

# Problem Statement of Edge Computing beyond Access Network for Industrial IoT

draft-geng-iiot-edge-computing-problem-statement-00

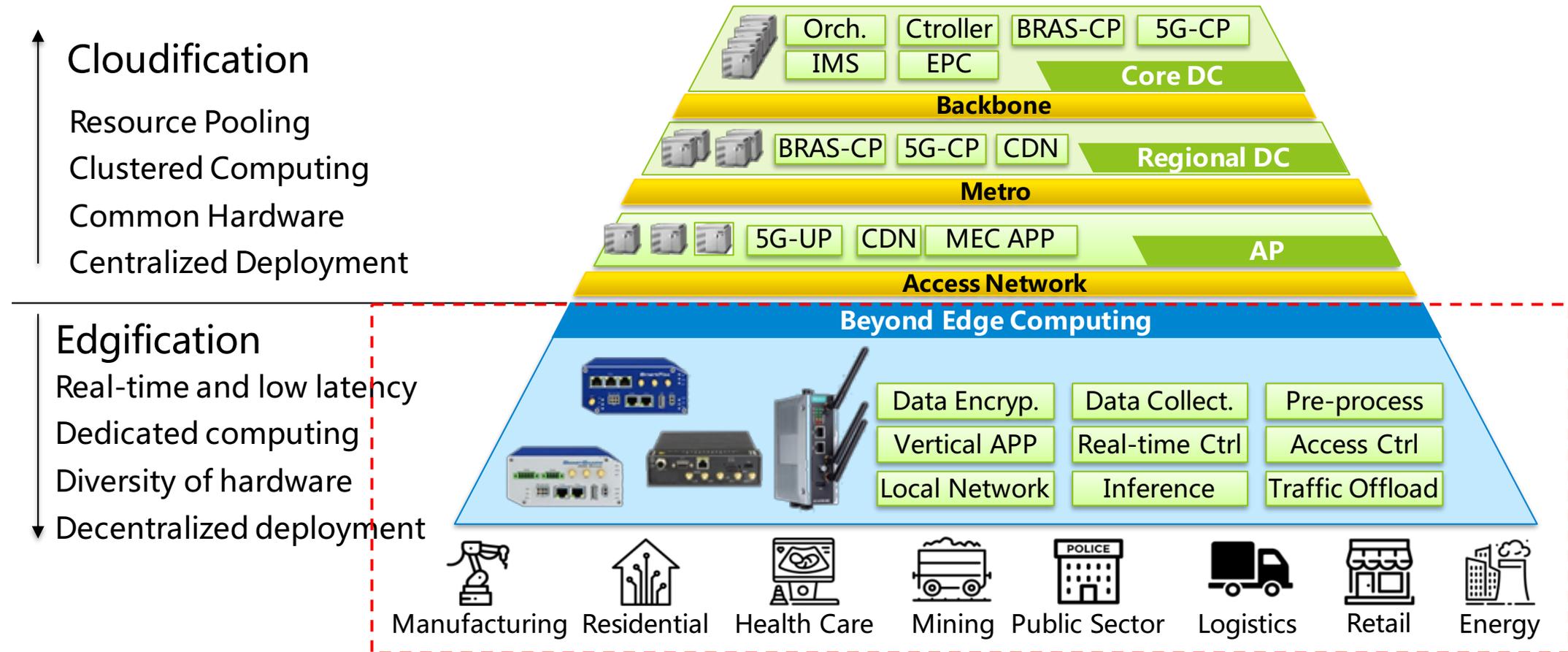
Liang Geng, China Mobile

Mingui Zhang, **Mike McBride (Presenter)**, Bing Liu, Huawei Technologies

# Background

- Focus on **use cases** and **problem statement**, related work in other SDOs and gap analysis.
- Create research discussion and standardization work for:
  - Protocol for east-west comm between multiple BEC gateways
  - Common API across various BEC platform.
  - User mobility: Edge to Edge
  - Edge device config/mgmt (both device-level and virtual-resource-level)
  - Light-weight virtualization technologies (container/unikernel)
  - Local edge security
- draft-geng-iiot-edge-computing-problem-statement-00

# What is Beyond Edge Computing (BEC)



BEC takes care of the first hop where the service of a particular industrial vertical connects to the network

