Blueprint: KubeEdge Edge Service Family
(Type 1: ML Inference Offloading)

Futurewei, China Mobile, ARM, Signalogic

Table of Content

AKRAINO EDGE STACK

General Blueprint Introduction

- The KubeEdge Edge Service blueprint family introduces Edge Services built on KubeEdge for various applications in mobile edge environment. The service is to enable application developers reach optimal latency/energy/performance/cost via balancing computation loads among device/edge/central DC(Cloud).
- Type I of KubeEdge Edge Service family focuses on ML Inference Offloading.
- Future types of this blueprint family will provide variations of end-to-end solution components around KubeEdge. E.g. hardware platform; additional service stack on top of KubeEdge

AKRAINO EDGE STACK

The ML Inference Offloading Blueprint Criteria

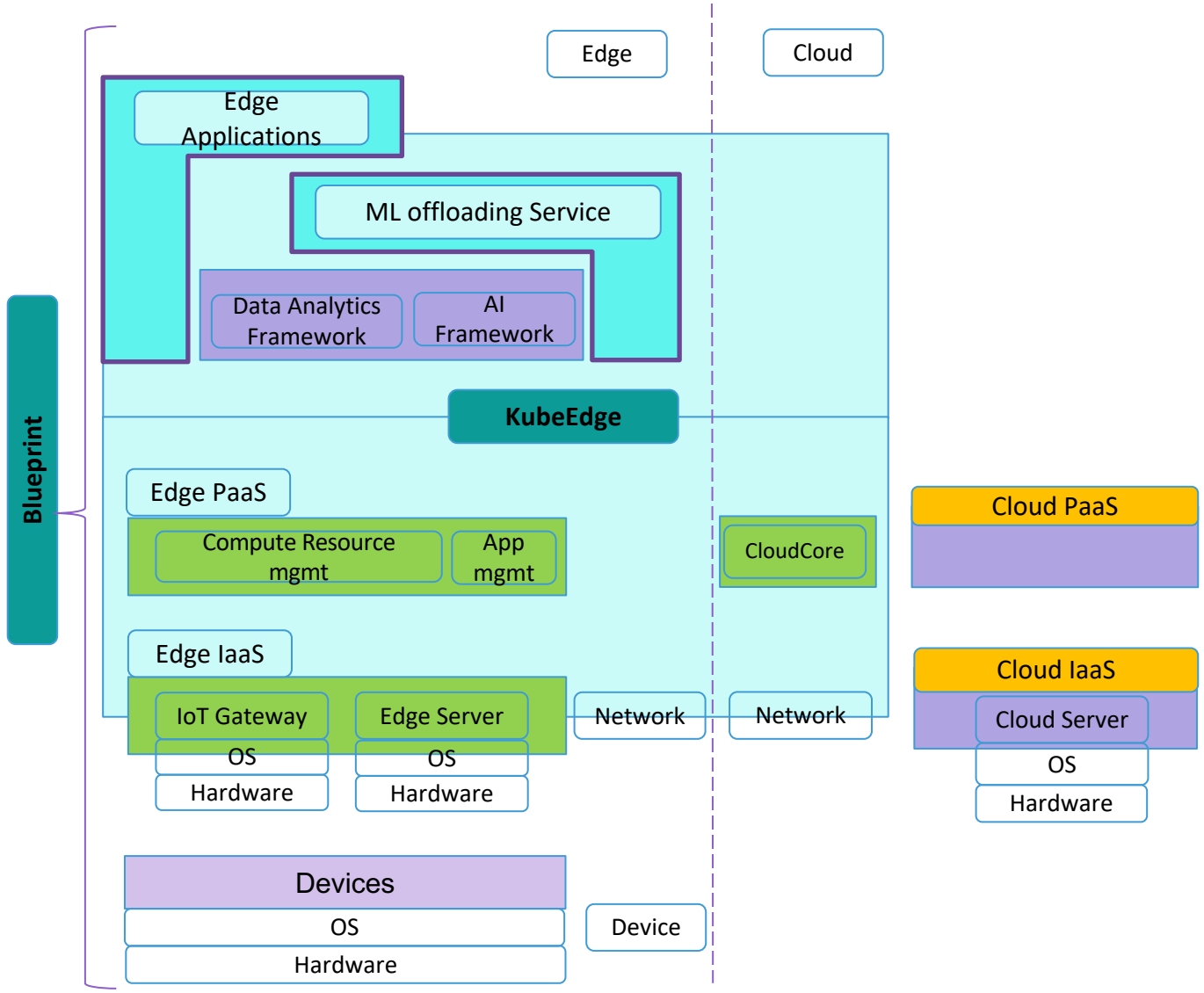
Case Attributes	Description	Informational
Type	New	
Blueprint Family	Edge offloading	
Use Case	Facial expression task offloading to edge node	
Blueprint proposed Name	The Device/Edge ML Offloading	
Initial POD Cost (capex)	Less than 100KUSD	
Scale & Type	Up to 1 servers, x86 server With nVIDIA Tesla P4/T4 GPUs	
Applications	Deep learning models（facial expression) offload from mobile device to Edge	
Power Restrictions	Less than 10Kw	
Infrastructure orchestration	Docker 18.09 OS – Ubuntu18.04 Python 3.5 ~3.7 CUDA>10.1 GPU driver release 19.03	
PaaS	Kubedge	
SDN		
Workload Type	Containers	
Additional Details		

