Project Introduction

November 2018
The IoT market is inherently heterogeneous...

**Domain Expertise**
- Analytics
- Data Mgmt
- Security
- System Mgmt
- Services

Many different tools and skill sets are required to address myriad industry verticals and use cases.

**Connectivity**
- ZigBee
- AllJoyn
- EAVE
- OPC UA
- MQTT
- OPC UA
- Modbus
- Modbus
- Bacnet
- IEEE 802.15.4
- KNX
- MDE
- zigbee
- 802

IoT standards work is progressing, but there will always be widespread fragmentation in connectivity.

**Application Environments**
- Java
- Python
- JS
- .NET
- Go
- C/C++

Variable preferences for coding and application environments among developers.

**Operating Systems**
- Ubuntu
- Windows
- Zephyr
- Pulsar Linux
- VxWorks
- Yocto
- Debian
- Red Hat
- SUSE

No line of sight to consistent choices across Linux, Windows and embedded/RTOS variants.
… and the majority of the challenges are at the Edge.

- Hundreds of protocols
- Mix of IP and non-IP connectivity
- Widely distributed computing nodes, often in unsecure areas
- Need for real-time response, regardless of backend connectivity
- OS fragmentation

- Broad protocol standardization
- Entirely IP-based connectivity
- Wide use of APIs
- Computing generally in physically secure locations
The Fragmented IoT Ecosystem

The IoT landscape is characterized by many software platforms reinventing the same foundational elements that also tend to lock end users in to one cloud. In order to scale, the market needs a common foundation to bring together innovative applications, domain knowledge, and services.
Sticky note from an early project solution brainstorm, in response to the question:

“What are we trying to accomplish as a community?”
EdgeX Foundry™ is a vendor-neutral open source project hosted by The Linux Foundation building a common open framework for IoT edge computing.

At the heart of the project is an interoperability framework hosted within a full hardware- and OS-agnostic reference software platform to enable an ecosystem of plug-and-play components that unifies the marketplace and accelerates the deployment of IoT solutions.

Architected to be agnostic to protocol, silicon (e.g., x86, ARM), OS (e.g., Linux, Windows, Mac OS), and application environment (e.g., Java, JavaScript, Python, Go Lang, C/C++) to support customer preferences for differentiation
Bridging Standards with An Ecosystem of Applications

The EdgeX framework enables developers to **decouple their preferred Edges and Clouds** as close to the physical world as possible, **minimizing reinvention of table stakes elements** for data ingestion, security and management and benefitting from **flexibility and choice** (e.g. build vs. buy from the open ecosystem).
It’s All About the APIs

• **Loosely-coupled microservices** bound by common APIs established through vendor-neutral collaboration in Linux Foundation

• **HW-, OS- and Protocol-agnostic**

• Polyglot: microservices can be written in any programming language (e.g. Java, Python, Go Lang, C) and deployed in containers or VMs

• **Granularity in API definition facilitates bringing together heterogeneous OSS and Commercial offerings for various functions**

• Once key APIs are established, **entire subsections can be replaced, combined, etc. with proprietary, differentiated “EdgeX-compliant” offerings**, even Core Services

EdgeX is architected to enable **commercial value-add** around a lowest common denominator interoperability framework.
EdgeX Enables Distributed Edge Deployments

• **Architected for tiered edge computing:** Loosely-coupled architecture enables distribution across nodes
  • Applicable for devices ranging from smart sensors to controllers, edge gateways, routers and servers

• **Highly scalable:** quantity and function of microservices deployed on a given node depends on the use case and capability of hardware
  • Extends Cloud-native and “Function-aaS (FaaS) models to the Edge for flexibility in deployment and lifecycle management

• **Not just for gateways:** Entire platform can run on one node or be distributed across many nodes
  • Discrete Device Services can be run on capable smart sensors/systems and communicate directly with other backend systems, including the cloud

Unlike other OSS frameworks that are either too rigid or cloud-centric, EdgeX Foundry was architected from the ground up to enable an open distributed edge computing ecosystem
Benefits to Key IoT Stakeholders

**Hardware OEMs**

(Examples: Controllers, Hubs, Routers, Gateways, Servers)

