

Video Security Monitoring R6 Architecture Document

- [Blueprint overview/Introduction](#)
 - [Use Case](#)
 - [Where on the Edge](#)
- [Overall Architecture](#)
- [Platform Architecture](#)
- [Software Platform Architecture](#)
 - [OTE Components](#)
 - [WebFrontend](#)
 - [OpenAPI](#)
 - [Scheduler](#)
 - [cluster-controller](#)
 - [k8s-cluster-shim](#)
 - [k3s-cluster-shim](#)
 - [NodeAgent](#)
 - [NodesServer](#)
 - [DataQueryServer](#)
 - [VMController](#)
 - [EdgeHub](#)
 - [EdgeController](#)
 - [Software components](#)
- [APIs](#)
- [Hardware and Software Management](#)
- [Licensing](#)

Blueprint overview/Introduction

The AI Edge is an Akraio approved blueprint family and part of Akraio Edge Stack, which intends to provide an open source MEC platform combined with AI capacities at the Edge, and could be used for safety, security, and surveillance. The MEC platform, which named ote-stack, targets on shielding the heterogeneous characteristics through underlying hardware virtualization and providing an unified access for cloud edge, mobile edge and private edge. In addition, the AI Edge utilizes the cluster management and intelligent scheduling of multi-tier clusters to enable low-latency, high-reliability and cost-optimal computing support for running AI applications at the edge. At the same time, it makes device-edge-cloud collaborative computing possible.

This blueprint mainly focus on building an edge manage platform with [Integrated Edge Cloud\(IEC\)](#) as infrastructure platform to implement the video security monitoring for School/Education on Arm servers in edge.

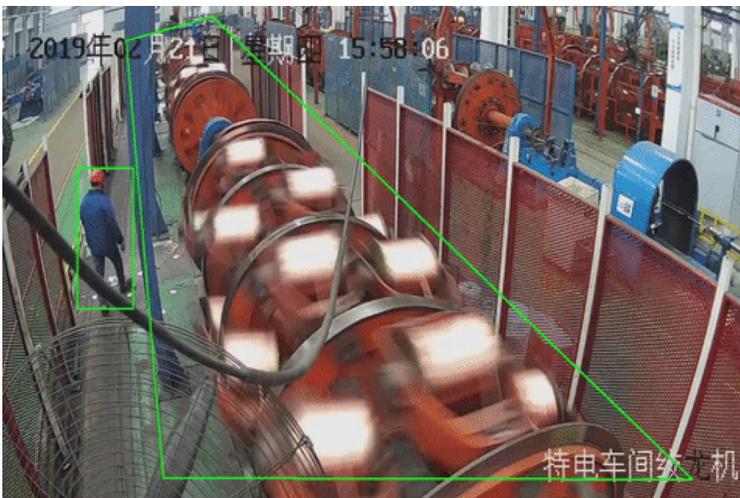
Use Case

Real time AI applications, which should be processed locally and securely, can be deployed on top of the OTE. Below are some brief descriptions of the apps.

<use case 1: Security monitoring>

Conduct smoke detection on densely populated places such as industrial parks and community properties to quickly detect whether there is a fire, reduce the damage caused by fire, and improve the safety of the park.

Real-time illegal/dangerous area intrusion detection to monitor whether there are any behaviors such as using mobile phones, making phone calls, falling, intruding into dangerous areas, and illegal parking of vehicles.