AKRAINO EDGE STACK

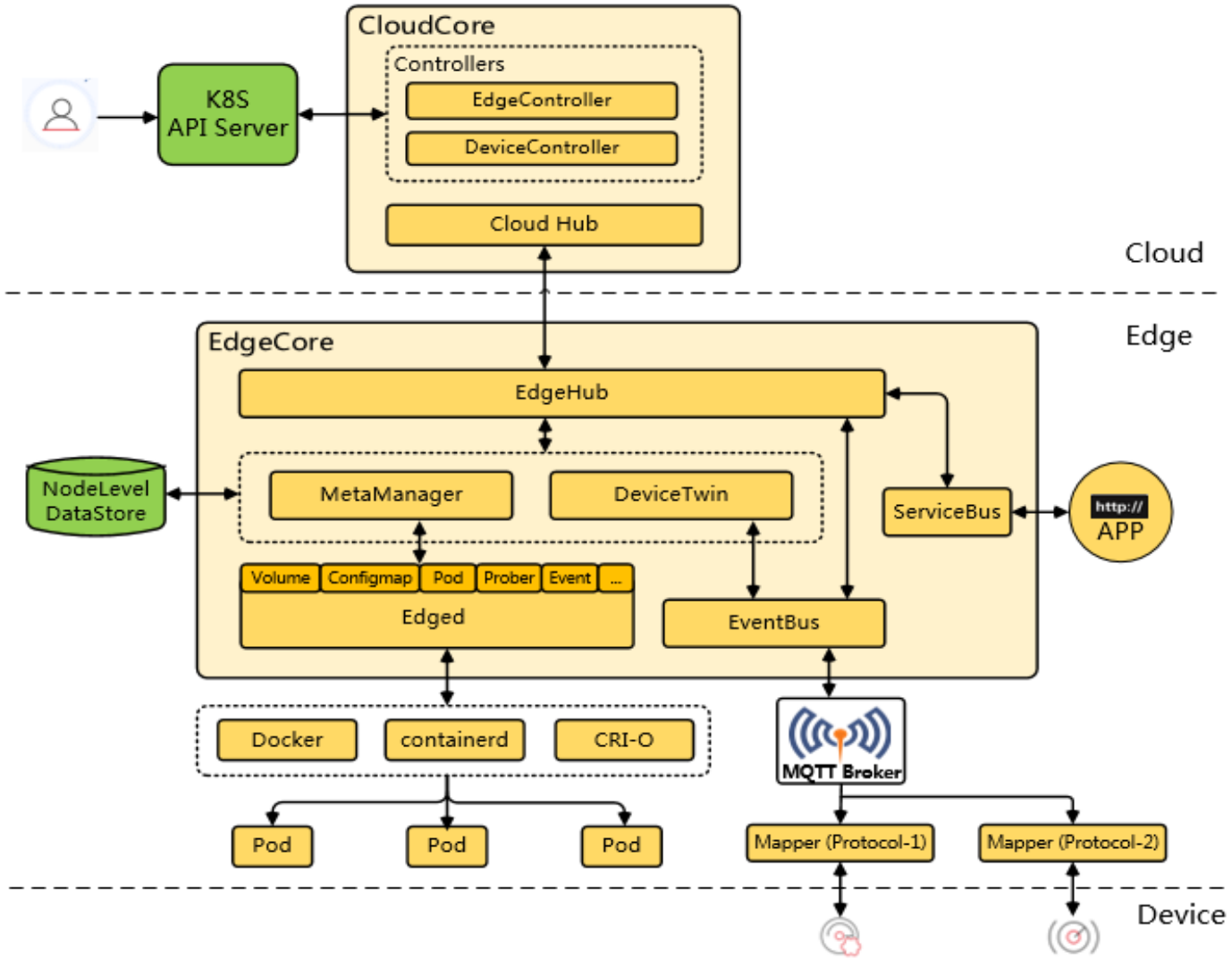
Project Background

1. KubeEdge is a CNCF sandbox project led by Futurewei, targeting at cloud/edge computing and networking.

2. Akraino is a LF Edge open source project promoting end-to-end solutions via blueprint projects. Akraino API sub-committee releases API whitepaper to market edge stacks introduced through blueprints.



