EdgeX Foundry
Introduction

Jim White
What is EdgeX Foundry?

• An open source, vendor neutral project (and ecosystem)
• A micro service, loosely coupled software framework for IoT edge computing
• Hardware and OS agnostic
• Linux Foundation, Apache 2 project
  • Started April 2017

“Edgey”: our mascot and spirit animal of the project
EdgeX connects “things” to your IT environments
Connecting Things to IT

• Two way communications with a device or sensor
  • Typically a physical sensor or device (a “thing” in IoT)
    • Examples: a vibration sensor, thermostat, camera, moisture sensor, ...
  • Can be a virtual sensor or other system made to look like a sensor
    • Example: getting the weather from the Internet or data passed from another edge gateway

• A micro service speaks a particular protocol
  • Uses that protocol to communicate with the sensor/device
  • Example protocols: Modbus, BACnet, GPIO, MQTT, REST, BLE, …
EdgeX Foundry Goals

• Build and promote EdgeX as the **common open platform unifying edge computing**

• Enable and encourage the rapidly growing community of IoT solutions providers to **create an ecosystem of interoperable plug-and-play components**

• Provide **tools to quickly create EdgeX-based IoT edge solutions**

• **Collaborate** with relevant open source projects, standards groups, and industry alliances to ensure consistency and interoperability across the IoT
A Brief EdgeX History

• Chartered by Dell IoT marketing in July 2015
  • A Dell Client CTO incubation project (Project Fuse)
• Designed to meet interoperable 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
  • Started with nearly 50 founding member organizations; today we have more than 75
• Release Cadence: 2 formal releases a year
  • Barcelona – Oct 2017
  • California – Jun 2018
  • Delhi – Oct 2018
  • Edinburgh – July 2019
  • Fuji – Oct 2019
  • Geneva – May 2020
  • Hanoi – Nov 2020
  • Ireland – June 2021 (v2)
  • Jakarta – Nov 2021 (first LTS)
  • Kamakura – May 2022
  • Levski – Nov 2022
  • Minnesota – May 2023 (EdgeX 3.0?)
  • Napa – Nov 2023 (next LTS)
  • Odessa – May 2024
EdgeX Primer - How it works

• A collection of a dozen+ micro services
  • Written in multiple languages (Go, C, Java, ... we are polyglot believers!!)

• 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 **Application 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
  • Many others services provide the supporting capability that drives this flow

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

• Micro services are deployed via Docker and Docker Compose
EdgeX Platform Architecture

Security Services
- Reverse Proxy
- Secret Store
- Additional Security Services

Application Services
- AWS
- Azure
- Google
- MQTT
- REST

Bi-Directional, Configurable Pipelines | Store & Forward | Filtering | Batching

Supporting Services
- eKuiper (Rules Engine)
- Scheduler
- Alerts & Notifications
- Additional Services

Core Services
- Core Data
- Command
- Meta Data
- Registry & Config

Device Services
- Virtual
- REST
- SNMP
- CoAP
- GPIO
- Modbus
- MQTT
- Add'l DS

System Management
- SM Agent
- EdgeX UI (demo only)
- CLI

Message bus

“Northbound” Cloud and IT infrastructure

“Southbound” OT Devices, sensors, Cameras and Control Systems

Stop the machine

“It’s 102°F

“Stop the machine
Boiling it down

• Crudely speaking, the layers (and services) of EdgeX constitute a dual transformation engine
  • 1x - Translating information coming from sensors and devices via hundreds of protocols and thousands of formats into EdgeX
  • 2x - Delivering data to applications, enterprises and cloud systems over TCP/IP based protocols in formats and structures of customer choice

<table>
<thead>
<tr>
<th>“South side”</th>
<th>APPLICATION SERVICES</th>
<th>“North side”</th>
</tr>
</thead>
<tbody>
<tr>
<td>CORE SERVICES</td>
<td>SUPPORTING SERVICES</td>
<td></td>
</tr>
<tr>
<td>DEVICE SERVICES</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
EdgeX Enables Tiered Fog Deployments

• In today’s IoT landscape, it is imperative to leverage compute, storage, network resources where ever they live
• Loosely-coupled architecture enables distribution across nodes to enable tiered edge/fog computing
• Scope includes embedded sensors to controllers, edge gateways and servers
• Quantity and function of micro services deployed on a given node depends on the use case and capability of hardware
Performance Characteristics