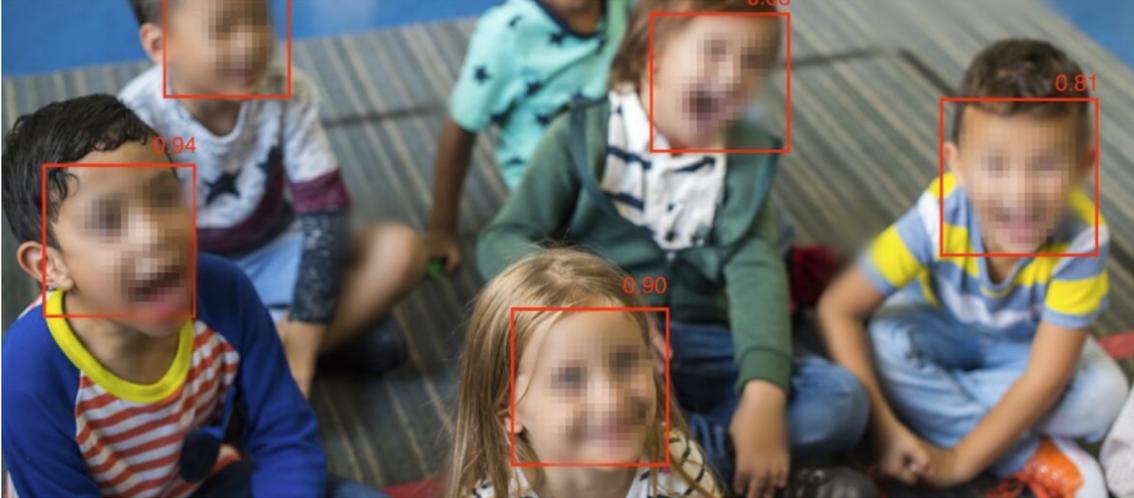


<use case 2: Classroom concentration analysis>

Conduct a full evaluation of the overall class and the concentration of individual students, help teachers and school authorities to fully understand the teaching situation, according to the concentration data of each course, conduct targeted class knowledge test and strengthen.

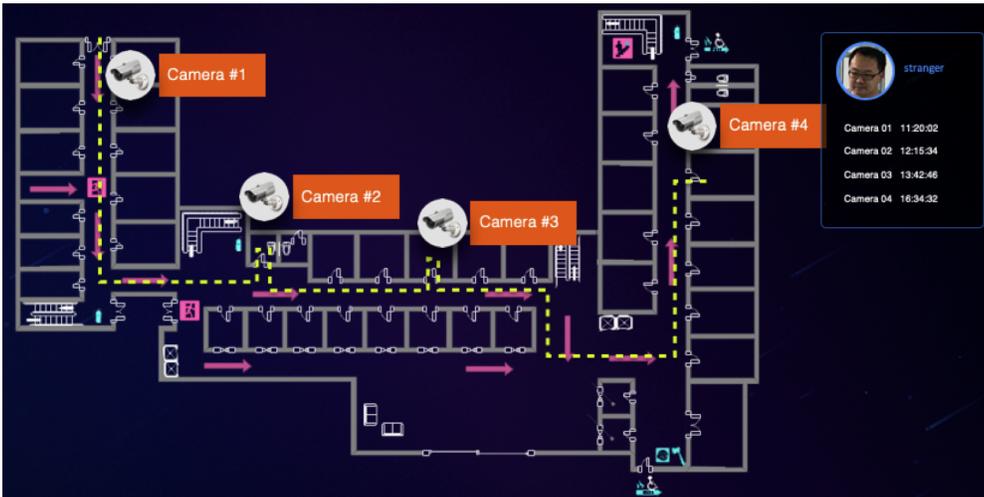
This application can accurately analyze the students' concentration in class in multiple dimensions, so as to comprehensively evaluate the teaching effect.

Based on the image information collected by the camera, this application can identify elements such as student behavior, expressions, micro-actions, objects, etc. in the classroom. It needs to arrange a camera in front of the classroom to collect video. And then it can automatically analyze the concentration of students and feedback various behavior statistics to the school management system in real time.



<use case 3: Backtracking of suspicious persons>

The video surveillance application can target suspicious persons or objects entering the school through the support of the OTE platform. It can automatically record and generate the trajectory of suspicious target, follow the target rotation and trigger the alarm operation.