# Capabilities of BEC(1/2)

- **Heterogeneous IoT device compatibility**

*Various IoT interface including Zigbee, WIFI, tens of Field Buses and etc.*

- **Low and deterministic service latency**

*Various mission-critical services (Motion control etc.)*

- **Data pre-processing and traffic offloading**

*Reduce WAN bandwidth and increase the effectiveness of data*

- **System resource isolation**

*Support multi-tenancy situation – which is a norm for IoT ecosystem*

- **Offline processing**

*Robustness even in unlikely circumstances*

# Capabilities of BEC(2/2)

- **End-to-end security**

*Security provided right from customer premises*

- **Distributed artificial intelligence**

*Making use of GPU/TPUs for AI applications (Graphic/Inferences)*

- **Real-time operation**

*Real-time operating system(i.e. Industrial Linux) for spontaneous events*

- **Unified API for multi-ecosystem edge application**

*API to expose Software/Hardware capabilities for application ecosystem*

- **Service isolation for network slicing**

*Mapping services to specific network slice for E2E service guarantee*

# Architecture

## BEC Management Platform

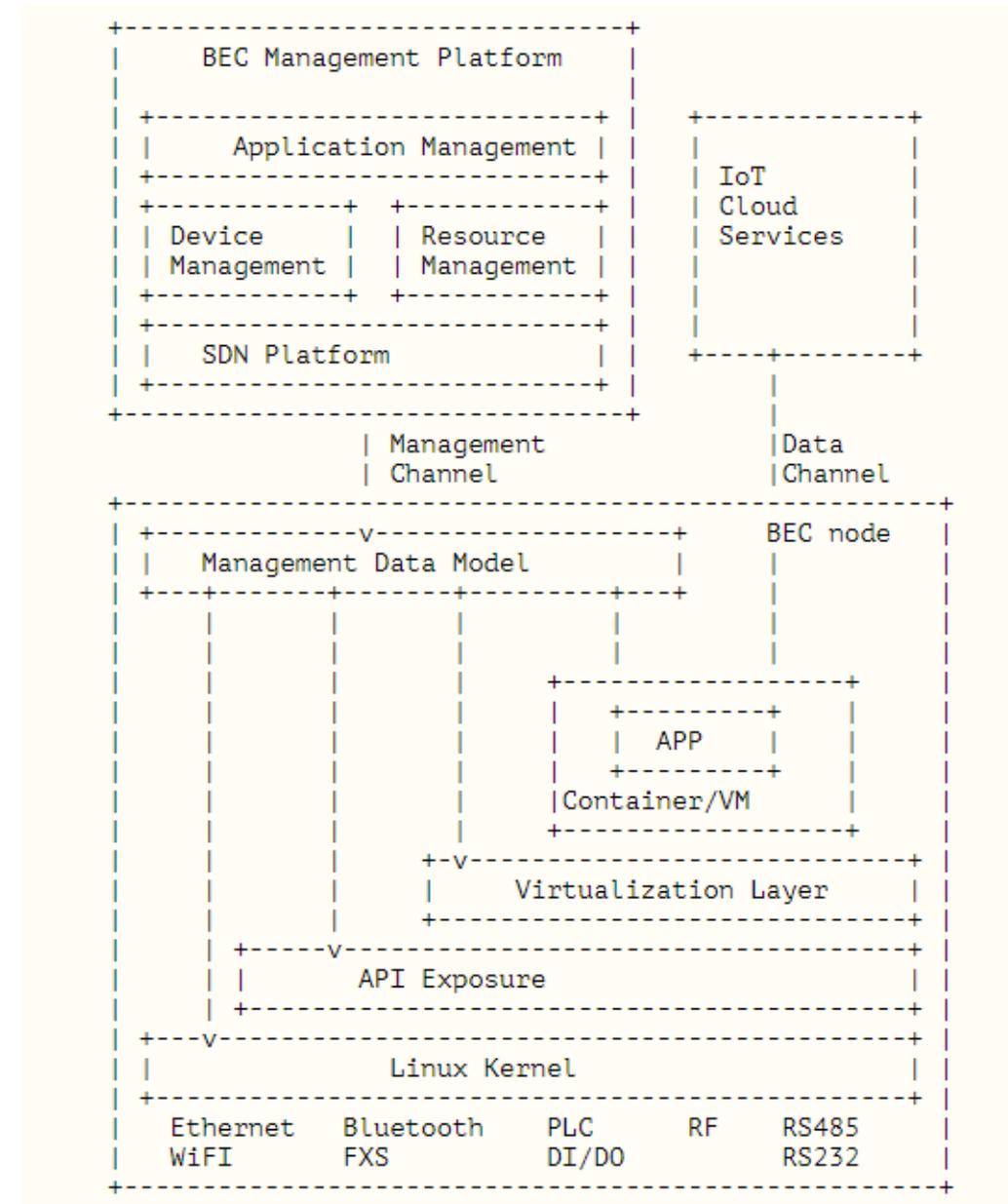
- Application management (Lifecycle, verification, publishing, Devops etc.)
- Device-level management + System resource management
- SDN platform –control and optimization for BEC network