- Scale faster with an interoperable partner ecosystem and more robust security and system management

**ISVs**

- Interoperate with 3rd party applications and hardware without reinventing connectivity

**Sensor/Device Manufacturers**

- Write a device driver with your selected protocol once using the SDK and get pull from all Solution Providers

**System Integrators**

- Get to market faster with plug-and-play ingredients combined with your own innovations

**End Customers:** Less confusion and faster time to ROI

edgexfoundry.org | @edgexfoundry
Key EdgeX project accomplishments since April 2017 launch

• Bi-annual release roadmap established and first three release dates met
• Now 75+ individual code contributors, 5X increase from January 2018
• Refactored entire code base to Go Lang
  • Full seed platform was ~2.5GB memory, booted in minutes; now ~200MiB and boots in ~6 seconds
• Initial security + management features, with California and Delhi releases
• IIC alliance formed and first IIC test bed in process from Wanxiang Group
• Entire documentation base refreshed @ https://docs.edgexfoundry.org/
• Now at 65+ project members with numerous marquee names joining in early 2019
• Increasing number of end customer PoCs in various industries
• Numerous tech providers integrating into commercial offers
  • IOTech announced Edge Xpert (Red Hat model) and xRT as a licensed real-time variant

State of the Union Blog from January 2018
Key EdgeX project accomplishments since April 2017 launch

Code releases:

- Barcelona
- California
- Delhi

Debut of Community Demonstrator (Oct 2018):

- Building Automation (More use cases to come)
- Serves as an evolving foundation for plug-fests and test bed efforts

Community and Commercial Dev Kits launched:

- https://www.edgexfoundry.org/devkits/community-devkit/
- Further accelerate adoption and build the foundation for an open IoT marketplace!
EdgeX Foundry In the Media

- EdgeX Foundry launches new developer kits for IoT
- Looking Back at EdgeX Foundry’s First Year
- EdgeX Adds Security and Reduces Footprint with California Release
- EdgeX Is a Step Forward in Addressing IoT Edge Complexity
- EdgeX Foundry Delivers Framework for IoT Edge Computing
- From Fuse to Foundry: The New Meritocracy in the Evolution of the IoT Edge
- EdgeX Foundry Is the Solution the IoT World Desperately Needs
- EdgeX Foundry and the Quest for Multivendor Interoperability
- EdgeX Foundry Unifies the IoT Marketplace to Accelerate Enterprise IoT Deployments
- Moor Insights White Paper

Much more public coverage of EdgeX can be found online.
EdgeX Foundry Videos

- [Getting started on EdgeX Foundry](#)
- [Introduction to EdgeX Foundry](#)
- [Enabling a Cloud-native Edge for IoT Scale](#)
- [Building an Open IoT Solution with EdgeX Foundry and The Zephyr Project](#)
- [EdgeX Tech Talk Series](#)

More videos can be found on [EdgeX Foundry’s Youtube Channel](#)!
Come see EdgeX in person!

- **FIWARE Global Summit**, November 27-28, Malaga
  - Janko Isidorovic (Mainflux) - Keynote on November 27 - Enabling a Cloud-Native Edge for IoT Scale
- **Internet of Manufacturing Southwest** - December 4-5, Dallas
  - Trevor Conn (Dell) - panel opportunity
- **International Congress on Management and Technologies**, December 8-13, Brazil
  - Dalton C G. Valadares (Federal University of Campina Grande) - Edgex Foundry: plataforma baseada em serviços para aplicações IoT
- **Emerging Computing Technology Conference**, January 29-30, 2019, Houston
  - Keith Steele (IOTech) and Jim White (Dell) - January 29 at 4:15 pm
- **IoT Evolution Expo**, January 29 - February 1, Fort Lauderdale
  - Jim White (Dell) - TBD - Edge Computing: Remotely acting on mission-critical decisions in the field
Lots in the works!

- Launch of **additional Vertical Solution Working Groups**
  - Targets: Building Automation, Transportation, open to others
- **Website and messaging refresh**
- Several **marquee names joining as new project members**!
Project Detail
Target Bi-Annual Release Roadmap

First three releases were delivered by the community per the committed schedule.