Where on the Edge

Business Drivers

The AI Edge will provide a cluster management for different logical MEC edge clusters. Through the standard api interface, the third clusters can join the management of AI Edge easily, so as to schedule deployment of an AI application to a specific edge node with the unified access. The benefits are: Lower cost on manage multiple edge clusters and more computing power of edge devices can be utilized ; Less load and latencies on network and more safely since the application is running locally; Edge cluster autonomy.

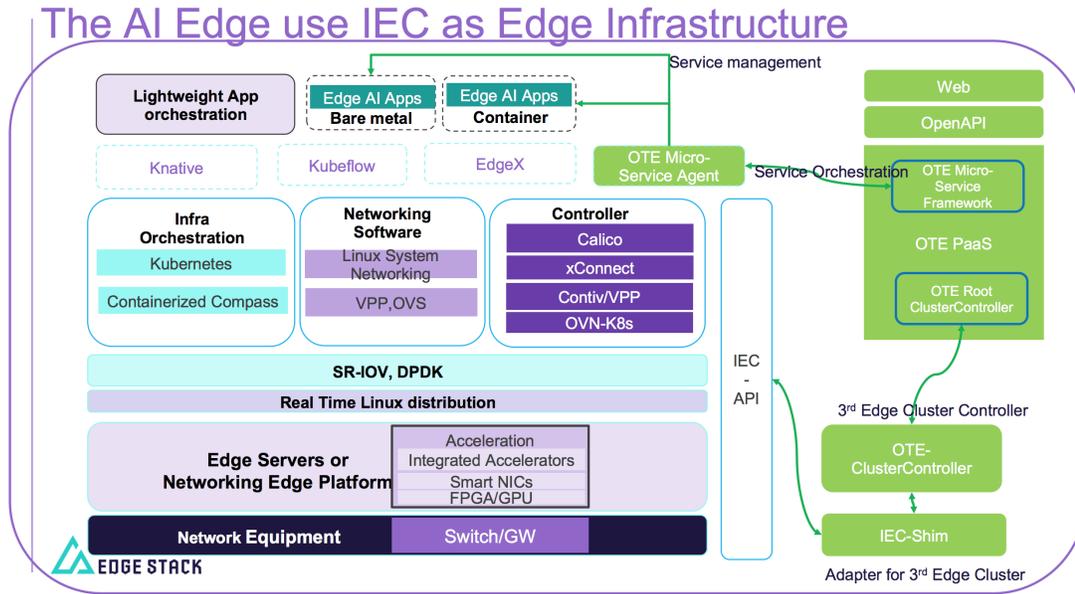
Overall Architecture

The AI Edge blueprint architecture consists of a cluster control manager with web platform at the cloud and multiple edge clusters. The number of clusters can be theoretically unlimited which can effectively solve the management and scheduling problems of large-scale mobile edge clusters in 5G era. For development environment we have tested with one IEC clusters with 3 nodes.

The cluster control manager, which consists of ote-web, openapi and lightweight cluster-controller, manages orchestration and life cycle of applications on the edge cluster and the hierarchical structure of clusters. While the ote-web and openapi provides access to the AI Edge, the cluster-controller provides core capabilities support for network connection, metadata synchronization and message transmit between cloud and edge and establishes the routing path for all edges. The edge, can be a kubernetes cluster, a k3s cluster or other private cluster, will be deployed a cluster-controller and a cluster-shim as to receive and process messages from upper cluster. Due to the autonomy of edge cluster, the network/workspace infrastructure and data volumes are managed by itself. Therefore, the deployed AI applications can still run normally when disconnection from the cloud occurred.

