

# KubeEdge BP Architecture Documents

## Blueprint overview/Introduction

KubeEdge Edge Service family focuses on a device/edge/cloud collaboration framework around KubeEdge. The verticals of focus have been IoT and MEC etc.

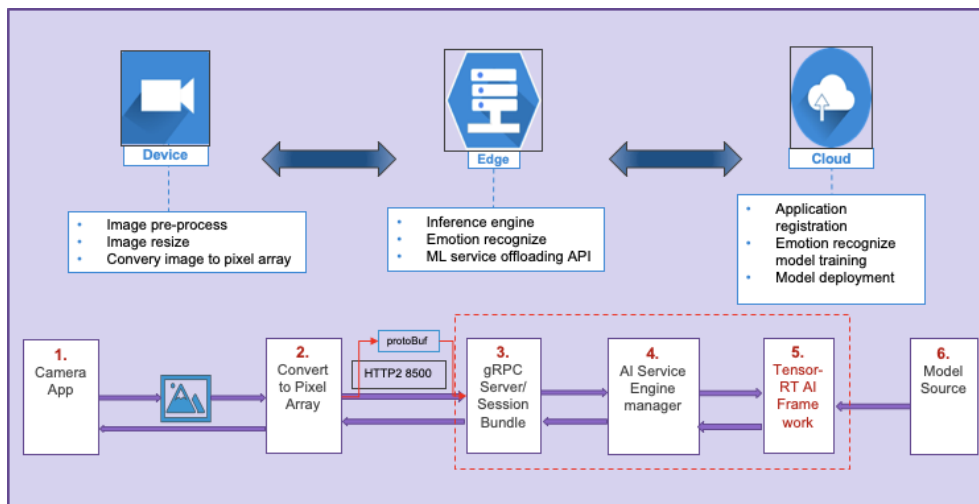
- The key component KubeEdge is a unique design from scratch of edge nodes and edge cloud, with all source code developed in the upstream CNCF KubeEdge Project.
- Type I of KubeEdge Edge Service family focuses an edge stack of the use case ML Inference Offloading on edge servers.
- This blueprint family will leverage various infrastructures. Arm servers will be supported. This blueprint family is infrastructure neutral.

## Use Case

### Type 1 project use case:

Type 1 use case is Emotion recognition scenario. This use case shows cloud, edge and device collaboration.

- Cloud - focus on Emotion recognition model training.
- Edge - provides a ML offloading service for inference.
- Device - preprocess the image and use edge ML offloading service to provide an emotion recognition service.



### Future Use cases:

Use case 1: Smart road and autonomous driving scenario: KubeEdge facilitates smart road, deploys MEC sites along the highway system. The smart road system is designed for the emerging autonomous driving industry. Those self-driving cars can leverage the smart road facility to process perception or navigation workload.

Use case 2: KubeEdge can be helpful in logistics, especially useful in tracking product and equipment conditions in cold chain logistics. To maintain low temperature as well as control other conditions, cold chain cargo needs to process inputs from sensors in it, and take actions in the real-time. KubeEdge enables cargo fleets on-board systems to control and communicate the situation of cold-chain conditions.

Use case 3: Smart building scenario, KubeEdge can be perfectly used in helping the smart building local data processing. Lately, IoT sensors have been used in commercial buildings, however due to the data security requirement, most of IoT data need to be processed on-perm. Hundreds of data processing systems support different commercial tenants.

## Where on the Edge

Industry Sector: Cloud, Enterprise, Telco

Edge computing leverages edge locations to distribute application loads among device/edge/cloud. A service layer is required to bridge infrastructure platform and applications. e.g. load distribution coordination, hardware platform agnostic, etc. KubeEdge extends native containerized application orchestration capabilities to hosts at Edge. Along with other vertical domain support such as device twin at edge, KubeEdge edge service stack is geared to offer feature rich support to applications while remain platform neutral.

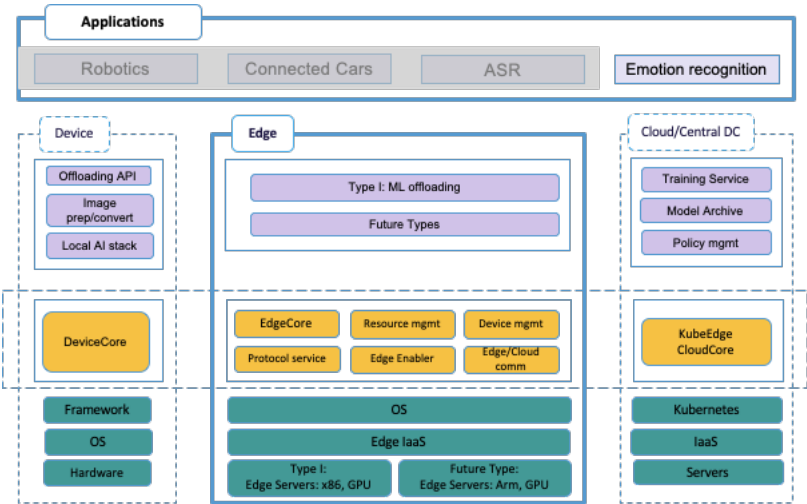
## Overall Architecture

The overall system contains cloud, edge and device three parts.

Cloud - Kubernetes platform with KubeEdge cloudcore deployed. TensorFlow training.

Edge - Linux OS, KubeEdge EdgeCore, TensorFlow inference, ML offloading service.

Device - Android smart camera or smart phone, emotion recognition App.



## Platform Architecture

**Cloud** - AWS virtual machine, 8 cores, 32 GB memory, 80GB EBS disk

**Edge** -

On Prem Edge: Intel Xeon E5-2620, 16 cores, 128 GB memory, 3TB disks

Cloud Edge: AWS virtual machine, 4 cores, 16 GB memory, 80GB EBS disk

**Device** - Android emulator, Pixel 2 API 29, Android 10.0

## Software Platform Architecture

Software	release/version number
Kubernetes	1.18
KubeEdge	1.4
TensorFlow	2.0
Android	10.0
Ubuntu (cloud & edge nodes)	20.04

## APIs

APIs with reference to Architecture and Modules

High Level definition of APIs are stated here, assuming Full definition APIs are in the [KubeEdge BP API Documents](#)

## Hardware and Software Management

### Hardware Management

Currently for this blueprint AWS Virtual Machines are being used for development, testing and CI/CD hence there is no specific hardware management to be done.

## Software Management

<https://github.com/kubeedge/kubeedge>

<https://github.com/futurewei-cloud/kubeedge-android-ai>

## Licensing

- GNU/common license