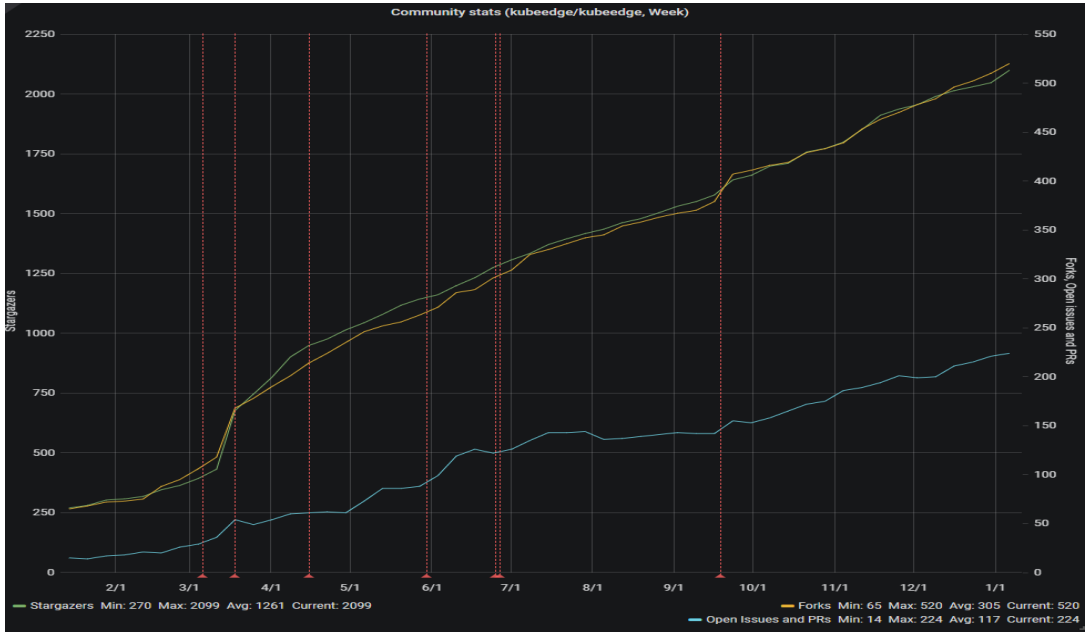
KubeEdge Infrastructure

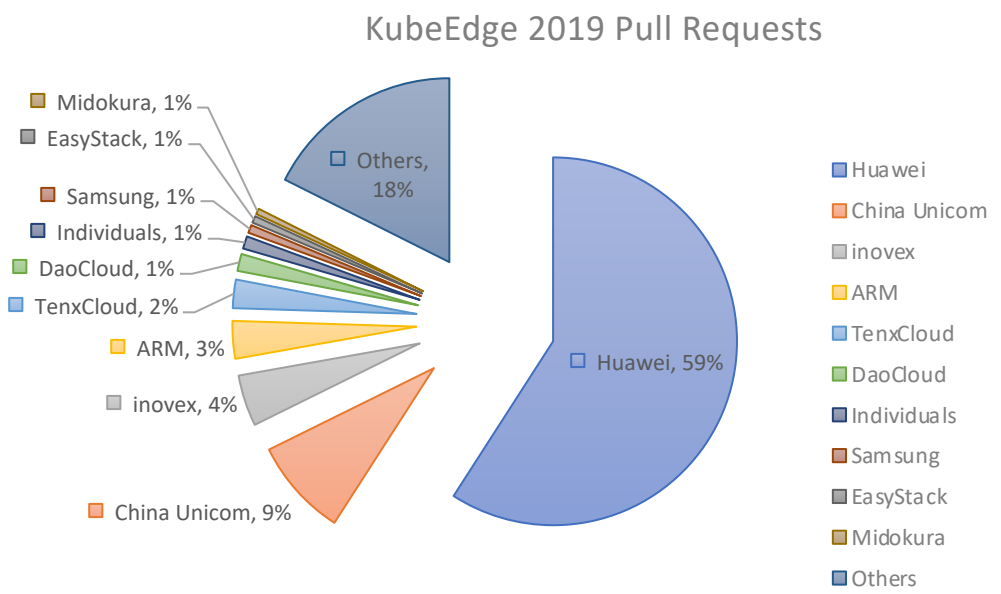


- **Cloud/Edge Nodes Unified management**
- **Device and Edge Application Unified Management**
- **Simplified development**: Developers can write applications, containerize them, and run them anywhere - either at the Edge or in the Cloud - whichever is more appropriate.
- **Cloud-Native, Kubernetes-native support**: Users can orchestrate apps, manage devices and monitor app and device status on Edge nodes just like a traditional Kubernetes cluster in the Cloud. Locations of edge nodes are transparent to customers. Extend K8s To Edge.