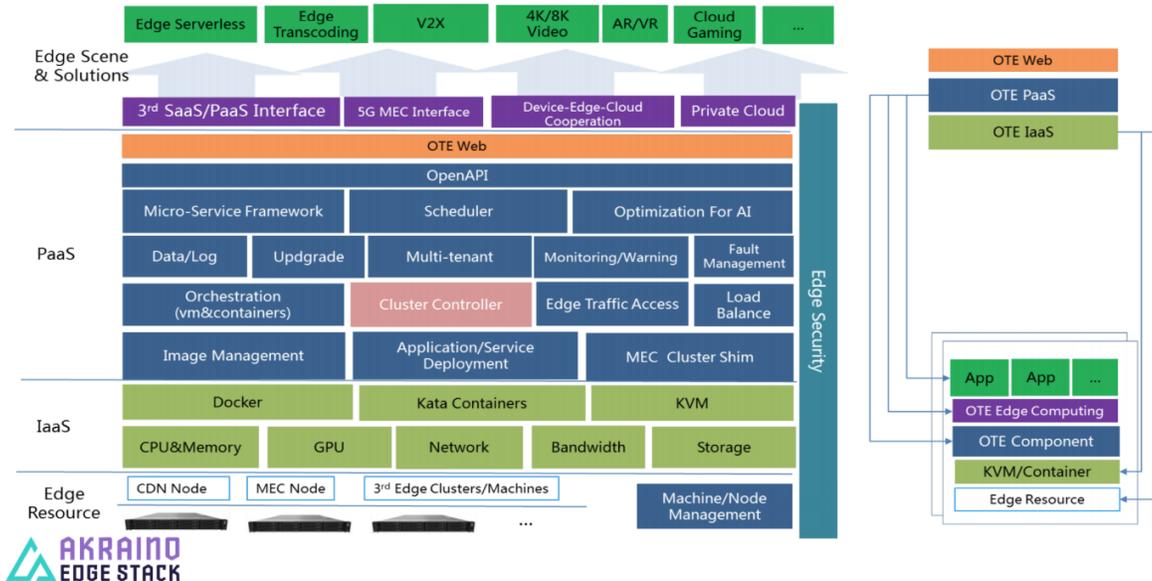
Many cloud native monitoring applications are used to collect container/node resource usage and running log, like prometheus, elasticsearch.

The below image shows the overall architecture for using IEC as edge infrastructure in AI Edge.



Platform Architecture

The detailed platform architecture of AI Edge blueprint is shown in the below diagram.



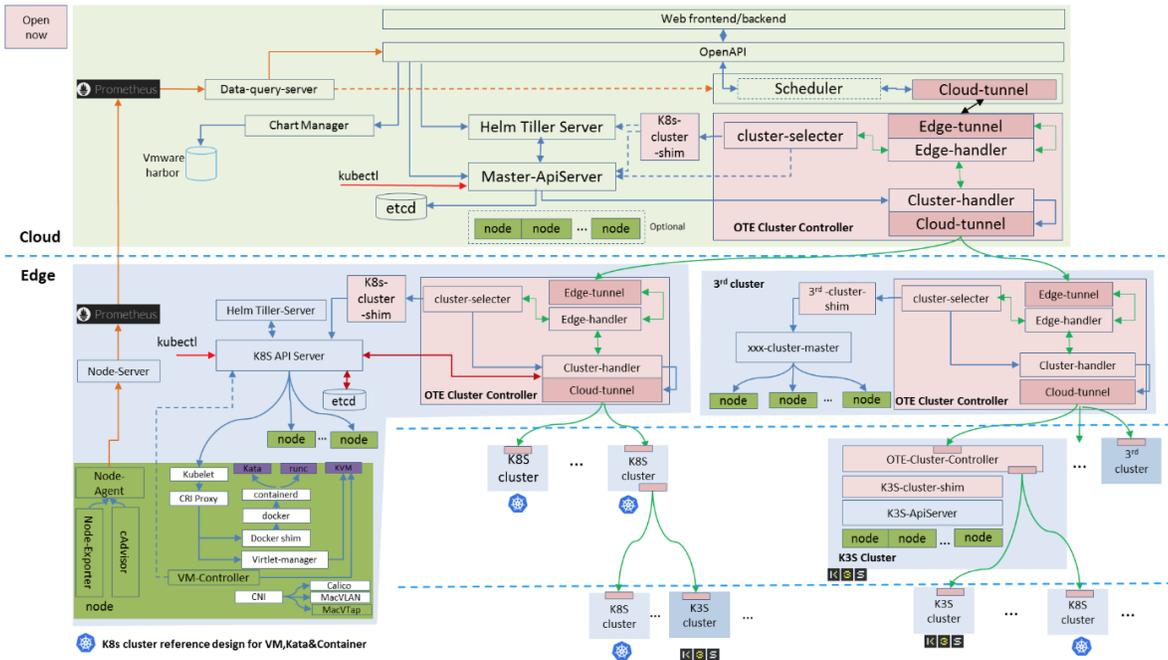
In the current release the components released are

