

Architecture Document

- [Use Cases](#)
- [Overall Architecture](#)
- [Optimization](#)

Main Progresses for Release 2

- [Build Of Materials \(BOM\)](#) / [Hardware requirements](#)

[Integrated Edge Cloud\(IEC\)](#) is an Akraino approved blueprint family and part of Akraino Edge Stack, which intends to develop a fully integrated edge infrastructure solution, and the project is completely focused towards Edge Computing. This open source software stack provides critical infrastructure to enable high performance, reduce latency, improve availability, lower operational overhead, provide scalability, address security needs, and improve fault management. The IEC project will address multiple edge use cases and industry, not just Telco Industry. IEC intends to develop solution and support of carrier, provider, and the IoT networks.

IEC Type 4 is focused on AR VR applications running on edge.

In general, the architecture consists of three layers: Iaas(IEC), PaaS(Tars), SaaS(AR/VR Application).

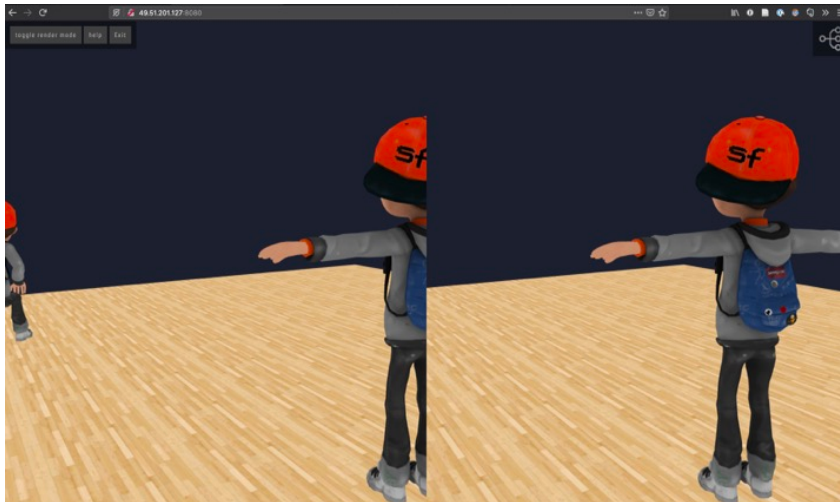