## Interfaces

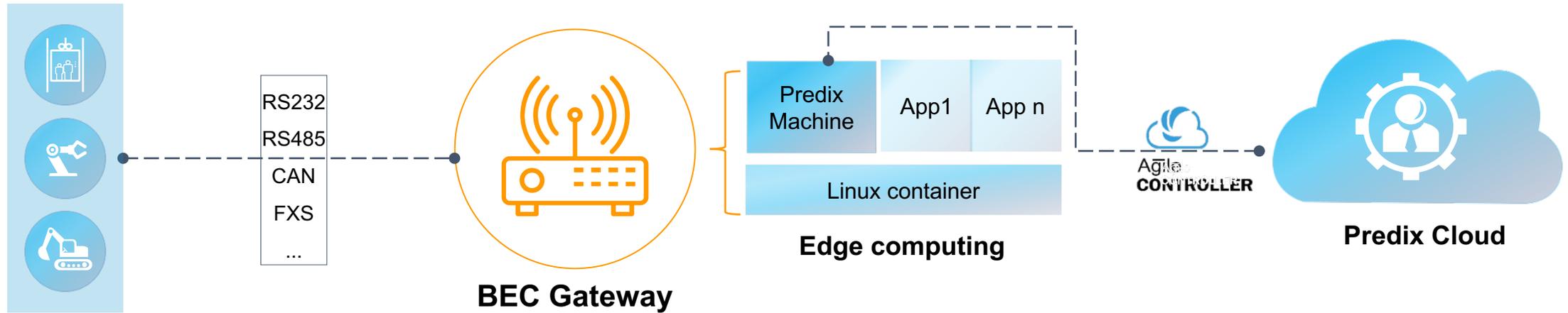
- Management interface – Lightweight interface for device configuration/monitoring and virtual resource management
- Data interface – Unified but flexible for interoperation in data plane (i.e. MQTT, OPC-UA and others)

## Distributed BEC platform

- Lightweight virtualization technology for resource isolation and fast application deployment
- Heterogeneous but modularized southbound interface
- Comprehensive and unified API for capability exposure



# Example of E2E BEC system



## Flexible industrial access

- Industry-grade design
- 17+ industrial interfaces
- Voice, Wireless, VPN in one box

## Pre-analysis of local data

- Data aggregation: industrial data analysis
- Data filtering: saving WAN traffic
- Data security: encryption