- 'Barcelona' (Released Oct 2017)
  - Improved fit and finish, formalized Core Service APIs, additional Device and Export Services, test apparatus

- 'California' (Released Jun 2018)
  - First integration of security, Java to Go code base, run in < 256MB RAM, come up in < 10 sec

- 'Delhi' (Released Oct 2018)
  - First manageability capabilities, Go / C device service SDKs & sample device services, EdgeX UI

- 'Edinburgh' (Apr 2019)
  - Support binary data, certification program, improved and more scalable northbound connectors / application services, additional southbound connectors to common protocols and devices

- 'Fuji' (Oct 2019)
  - Distribution support/east-west support, more management capability, ARM 32 support

Released Oct 2017
Released Jun 2018
Released Oct 2018
Apr 2019
Oct 2019
Delhi Release - Major Themes & Objectives

• Released October 2018
• High level scope
  • Initial System Management APIs and agent
  • Device Service SDKs (Go/C) and some example device services
  • The next wave of security features (access control lists to grant access to appropriate services, and improved security bootstrapping, …_)
  • Improve testing (better/more unit, complete black box testing)
  • Refactored and improved Go Lang micro services
  • An EdgeX UI suitable for demos and smaller installations
Edinburgh Release – Major Themes & Objectives

• Releasing April 2019
• Ratified during EdgeX TSC Face-to-Face meetings in UK, Nov 2018
• High level scope
  • Improved on-boarding for EdgeX Users (docs, tutorials, dev kits, etc.)
  • Support of ingestion, use, export of binary data (via CBOR format)
  • Automate performance testing, automate security testing
  • Add many device services (improving out-of-the-box southbound connectivity)
  • Provide application services – a more scalable and flexible exportation capability (eventually replacing the existing export services)
  • Refactor database-using services to be more loosely coupled to the persistence mechanism (allow for use of alternative persistence stores and technologies in future releases)
  • Outline a certification program for micro service drop in replacements
Fuji Release – Anticipated Major Themes & Objectives

• Releasing October 2019
• To be ratified during EdgeX TSC Face-to-Face meetings in Seoul, April 2019
• Anticipated high level scope
  • Better distribution support, initial east/west support (load balancing, service failover, dynamic workload allocation, etc.)
  • System management control plane extensions via alternate protocols
  • System management alerts/notifications
  • Support to securely provisioning new devices/sensors
  • Securing service to service communications with AuthN/AuthZ
  • Dynamic configuration, configuration testing
  • Improve application services and expanded north side connectivity
  • Official Arm 32 Support and Windows development support
EdgeX Performance Targets

• The target is to run all of EdgeX on a low end Raspberry Pi type of device: 256MB RAM, 64bit CPU, <32GB storage space
  • Note: current release roughly meets this target, with half of the memory utilization being consumed by the MongoDB reference database

• Additional “developer community” targets
  • Startup in 5 seconds or less (post OS boot)
  • <300ms round trip from data ingestion to actuation through reference rules engine

• Remaining OS and Hardware agnostic
  • Windows, Linux, *nix, …
  • Intel/Arm 64/Arm 32
Planned EdgeX Security Modules

• With the EdgeX framework and associated APIs end users benefit from a global network of security experts defining layer upon layer of defense-in-depth in four key areas

• Phased development approach, based on priority for securing overall stack
  1. Define standards to leverage and API requirements for each module
  2. Develop lean reference implementations for the open source code

• As with rest of platform, security reference modules will be replaceable with proprietary value-add
In process for the late October Delhi release is the first management functionality

EdgeX is leveraging an aggregator service – the System Management Agent (SMA) – to facilitate control plane communications with heterogeneous management systems

Provides flexibility for various OSS and commercial management tools/consoles, both local and remote and leveraging various protocols (e.g. SNMP, LwM2M)

Translates commands from the external management system to the internal EdgeX microservice management APIs

It translates EdgeX control plane data / metrics into a format/protocol that external management systems can understand

The SMA also provides grouping functions

e.g. stop all EdgeX microservices at once (vs. an individual microservice stop command)

The SMA will have a well-defined API
The EdgeX Management API

Each EdgeX microservice will have a control/management API exposed to the SMA.