- **Our platform goals since our first release**
  - Be able to run on a Raspberry Pi 3 type of device
    - 1 GB RAM, 64bit CPU, at least 32GB storage space
  - **Additional “developer community” targets**
    - Startup in 10 seconds or less (post OS boot)
    - Latency for one piece of data from data ingestion to actuation will be < 1 second

- **Ireland Release – the current state**
  - **Memory**: 403MB
    - EdgeX services minus 3rd party (Consul, Vault, etc.): ~100MB
    - Minimal EdgeX Deployment: 39MB
  - **CPU**: 28% maximum
    - Minimal EdgeX Deployment: 6%
  - **Container size**: 1020MB
    - Consul takes 115MB
    - Kong and Vault take >500MB
    - Minimal EdgeX Deployment: 112MB
  - **Data from DS to export time = < 3ms**
  - **Test Platform**
    - Running on HP MP9 gateway
    - Intel Core i7-8700T processor @2.4GHz
    - 16GB RAM
    - Ubuntu 20.04 LTS OS
  - **Test includes Redis, Consul, Kuiper, Kong, Vault and 2 device services (REST and virtual device)**
Project Momentum

- Over 225 EdgeX contributors
- > 8.2 million container downloads
- ~ 870K deployments
- ~ 2-3K website users a month
- ~20-30 global contributors per month
- > 200 commits per month
- See Insights | Linux Foundation for more details
EdgeX Foundry Adoption

- EdgeX is incorporated into these products today
  - IOTech Systems
    - IOTech’s Edge Xpert – the commercially supported version of EdgeX
  - Accenture AIP+
    - AIP+ is Accenture’s collection of modular, pre-integrated AI services and capabilities, designed to make it significantly easier to adopt AI
  - ThunderSoft TurboX Smart Core Platform
    - Platform to drive faster development of innovative smart devices on the edge
  - Jiangxing Intelligence EdgeBox
    - An advanced AIoT edge computing system
  - HP – HP Engage Edge – the world’s first retail product powered by the EdgeX Foundry
  - Tibco Project Air
    - Enables centralized access and management of IoT devices, efficient processing and storage of IoT derived data, and support for running analytics both at the edge and in the cloud
  - HomeEdge – open source project under LF Edge
    - A robust, reliable and intelligent home edge computing open source framework

- See the [EdgeX Vertical Solutions Working Group](https://community.lfedge.org/) for details and recordings from these groups
Where is EdgeX Foundry used?

- **Utilities (Power, Gas, Water):**
  - Remote monitoring telemetry of supply networks and consumer meters
  - Supply network optimisation
  - Fault detection and management
  - Predictive maintenance of plant and assets

- **Manufacturing:**
  - Manufacturing process monitoring, control and optimisation
  - Stock and materials tracking
  - Predictive maintenance of plant and assets
  - Worker health and safety

- **Retail:**
  - Fraud / loss prevention
  - Customer buying patterns and targeted promotions
  - Real-time stock management

- **Smart Buildings & Cities:**
  - Energy use monitoring and optimisation
  - Public safety and security
  - Transportation monitoring, capacity and route optimisation
  - Environment / incident response

- **Oil & Gas / Mining / Chemicals:**
  - Process monitoring, control and optimisation
  - Material flow optimisation
  - Predictive maintenance of plant and assets
  - Worker health and safety

- **Transportation:**
  - Route monitoring / optimisation
  - Semi-autonomous vehicle control
  - Predictive maintenance of vehicles
Resources

• Code
  • github.com/edgexfoundry

• Docs
  • docs.edgexfoundry.org (note selection for each version)

• Slack
  • edgexfoundry.slack.com (lots of channels– use #general when you are not sure)

• EdgeX YouTube Channel
  • www.youtube.com/channel/UC30Dbbs0qbkGJYJ5omJJnsng
  • Just search for EdgeX Foundry

• Social Media
  • Email Forum🔗 https://lists.edgexfoundry.org/mailman/listinfo
  • Twitter🔗 https://twitter.com/EdgeXFoundry
  • LinkedIn🔗 https://www.linkedin.com/company/edgexfoundry/
Questions And Some Answers
A Linux Foundation project under LF EDGE