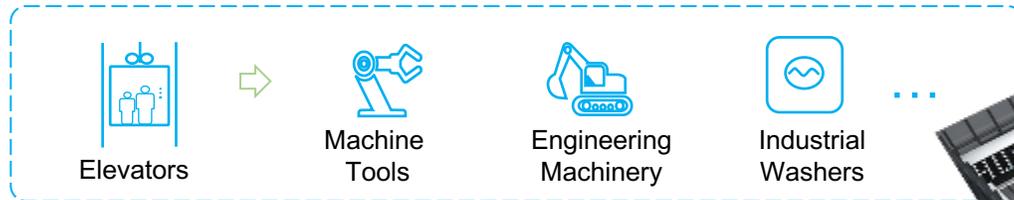
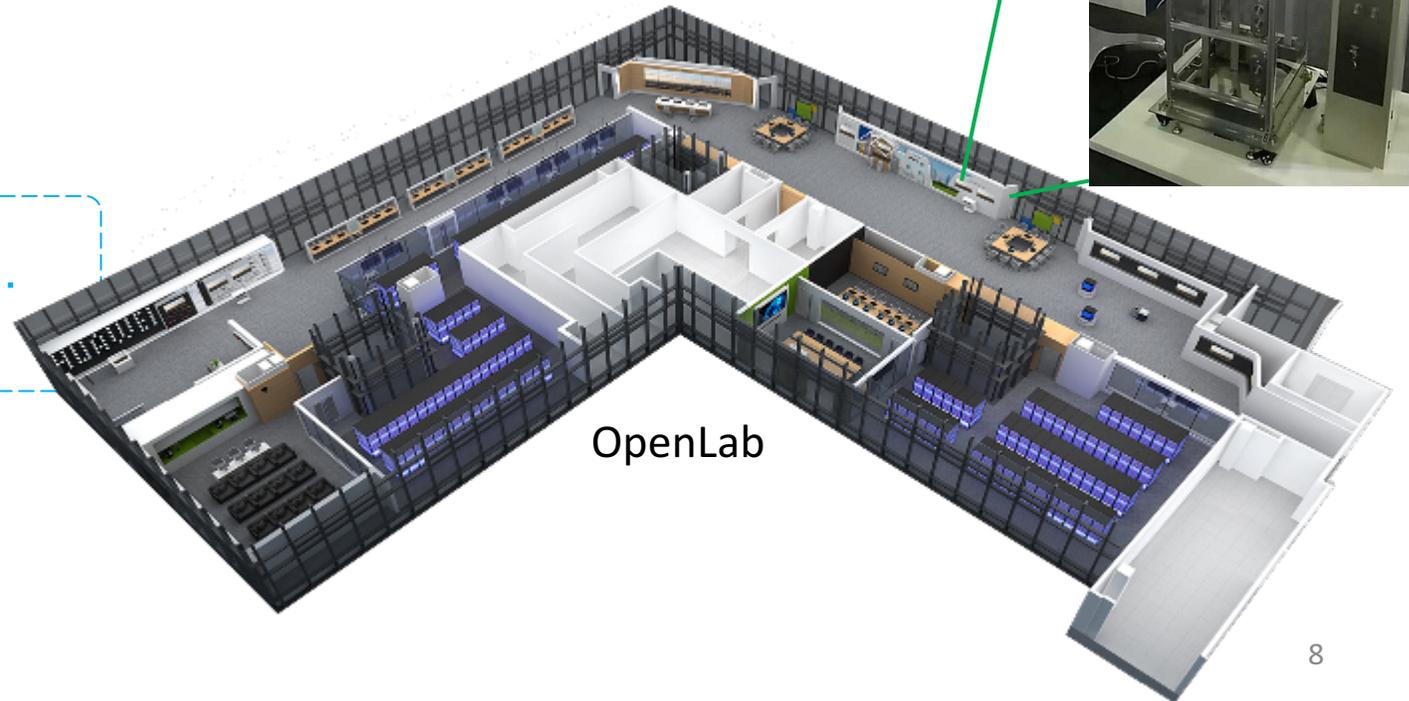
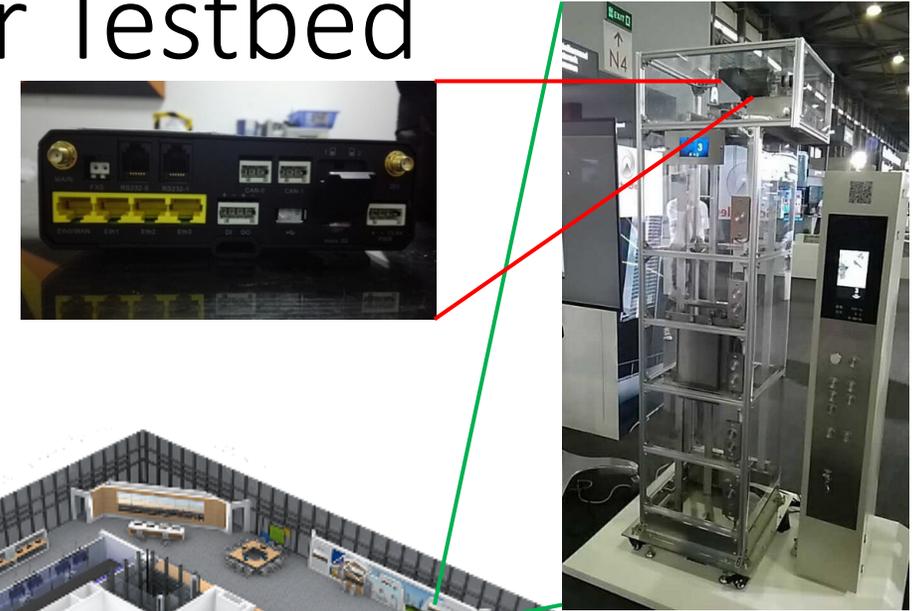
## Edge and Cloud Collaboration

- Predix and apps at edge
- Edge: real-time and reliability
- Could: unified policies and Big Data

# Use case -Connected Elevator Testbed

- **Market Segment**

- Buildings & Facilities : Elevators
- This testbed may also be applicable to other verticals such as
  - Machine Tools
  - Engineering Machines
  - Industrial Washers



- **Location**

- The OpenLab in Nanjing, China

# Gap Analysis in IETF

- **Multiple Virtualization Technologies Coexistence/Coordination**

*Interface/Model required for selection between various virtualization technologies for specific vertical use cases/requirements*

- **Light weight Device-level management and virtual resource**

*Massive number of isolated resource pool need much lighter interface for management – Netconf+YANG for VIM?*

- **Framework and APIs for multi-ecosystems**

*Unified and modularized APIs across multiple compiling environment for maximum flexibility and friendly ecosystem*

- **Runtime Updates**

*Energy efficient software/firmware update without service break-down*

# Conclusion

- BEC methodology – Distributes as much as you can, centralized only if you must (i.e. massive computing/storage).
- Research interest – very popular topic across the industry (IIC, ECC, EdgeXFoundry, OpenFog, OPENIL, RIOT and etc.)
- IETF and IRTF – Architecture and framework for RG, interface/API deliverables might be for WG (see SUIT BoF)
- Further discussion @ Bar BoF Thursday 20:00-21:30 Hullet