Use Cases

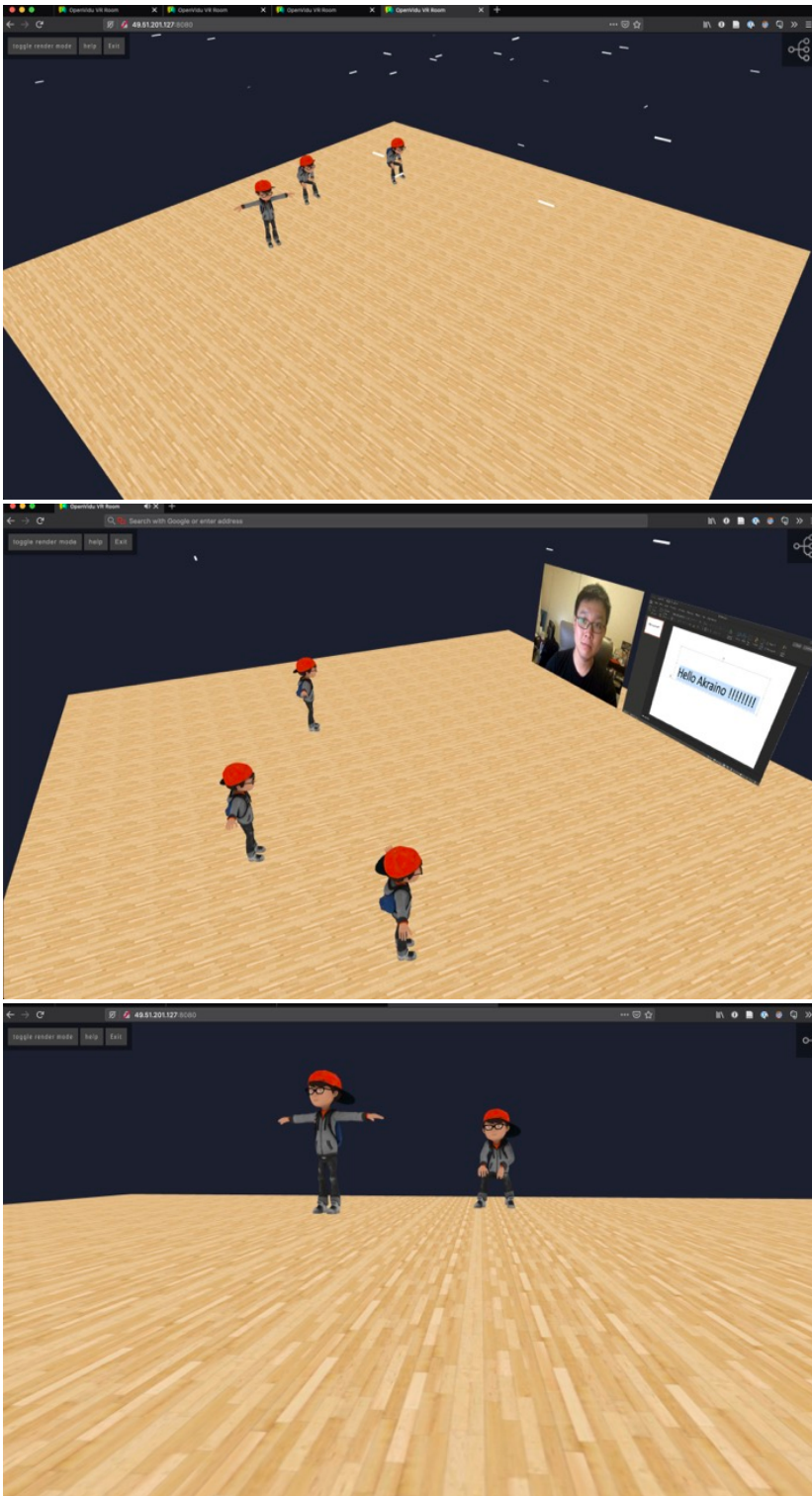
In general, there are multiple use cases for AR VR itemized below. For Release 2, we focus on building the infrastructure and virtual classroom application (Highlighted in dark purple color).

Virtual Classroom is a basic app that allows you to live a virtual reality experience simulating a classroom with teachers and students.

UseCases	Value Proposition
Operation Guidance	Predict the next step for the operations(like assembling Lego blocks, cooking sandwiches, etc) and help people to achieve a goal.
Virtual Classroom	Simulating a virtual classroom, which improves online education experiences for the teachers and students.
Sports Live	Augment and simulate the sports live, which gives the audiences an amazing immersive watching experience.
Gaming	Augment and simulate the game scenario, let players enjoy an immersive game world.

After successfully deploy the virtual classroom application, the following virtual classroom is waiting for you. Enjoy!