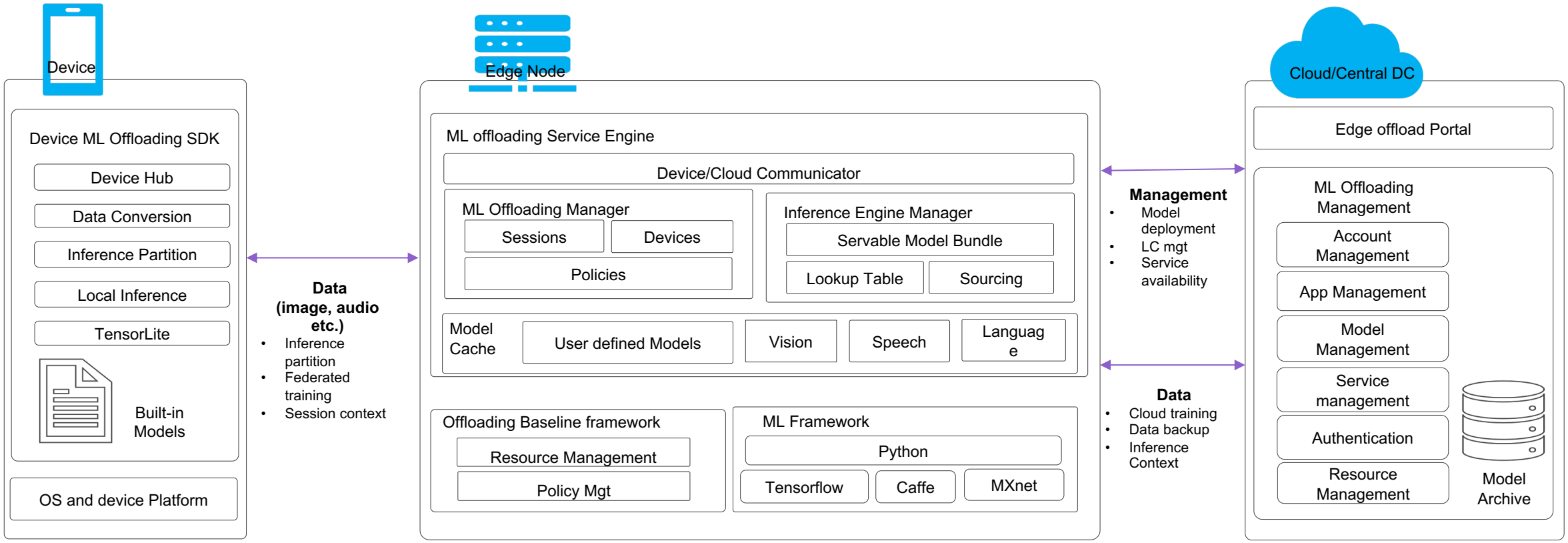
KubeEdge Status

- 2019 Mar entered CNCF Sandbox
- 5 minor(feature) releases, v1.2 released 2020.2
- 2.1k+ Star, 500+ Fork on Github, 1k+ wechat members
- 250+ Contributors (90 submitted code);
  > 10 Approvers (**1 Infoblox, 1 HP, 1 Microsoft**)