- cluster-controller and controller-manager
- k8s-cluster-shim

Other components, such as openapi, ote-web, are currently released as docker images and will be open source in the future.

Software Platform Architecture

The below image shows the software architecture for this release.



OTE-Stack features a pluggable architecture, making it much easier to build on.

- The global scheduler is fully compatible with kubernetes. Users can operate directly using kubectl;
- Using websocket for the edge-cloud communication;
- In addition to the cluster name, the cluster tag can be added customically. Cluster tag matching through intelligent cluster-selector to achieve accurate routing of messages;
- Through k8s-cluster-shim to achieve the management of kubernetes cluster, shielding the native implementation within the kubernetes cluster;
- According to the interface of OTE-Stack, the cluster shim of the third party cluster can be realized to access and schedule the third party cluster. The internal implementation of the third party cluster is shielded;
- Each layer can be used as a control entry to control all sub-clusters below this layer. Users can also use kubectl or API to implement custom cluster management and scheduling.

On the edge, OTE-Stack provides the ability to support node autonomy within the cluster.

blocked URL

- Use EdgeHub to take over the kubelet, kube-proxy or k3s agent components of the node.
- EdgeHub provides load balancing capabilities for node requests.
- During node autonomy, you can use EdgeController instead of node to complete node status reporting and updating.
- EdgeHub and EdgeController have both the autonomy capabilities of k8s and k3s clusters.

OTE Components

WebFrontend

WebFrontend is a web portal which provides a graphical user interface to help users manage deployment in multiple clusters.

OpenAPI

OpenAPI is a HTTP server that accepts REST-like API requests and preprocess these requests in advance to determine whether they match the required criteria, then pass these request to the backend components for further processing.

Scheduler

Scheduler aims to schedule pod and resource across different clusters that belongs to user. This function will be available in the future and is not supported right now.

cluster-controller

ClusterController is a message connector which joins the multiple clusters. It consists of the following components:

