IoT challenges and EdgeX introduction

Evolving business models and mindsets

Jonas Werner
Cloud Solutions Architect
Global Alliances and Service Providers
Dell Technologies Japan
APJ use cases in 2019:
• Manufacturing
• Subway doors / sensors
• Food manufacturer
• Smart lampposts
• Surveillance

Globally:
• Power plants
• Building automation
• Plant process automation
• Retail
Mapping out the IoT & Multi-Edge challenges

Leverage SD-WAN Overlay Services

Scale Edge Resources for Demand and Latency

De-risk Future Traffic Growth and Designs

Improve response times to near-real time

Manage Performance Centrally / Key Locations

Implement Application Policies Faster

Data layer

Management layer

ON PREMISE
- DATA INGESTION
- DEVICE CONTROL
- ZERO-TOUCH
- RUN APPS

MICRO DC
- MANAGING DATA STREAMS
- DATA INFERENCING

CORE INSTANCE
- LONG-TERM DATA STORAGE
- DATA MONETIZATION

OPERATIONS
- CENTRALIZED MANAGEMENT
- SECURITY & UPDATES

MULTI-CLOUD
- FEDERATED ANALYTICS & ML
- CLOUD & ON-PREM APP LINK

Data Center

NOC

Cloud A

Cloud B

Cloud C
EdgeX Foundry
An open source IoT solution
Introducing EdgeX Foundry
Short history and background

• Chartered by Dell IoT marketing in July 2015
  • A Dell Client CTO incubation project (Project Fuse)

• Designed to meet inter-operable and connectivity concerns at the IoT edge

• Started with over 125,000 lines of Dell code

• Entered into open source through the Linux Foundation on April 24, 2017

• Now with over 2 million container downloads
Release Cadence: 2 formal releases a year

- Barcelona: Oct 2017
- California: Jun 2018
- Delhi: Oct 2018
- Edinburgh: July 2019
- Fuji: Oct 2019
- Geneva: April 2020
- Hanoi: Oct 2020
- Ireland: April 2021
- Jakarta: Oct 2021
Open source
Multi-protocol
Containers / microservices
Data ingestion from anywhere
Export to any cloud / DC
Doesn’t require an agent

https://www.edgexfoundry.org/
Now Backed by 75+ Members

With more in process!
Now Backed by 75+ Members

With more in process!
EdgeX Primer - How it works

• A collection of a dozen+ microservices
  • Written in multiple languages (Java, Go, C, … we are polyglot believers!!)
  • Several commonly used library projects (common domain objects, client libraries, etc.)

• EdgeX data flow:
  • Sensor data is collected by a Device Service from a thing
  • Data is passed to the Core Services for local persistence
  • Data is then passed to Export Services for transformation, formatting, filtering and can then be sent “north” to enterprise/cloud systems
  • Data is then available for edge analysis and can trigger device actuation through Command service

• REST communications between services
  • Some services exchange data via message bus (core data to export services and rules engine)

• Microservices are deployed via Docker and Docker Compose
REST
BACNET
BLE
MODBUS
OPC-UA
MQTT
Templates:
How to get data FROM and how to send commands TO connected sensor devices
Supported Device Services (South) Interfaces today

- REST
- Modbus
- BACNet
- MQTT
- OPC-UA
- SNMP
- BLE
- Device Service SDK’s in Go and C available
One API to control any device

• Has the commands available for any managed device
• Use one API interface to control all Edge devices
• Commands triggered automatically (based on rules) or from external systems

Just like the “One ring” from Lord of the Rings
Supported Export (Northbound) Interfaces today

- HTTP/HTTPS
- MQTT/MQTT
- Google IoT Core
- Azure IoT Hub
- XMPP
- ThingsBoard IoT
- Brightics IoT
- AWS
Location of EdgeX Foundry in the ecosystem map

- **ON PREMISE**
  - Data Ingestion
  - Device Control
  - Zero-Touch
  - Run Apps

- **MICRO DC**
  - Managing Data Streams
  - Data Inferencing

- **CORE INSTANCE**
  - Long-Term Data Storage
  - Data Monetization

- **OPERATIONS**
  - Centralized Management
  - Security & Updates

- **MULTI-CLOUD**
  - Federated Analytics & ML
  - Cloud & On-Prem App Link

---

Leverage SD-WAN Overlay Services
Scale Edge Resources for Demand and Latency
De-risk Future Traffic Growth and Designs
Improve response times to near real-time
Manage Performance Centrally / Key Locations
Implement Application Policies Faster
Architecture & Technology
It's 102°. Stop the machine.
EdgeX Foundry demo
Demo environment: Logical diagram

- Humidity
- Temperature
- Light
- Distance

Rasp.Pi Sensor cluster → REST → Rasp.Pi EdgeX

CloudMQTT MQTT broker on AWS

MQTT

Precision laptop

- MQTT client
- InfluxDB
- Grafana
Video of a similar demo is available here:

https://youtu.be/WKsGv5UVftk
EdgeX Foundry quick start