The SMA relays management commands from the desired remote/local console, e.g.:

- Start service
- Stop service
- Restart services
- Get configuration
- Get metrics
- Determine QoS needs
- Etc.

The API is meant to be an internal API, exposed only to the SMA, but could be used by other management systems.
Summary of Deployment Patterns

**IT**
- Cloud/ Data Center
- Edge Servers/ “Fog Nodes”
  Memory: 16GB+
- Edge Gateways
  Memory: 2GB+
- Open Source Baseline
- Proprietary EdgeX-compliant Offers

**OT**
- Cloud
- Hard-Real-Time
- Responsive
- Seconds to days
- Milliseconds to seconds
- <10ms, deterministic
- Field Devices
  - PLCs, PACs, Microcontrollers
  Memory: <10MB
- General-Purpose Edge Gateway
- On-prem with EdgeX-enabled PLCs
- High-Bandwidth Streaming Analytics
Real Time Enabled Via Code Extensions

<table>
<thead>
<tr>
<th>Through Community Extensions</th>
<th>Open Source Baseline</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hard Real Time</td>
<td>Soft Real Time (“Relevant Time”)</td>
</tr>
</tbody>
</table>

**Response Time**
- High bandwidth, QoS, sub-millisecond, deterministic
- Milliseconds+

**OS**
- RTOS
- Traditional Linux or Windows

**Example Use Cases**
- Smart Building, Energy Management, Factory Optimization, Predictive Maintenance, Quality Control, Supply Chain Management, Remote Asset Management, Fleet Management, Logistics, Environmental Monitoring
- High-speed Process Control, Robotics, Safety Systems, Autonomous Vehicles
Opportunities for Proprietary Performance Extensions

- Targeted at time-sensitive use cases requiring low footprint (e.g. <10MB), high streaming bandwidth, and predictable low latency (e.g. <10ms)
- Enabled via lightweight, EdgeX-compliant instances of Core Services and pluggable high performance data/message bus for intercommunication between microservices
- Compatible with foundational EdgeX APIs to benefit from broader ecosystem

Lightweight, high performance version of Core Services leveraging the key EdgeX APIs at the perimeter. Entire core can be compressed to a proprietary C-based binary and still benefit from plug-in EdgeX-compliant services from the community.
What’s with that ‘X’?

Fundamental goal of the EdgeX project is to provide a stable, product-quality open source foundation for interoperable commercial offers

• The ‘X’ in EdgeX allows the project name to be trademarked for use as a certification mark

• A certification program will be established in the project for commercial offerings to verify that key EdgeX interoperability APIs were maintained alongside proprietary value-add
  • Initial program launch targeted for ‘Edinburgh’ release in April 2019 with ramp through 2H 2019

• Stability for key elements (e.g. core APIs and certification process) is maintained through the EdgeX Technical Steering Committee (TSC) and clear versioning system

• Licensed under Apache 2.0, anyone can leverage the EdgeX code base as a foundation for their commercial offerings
  • Can be a full EdgeX-compliant IoT platform, value-added plug-in microservice(s) or a services model
Key Project Links

Access the code: https://github.com/edgexfoundry

Access the technical documentation: https://docs.edgexfoundry.org/

Access technical video tutorials: https://wiki.edgexfoundry.org/display/FA/EdgeX+Tech+Talks


Join an email distribution: https://lists.edgexfoundry.org/mailman/listinfo

Join the Rocket Chat: https://chat.edgexfoundry.org/home

Become a project member: https://www.edgexfoundry.org/about/members/join/

LinkedIn: https://www.linkedin.com/company/edgexfoundry/

Twitter: https://twitter.com/EdgeXFoundry

Youtube: https://www.youtube.com/edgexfoundry
Project Organization and Membership
EdgeX Foundry Governance Structure

EDGEX FOUNDRY MEMBER COMPANIES (60+)

GOVERNING BOARD (GB)
Composed of appointed and elected individuals; manages the business of the EdgeX Foundry.

TECHNICAL STEERING COMMITTEE (TSC)
Leads the technical work of EdgeX Foundry. Oversees and aligns working groups.