  > 14 Reviewers **(1 China Unicom, 1 ARM, 1 Infoblox, 1 Inovex, 1 HP, 1 Microsoft)**
- Over 40% PR made by non-huawei contributors in 2019





Edge ML Offloading Architecture Diagram



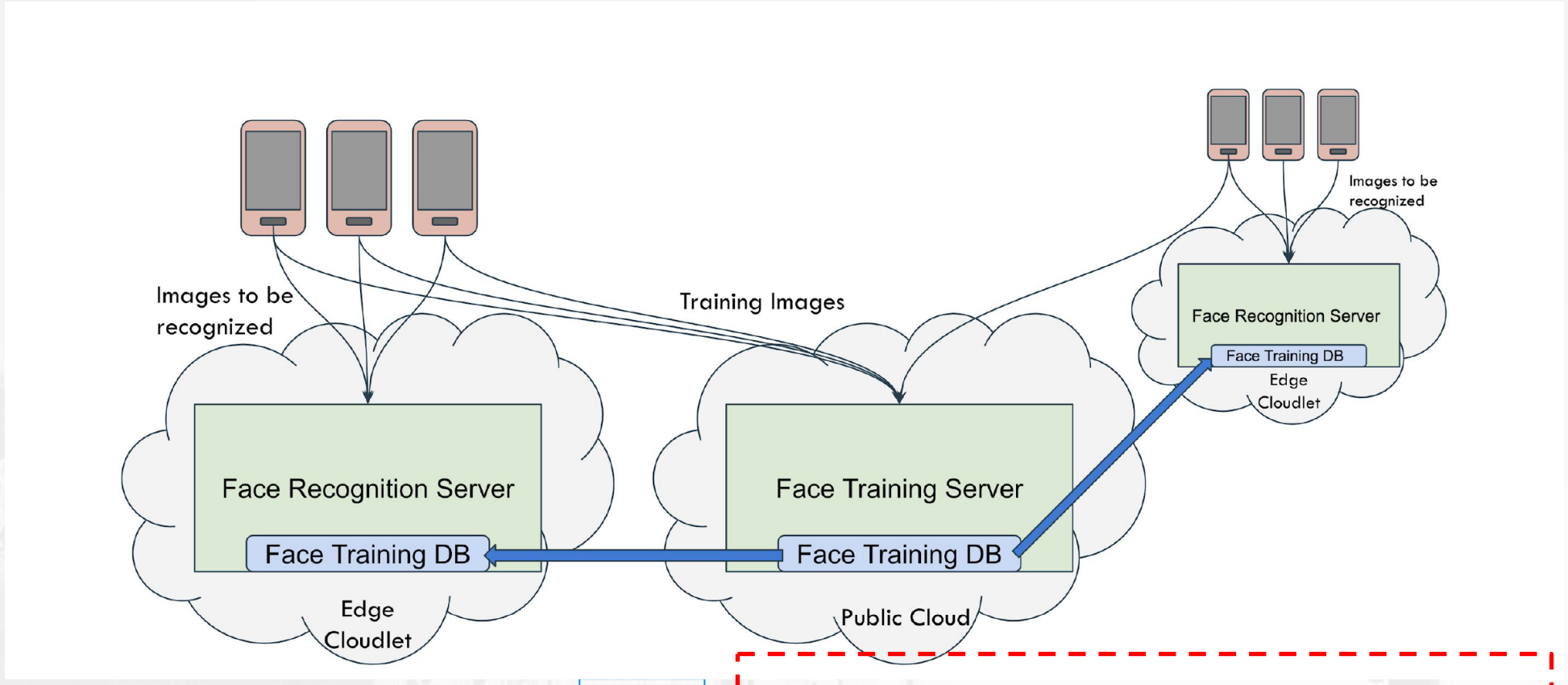
Device/Edge/Cloud Collaborated
ML Offloading architecture. It consists of Device(SDK), Edge offloading Service and Central management service.