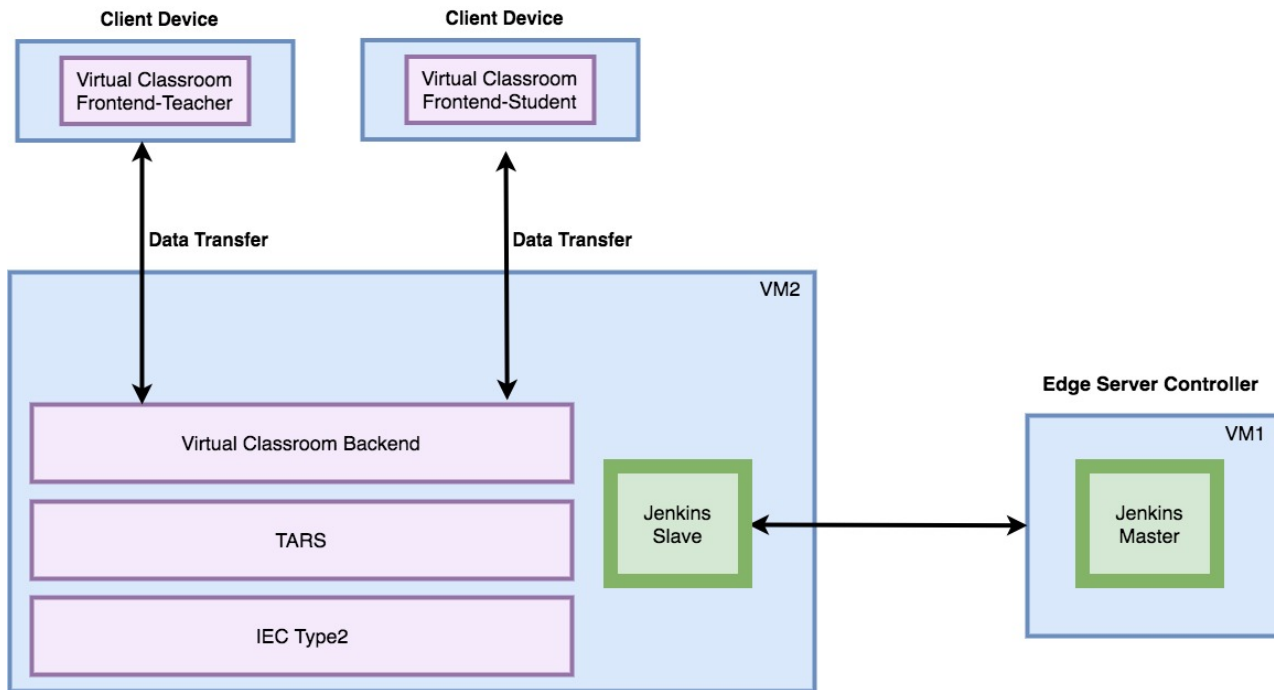


Overall Architecture

The whole architecture, shown below, consists of two parts: the front end and the backend.

- For the front end, the minimal requirements are two clients, one for the teacher and the other one for the student. The client device could be a cellphone, tablets, wearable devices, personal computers, etc. The client collects information from the real world and transfers the data to the backend for calculation. Beyond data transfer and calculation, render is another function running on the front end client-side.
- For the backend, we deploy the backend in two virtual machines in the Cloud.
 1. To make the VR backend work well, we deploy IEC in the IaaS Layer, Tars framework in PaaS Layer, Virtual Classroom Backend in SaaS Layer.

2. To make CI/CD available, we deploy Jenkins Master in one Virtual Machine. The Jenkins master issues command to trigger the script run on the dedicated VM.



Tars is a high-performance microservice framework based on name service and Tars protocol, also integrated administration platform. For more information about Tars, refer to the following documents:

[TARS_Introduction.pdf](#)

IEC (Integrated Edge Cloud) is a platform that will enable new functionalities and business models on the network edge. The benefits of running applications on the network edge are - Better latencies for end users - Less load on network since more data can be processed locally - Fully utilize the computation power of the edge devices. Here IEC Type2 is used as the main working base for IEC Type4 deployment orchestration. For more information about IEC, refer to the following documents:

[IEC Release Type2 Documentation](#)

Optimization

To address the requirements of time-critical Application, some optimizations are itemized below:

- High-Performance RPC: Tars Framework supports a high-performance RPC Protocol, called Tars Protocol. To improve the performance, the Tars protocol re-design a brand new RPC protocol and make the transfer between "request" and "Binary stream" more efficiently. In terms of the detail of Tars protocol, refer to the Tars.h file in the <https://github.com/TarsCloud/Tars>.
- Lightweight framework
 1. To make Tars can be well deployed on the edge computing platform, we make the framework pluggable. The customer plugin the pluggable components when we really need it. Unnecessary components can be avoided.
 2. Rewrite some functions(like scale-out, monitor and so on) to reduce CPU consumption.
 3. Orchestrate no-urgent functions(like monitor data calculation and so on) from edge to DC(or higher level edge). Reduce resource consumption.
- IEC Optimization: IEC uses Calico as the main container networking solution which uses a pure IP networking fabric to deliver high-performance Kubernetes networking, and its policy engine enforces developer intent for high-level network policy management. Together, Calico and Kubernetes provide a secure, cloud-native platform that can scale your infrastructure to support hundreds of thousands of users. Calico CNI also supports multi-cni plugin. In this way, users can customize their own edge systems.

Main Progresses for Release 2

Release 2 is the first release for IEC Type 4. So everything is new.

Build Of Materials (BOM) / Hardware requirements

For R2, we deploy IEC in Tencent Cloud for Release. The detailed hardware is itemized below:

CPU+Memory	Drive	Deployment
4Core * 16G	50G	Jenkins Master
4Core * 16G	50G	Jenkins Slave + Virtual Classroom Backend + Tars + IEC