CERTIFICATION COMMITTEE
Develops and oversees the certification program for EdgeX Certified components.

LF SUPPORT TEAM
EdgeX Project Organization

EdgeX Foundry Technical Steering Committee

Working Groups
- Core Working Group
  - Chair: Trevor Conn
  - Dell
- Device SDK Working Group
  - Chair: Steve Osseleto
  - IOTech
- Applications Working Group
  - Chair: Janko Isisborovic
  - Mainflux
- System Management Working Group
  - Chair: Jim White
  - Dell
- Security Working Group
  - Chair: David Ferriera
  - ForgeRock
- Vertical Solutions Working Group
  - Chair: Moonki Hong
  - Samsung
- DevOps
  - Chair: Jeremy Phelps
  - Linux Foundation
- QA and Test
  - Chair: Andrew Foster
  - IOTech

Projects
- Core MVP
- Device SDK MVP
- Export Services SDK MVP
- System Management Services MVP
- Security Services MVP
- Smart Factory
- Oil and Gas
- Retail
- Continuous Integration MVP
- Testing MVP

Additional use-case specific projects

TSC At-Large Members
- Tony Espy
  - Canonical
- Drasko Draskovic
  - Mainflux

edgexfoundry.org | @edgexfoundry
Engagement Options

• Project is a technical meritocracy. Anyone can contribute to or use the EdgeX Foundry code for free.

• Technical Steering Committee (TSC) and Working Group (WG) meetings are open to the public

• TSC and WG Chairs in addition to code committers and maintainers are voted in based on technical acumen and alignment to project tenets. This ensures robustness and stability in the architecture, technology choices, roadmap and code base.

• Joining as a paid project member affords maximum influence over project direction
Member Benefits

• Additional influence to shape the overall platform architecture to enable commercialization needs

• Recognition for Industry thought and technology leadership

• Marketing and networking within the EdgeX Project for business opportunities (effectively a vendor-neutral partner program)

• Discounted sponsorships at Linux Foundation and EdgeX Foundry-produced events (e.g. trade shows, hackathons, etc.)

• Learning and engagement
Project Membership Options

- Platinum
  - Appoint one (1) representative to the EdgeX Governing Board (GB)
  - Appoint one (1) representative as a voting member in any subcommittees or activities of the GB
  - Appoint one member to the start-up TSC (6 month position)
  - Enjoy most prominent placement in displays of membership
  - Access to LF’s invitation-only Open Source Leadership Summit
  - Ongoing, individual engagement with EdgeX executive director and staff
  - $150,000 annually

- Silver
  - Participate as one of three (3) Silver representatives to the EdgeX Governing Board
  - Enjoy prominent placement in displays of membership
  - Ongoing engagement with EdgeX executive director and staff
  - $2,500 to $50,000 annually, depending on employee count

- Associate (non-profits)
  - Limited to pre-approved non-profits, open source projects, and government entities
  - Entitled to identify their organization as members supporting the mission of EdgeX and any other rights or benefits as determined by the Governing Board
EdgeX Deployment Patterns
Embedded Device Services

- Planned work will enable C-based Device Services to be embedded in constrained microcontrollers running a RTOS for hard real-time use cases (e.g. within a smart sensor or PLC)
- Due to loosely-coupled architecture, baseline EdgeX-compliant Device Services can be deployed directly on smart sensors or systems capable of hosting a microservice (via container or VM)
- IP-capable sensors with an EdgeX Device Service / APIs can communicate directly with Core Services running on any other compute node such as a gateway, server or directly to the cloud

Embedded Control Systems (e.g. PLCs)

Smart Sensors

Smart Systems

Edge Gateway

Edge/Fog Server

EdgeX hosted within PCF, Azure, AWS, etc. or proprietary stack leveraging EdgeX APIs
Simple Linking Device

- A minimal deployment of EdgeX can function as a linking device which simply converts one protocol into another
- Typical protocol combinations vary by vertical and installation, some typical examples:
  - Energy: DNP3 to MQTT, Modbus to REST
  - Manufacturing: Profibus to OPC-UA
  - Buildings: BACnet MSTP (serial) to BACnet IP, MQTT, etc.