Edge platform agnostic
The Edge offloading service can be deployed in any container based platforms. There is no dependency on special services from platforms. It can leverage partner services for advanced features. However those are not offloading core functions.

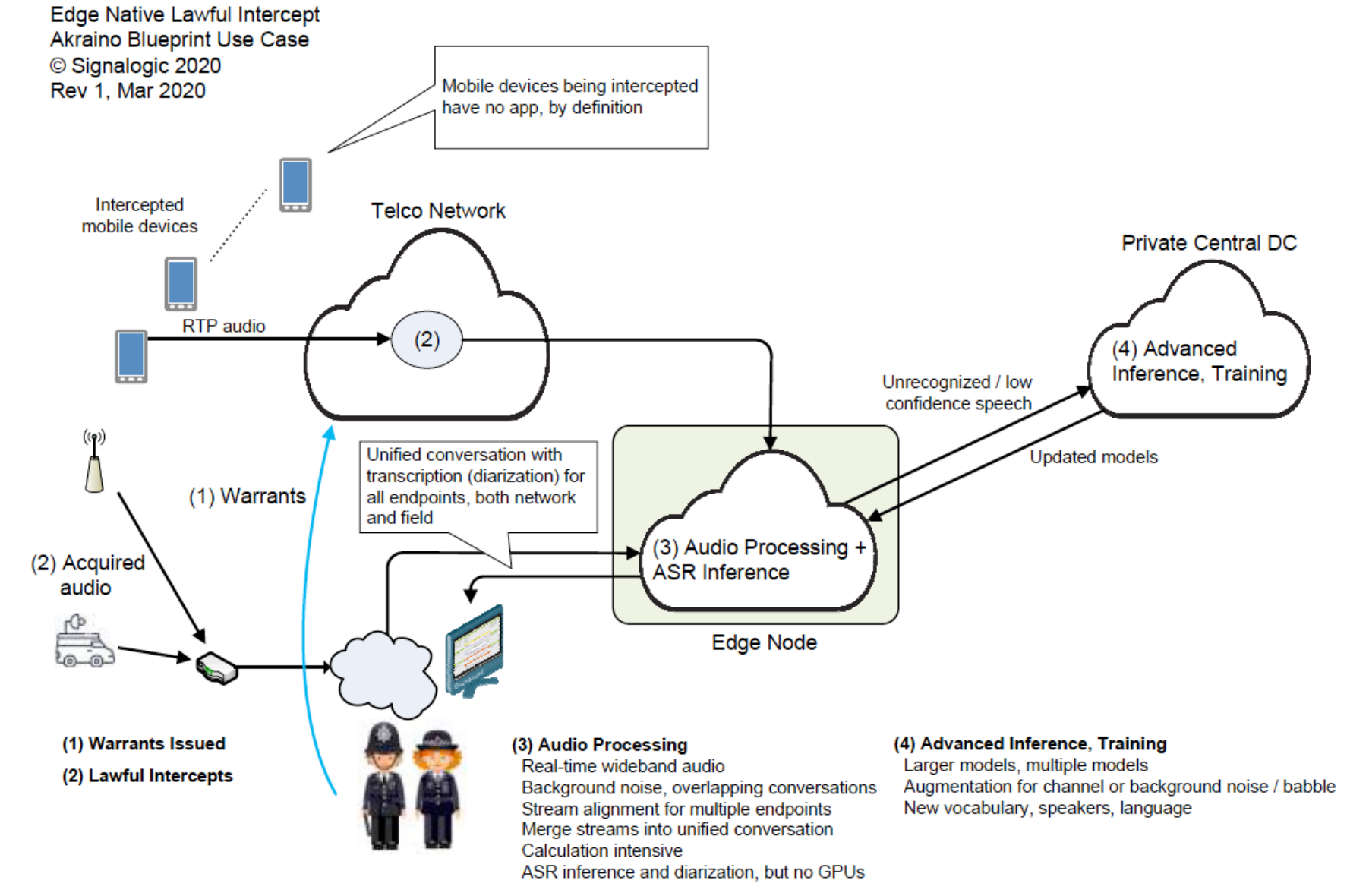
Offloading as a Service (OaaS)
OaaS can be offered to any applications which needs edge capabilities. It serves as a SaaS on top of Edge PaaS layer.

