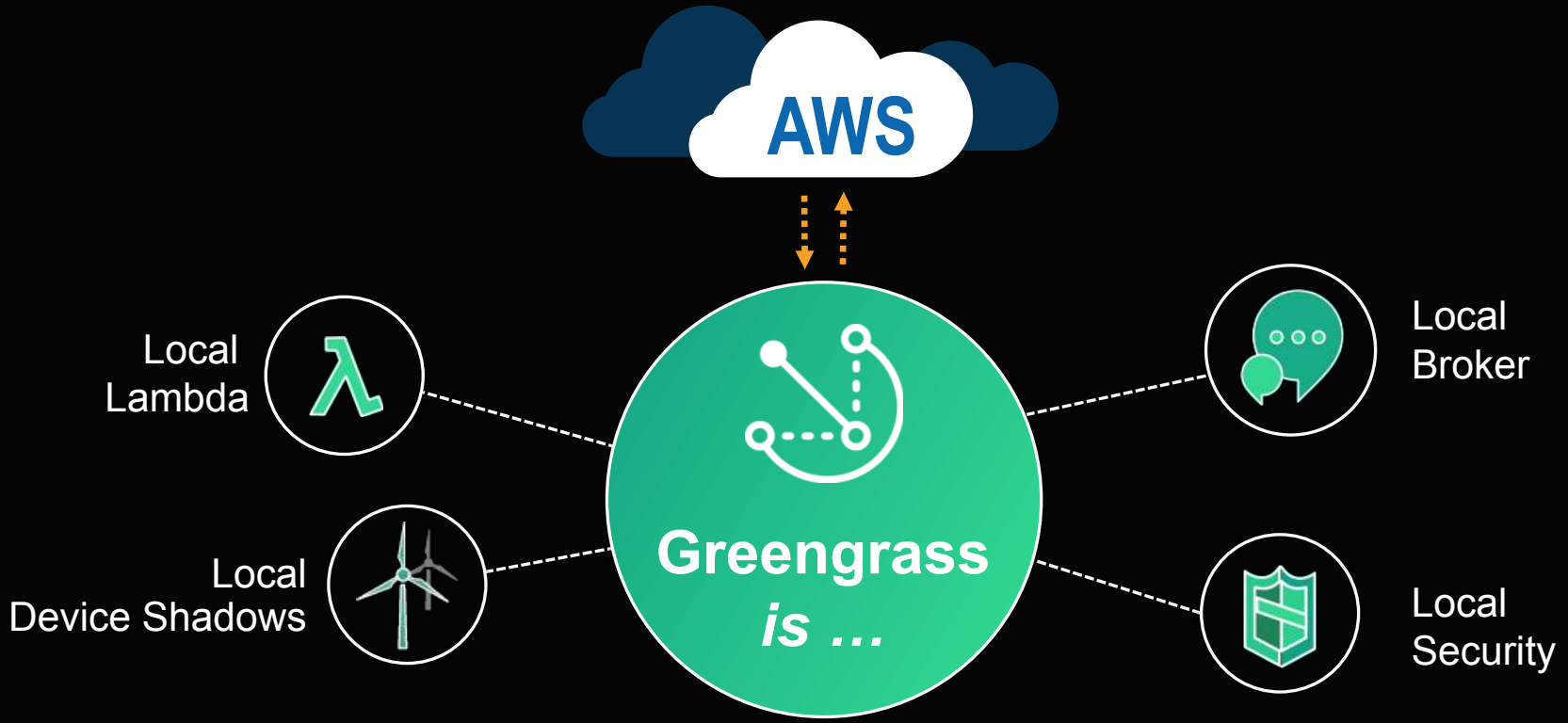
Devices work together with the cloud



Greengrass works with AWS IoT to maintain long-lived connections and process data via the rules engine

Your Lambda functions can also interact directly with other AWS services

Features of AWS Greengrass


AWS
Local
Lambda
Local
Device Shadows
Greengrass
is …
Local
Broker
Local
Security

Local Lambda

Lambda functions are event-driven compute functions

With Greengrass you can write Lambda functions in the cloud and deploy them locally

Local Lambda

Greengrass runs Lambda functions

written in Python 2.7

Invoke Lambda functions with messaging and shadow updates

Local Lambda – What you can do

Command and control

Offline operation

Data filtering & aggregation

Iterative learning

Shadows

JSON documents that represent state of your devices and Lambdas

Define them however is logical to you—a car, an engine, a fleet

Sync to the cloud or keep them local

Shadows – What you can do

Device state (current and desired)

Granular device state (only synched to the cloud for debug)

Dynamic configuration (e.g,. numeric factors of an ML model)

Messaging

Local MQTT Pub/Sub messaging

Define subscriptions between publishers and subscribers

Apply MQTT topic filters

Messaging – What you can do

Bridge to the cloud

Local distributed system

Security

Mutual auth, both locally and also with the cloud

Certificate on your devices can be associated to SigV4 credentials in the cloud

You can directly call any AWS service from AWS Greengrass

Greengrass pricing

Active Devices	Price
3	Free for 1 year
3–10,000	From $0.16/month From $1.49/year
10,000+	Contact us

Now it's your turn!

Whitepaper: "Core Tenets of IoT"
https://d0.awsstatic.com/whitepapers/core-tenets-of-iot1.pdf

Whitepaper: "Big Data Analytics Options on AWS" http://d0.awsstatic.com/whitepapers/Big_Data_Analytics_Options_on_AWS.pdf

Learn more about AWS IoT & Greengrass
https://aws.amazon.com/iot/
https://aws.amazon.com/greengrass/
https://aws.amazon.com/blogs/aws/aws-greengrass-run-aws-lambda-functions-on-connected-devices/

Learn about the AWS Free Tier https://aws.amazon.com/free/

Get started! https://aws.amazon.com/getting-started/

Thank You !

Julien Simon
Principal Technical Evangelist
Amazon Web Services

@julsimon

amazon web services