- **EdgeTunnel**
Northbound interface of Controller. By establishing websocket connection with CloudTunnel of upper cluster, messages between clusters can be transmitted smoothly.
- **EdgeHandler**
It can add tags to cluster, receive and process messages from upper cluster, transmit messages to ClusterHandler, receive messages from ClusterHandler and realize cluster disaster recovery automatically.
 - Users can configure their own cluster name or add cluster tags to achieve complex cluster management.
 - Used for receiving messages sent by EdgeTunnel and forwarding them to Cluster Selector for routing or direct transmission to ClusterHandler after processing.
 - Receive messages sent back by ClusterHandler or shim (such as changes in sub-cluster, status, etc.) and pass them to the upper cluster through EdgeTunnel after processing.
 - Once the connection between the current cluster and the parent cluster is established, the sibling cluster of the parent, the parent cluster of the parent and the sibling cluster of itself will be automatically acquired as the alternative cluster. When Disconnected the alternative one is connected automatically. The connection to the central can be quickly restored. Meanwhile, it regularly checks whether the previous parent cluster is restored, and once restored, it restores the previous connection topology.
- **ClusterSelector**
It is used to complete the routing of cluster messages, and it accepts the processing of two kinds of cluster routing rules.
 - If it is a real list of cluster names, it matches the names according to the cluster routing rules and looks for the next hop until it reaches the specified cluster accurately.
 - If it's a cluster's fuzzy rules, such as * tagA*, it matches all tagA-containing clusters in the tag and maps them to the real names of the clusters. Then it uses the above rules to pass down until it reaches the specified cluster accurately.
- **ClusterHandler**
It's core components of cluster management.
 - Store the names and labels of all subclusters.
 - Establish routing rules that store the next hop cluster name to any sub-cluster to support accurate delivery of messages.
 - Notify the upper cluster in time when the sub-cluster changes (such as disconnection, status updates, etc.)
- **CloudTunnel**
Southbound interface of Controller. By establishing websocket connection with EdgeTunnel of sub-cluster, messages between clusters can be transmitted smoothly.

k8s-cluster-shim

It is an adapter of kubernetes cluster, which receives and parses cluster messages forwarded by OTE Cluster Controller, sends them to kubernetes cluster for corresponding processing, and returns the results and status to OTE Cluster Controller in time.

k3s-cluster-shim

It is an adapter of k3s cluster, which receives and parses cluster messages forwarded by OTE Cluster Controller, sends them to k3s cluster for corresponding processing, and returns the results and status to OTE Cluster Controller in time.

NodeAgent

It is deployed on edge nodes to retrieve data from cAdvisor and Node-Exporter which will be uploaded to NodesServer in edge clusters.

NodesServer

In the edge cluster, it is used to aggregate data of each node and provide it to Prometheus (Prometheus can also directly collect data of the node, but requires the node to open the corresponding ports)

DataQueryServer

Exposing Prometheus data as APIs to OpenAPI and Scheduler

VMController

Operations for a single VM, such as start, stop, etc.

EdgeHub

Proxy between master and node in the edge cluster.

EdgeController

It is used to take over the state management of the node when the node is disconnected from the master.

Software components

S.No.	Component	Category	Version	License
1	IEC	Edge Infrastructure	Integrated Edge Cloud (IEC) Blueprint Family -R2	Apache 2.0
2	Etdc	Database	v3.3.18	Apache 2.0
3	Helm	Application Package Manager	v2.13.1	Apache 2.0
4	Mysql	Database	>5.5	GNU General Public License Version 2
5	Node-exporter	Node Metrics	v0.18.1	Apache 2.0
6	Prometheus	Monitoring	v2.15.0	Apache 2.0
7	Fluent-bit	Log Processor and Forwarder	1.3	Apache 2.0
8	Elasticsearch	Log Search Engine	6.5.0	Apache 2.0
9	Alertmanager	Monitoring	v0.20.0	Apache 2.0
10	OTE WebFrontend	OTE component-Web Platform	0.1.14	Apache 2.0
11	OTE OpenAPI	OTE component-API Gateway	0.1.5.8	Apache 2.0
12	OTE Cluster Controller	OTE component	2.0	Apache 2.0
13	K8s-cluster-shim	OTE component-Performer of Edge Cluster	2.0	Apache 2.0
14	NodeAgent	OTE component-Metrics Collector Agent	2.0	Apache 2.0
15	NodeServer	OTE component-Metrics Aggregation Service	2.0	Apache 2.0
16	DataQueryServer	OTE component-Metrics Search Service	2.0	Apache 2.0
17	EdgeHub	OTE component	1.0	Apache 2.0
18	EdgeController	OTE component	1.0	Apache 2.0

APIs

[Video Security Monitoring R5 API Document](#)

Hardware and Software Management

Software Management: [Gerrit Repo](#)

Licensing

- Apache 2.0 license