Open and Expandable Architecture
This diagram focuses on ML inference edge offloading use case. The overall architecture can be expanded to support various other ML offloading services including federated training.

Use Case 1: Mobile app AI model inference offloading workflow



Use Case 2: Edge Offloading Speech Recognition in Operation Field



Appendix: Assessment Criteria

Criteria	This BP	Criteria	This BP
Each initial blueprint is encouraged to take on at least two committers from different companies	Futurewei, ARM, CMCC	Project contact name, company, and email are defined and documents	Yin.ding@futurewei.com
Complete all templates outlined in this documents	Detailed in this slide	Description of the project goal and its purpose are defined	A neutral Kubernetes Edge platform
A lab with exact configuration required by the blueprint to connect with Akraino CI and demonstrate CD. User should demonstrate either an existing lab or the funding and commitment to build the needed configuration.	Validation lab hosted by Futurewei	Scope and project plan are well defined	Targeting R4
		Resource committed and available	Yes
Blueprint is aligned with the Akraino Edge Stack Charter	Yes	Contributors identified	Futurewei, ARM, CMCC
Blueprint is code that will be developed and used with Akraino repository should use only open source software components either from upstream or Akraino projects.	Yes	Initial list of committers identified (elected/proposed by initial contributors)	Futurewei, ARM
For new blueprints submission, the submitter should review existing blueprints and ensure it is not a duplicate blueprint and explain how the submission differs. The functional fit of an existing blueprint for a use case does not prevent an additional blueprint	Yes, KubeEdge focused blueprint is new to the community	Meets Akraino TSC policies	Yes. The project will operate in an open, collaborative and ethical manner
		Proposal has been socialized with potentially interested or affected projects and/or parties	Yes
Name of the project is appropriate(no trademark issues etc.); Proposed repository name is all lower-case without any special characters.	KubeEdge Edge Services	Cross Project Dependencies	KubeEdge framework

AKRAINO EDGE STACK

Legal Notices

The Linux Foundation, The Linux Foundation logos, and other marks that may be used herein are owned by The Linux Foundation or its affiliated entities, and are subject to The Linux Foundation's Trademark Usage Policy at https://www.linuxfoundation.org/trademark-usage, as may be modified from time to time.

Linux is a registered trademark of Linus Torvalds. Please see the Linux Mark Institute's trademark usage page at https://lmi.linuxfoundation.org for details regarding use of this trademark.

Some marks that may be used herein are owned by projects operating as separately incorporated entities managed by The Linux Foundation, and have their own trademarks, policies and usage guidelines.

TWITTER, TWEET, RETWEET and the Twitter logo are trademarks of Twitter, Inc. or its affiliates.

Facebook and the "f" logo are trademarks of Facebook or its affiliates.

LinkedIn, the LinkedIn logo, the IN logo and InMail are registered trademarks or trademarks of LinkedIn Corporation and its affiliates in the United States and/or other countries.

YouTube and the YouTube icon are trademarks of YouTube or its affiliates.

All other trademarks are the property of their respective owners. Use of such marks herein does not represent affiliation with or authorization, sponsorship or approval by such owners unless otherwise expressly specified.

The Linux Foundation is subject to other policies, including without limitation its Privacy Policy at https://www.linuxfoundation.org/privacy and its Antitrust Policy at https://www.linuxfoundation.org/antitrust-policy. each as may be modified from time to time. More information about The Linux Foundation's policies is available at https://www.linuxfoundation.org.

Please email legal@linuxfoundation.org with any questions about The Linux Foundation's policies or the notices set forth on this slide.

AKRAINO EDGE STACK