Deployed Microservices:
- Single Device Service
- Core Services
- Single Export Service
- Basic security and manageability

Protocol A

“Southbound”
(format required for field device)

Protocol B

“Northbound”
(any format desired for backend application)
Full Edge Gateway Stack

Deployed Microservices:
- Multiple Device Services for data ingestion and control across heterogeneous protocols
- Local database for buffering during periods of lost connectivity
- 3rd party streaming analytics
- Various security services
- 3rd party remote management console
- MQTT Export Service

Temperature + vibration via BLE Sensors with vendor-embedded EdgeX Device Service

Voltage + current from robot arm motor via power meter, Modbus TCP over Ethernet

Process data from conveyor PLC via proprietary protocol over RS-485 Serial

Distributed I/O

PLC

MQTT Export
# Tiered Edge Deployment

Number of deployed microservices and functionality increases higher in tier

<table>
<thead>
<tr>
<th>Field Devices</th>
<th>Simple Edge GWs</th>
<th>Intelligent Edge GWs</th>
<th>Edge Servers</th>
<th>Cloud</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Room Level</strong>&lt;br&gt;  • Ingestion for local temperature and occupancy data&lt;br&gt;  • Simple rules engine to control local temperature and lighting settings</td>
<td><strong>Floor Level</strong>&lt;br&gt;  • Integration of temperature and occupancy plus add’l events from surveillance cameras and overall energy usage data&lt;br&gt;  • Basic ML / streaming analytics for reacting to local events (e.g. alert security when intruder detected)</td>
<td><strong>Building Level</strong>&lt;br&gt;  • Aggregated data for analytics of overall building performance&lt;br&gt;  • Streaming data from all floors, more complex analytics</td>
<td><strong>Portfolio Level</strong>&lt;br&gt;  • Deep learning in the cloud to optimize energy usage across entire real estate portfolio</td>
<td></td>
</tr>
</tbody>
</table>
Distributed Computing

- Introducing specific microservices to address QoS, failover between nodes, redundancy and “east-west” communication
- Workloads deployed dynamically at different tiers to optimize performance and results.
- In a manufacturing example, data can be coordinated for manufacturing process, building performance energy usage and logistics across various buildings, plants and trucks.
Example EdgeX PoCs In Process
(End Customers NDA)
Industrial Automation PoC

- Large industrial automation provider
  - Working with EdgeX to bridge legacy and new OT infrastructure to SCADA and proprietary cloud environments
  - Software stack/platform will be deployed in different operational configurations
  - Need the capability/flexibility to provide common software functions independent of the hardware configuration
  - Example: deploy the stack on a standalone very low footprint micro gateway connected directly to the cloud or distribute entire stack to a larger on-prem node

- Need the platform independence and small footprint EdgeX offers to run on their gateways

- Conducting gap analysis between existing data models to EdgeX model; exploring options for model changes or extensions

- Exploring 3rd party integration for system management
Building Management PoC

• A mid-sized building management company in Germany needs to connect legacy systems to a central IoT system to unlock near real-time data on energy spend, space utilization and occupancy
  • Highlight resource usage discrepancies
  • Make informed cost saving decisions from data collected

• They want to build advanced analytics and visualization capability on top of a common/open platform to deliver a dynamic building automation solution that integrates Lighting and HVAC systems
  • Will automatically respond to occupancy trends, people comfort and cost saving goals

• EdgeX is attractive as given the size of their IT organization, they want to automate more tasks and use an open platform in order to leverage community assets where possible

• Plan to setup notifications and alerts to be notified when system performance falls outside of expected thresholds
Oil & Gas PoC

• A global oil and gas supplier has the need to incorporate numerous sensors/devices/controllers through a real-time bus while also integrating various controller applications

• The EdgeX APIs are attractive as interoperability “glue” that brings the mix of devices/control applications together

• Hard real-time needs will be provided by an EdgeX commercialization and specialization firm through a compressed, C-based variant of the EdgeX code base

• Resulting platform still leverages the EdgeX interop APIs at the boundary to take advantages of the growing ecosystem of device and application services
Thank You!