

Release 4 Blog of IEC Type 4 AR/VR Blueprint

The IEC Type 4 focuses on AR VR applications running on the edge. In general, the architecture consists of three layers:

- We use IEC as IaaS Layer.
- We deploy TARS Framework as PaaS Layer to governance backend services.
- And for the SaaS layer, AR/VR Application backend is needed.

There are multiple use cases for AR VR itemized below. For Release 4, we focus on building the infrastructure and virtual classroom application, which I highlighted here.

Use Cases	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.

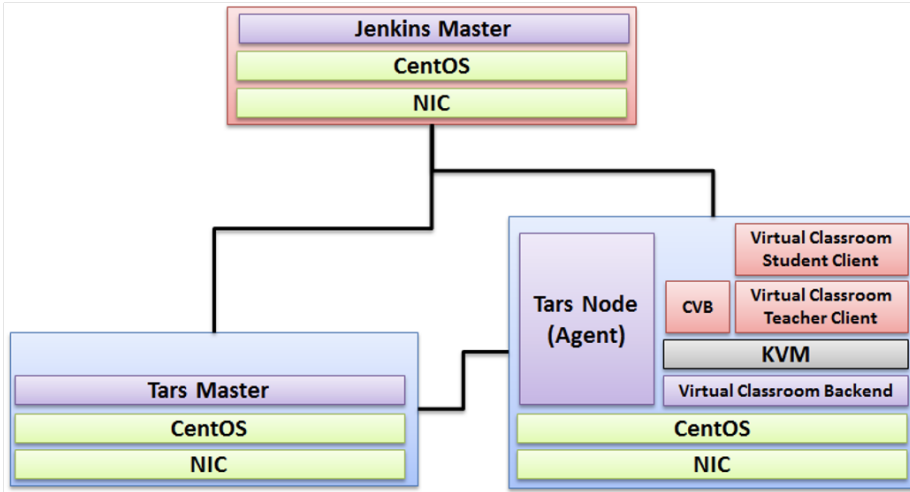
Let's have a quick stare at what is Virtual Classroom. Virtual Classroom is a basic app that allows you to live a virtual reality experience simulating a classroom with teachers and students.



Generally, it has two modes, the Teacher mode, and the Student mode.

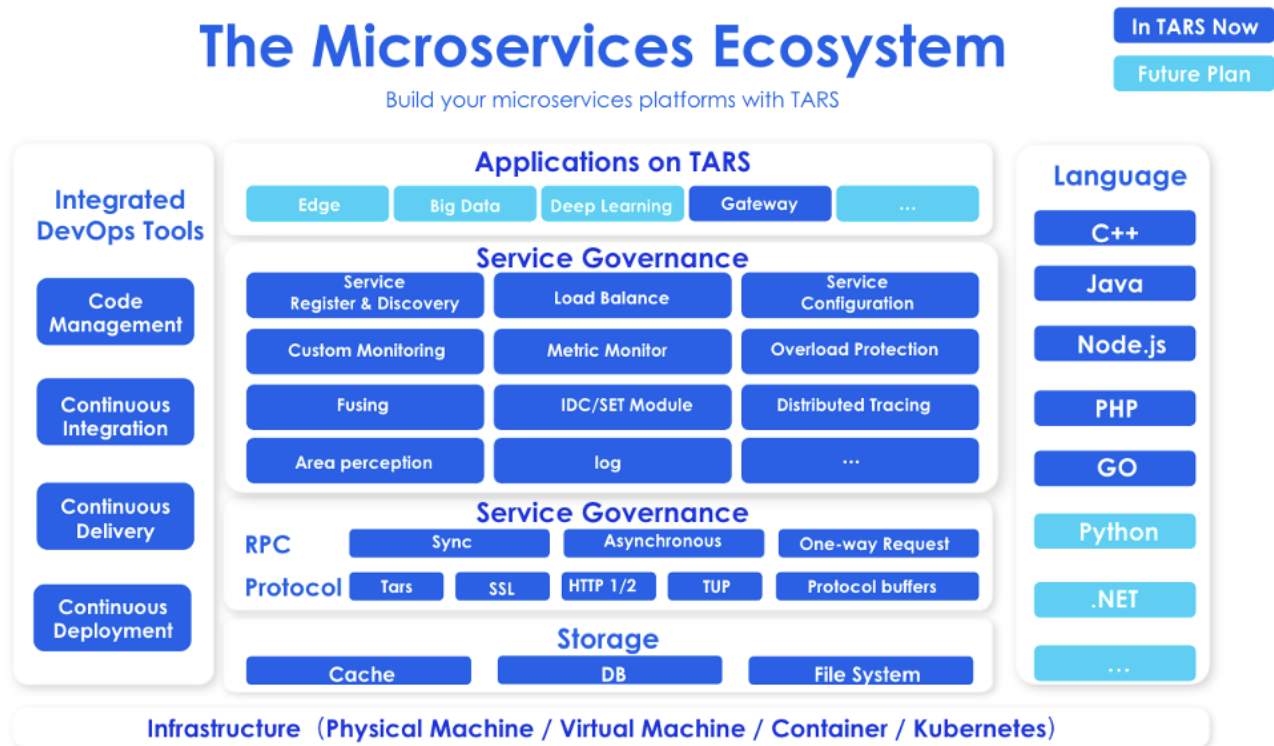
- In Teacher mode
 - You will see the classroom as a teacher's view.
 - You can see some students are in the classroom and are listening to your presentation.
- In Student mode
 - You will see the classroom as a student's view.
 - You can see the teacher and other students on the remote side.

The whole architecture, shown below, consists of three nodes: Jenkins Master, Tars Master, and Tars Agent with AR/VR Blueprint.



- For the Jenkins Master, we deploy a Jenkins Master for our private lab for testing.
- For the Tars Master, we deploy a Tars Platform for serverless use case integration.
- For the Tars agent, we deploy the Virtual Classroom backend on this node and two front-end clients as Virtual Classroom teacher and student on KVM.

It's not a very difficult architecture. As I mentioned before, the TARS Framework plays an important role as PaaS Layer to governance backend services. Then let's go a little bit further with TARS.



TARS is a high-performance microservice framework based on name service and Tars protocol, an integrated administration platform, and an implemented hosting service via a flexible schedule. TARS adds support for ARM, x86, and multiple platforms, including macOS, Linux, and Windows.

TARS can quickly build systems and automatically generate code, taking into account ease of use and high performance. At the same time, TARS supports multiple programming languages, including C++, Golang, Java, Node.js, PHP, and Python. TARS can help developers and enterprises quickly build their own stable and reliable distributed applications in a microservices manner to focus on business logic to improve operational efficiency effectively.

TARS is working in PaaS. It can run on physical machines, virtual machines, and containers, including Docker and Kubernetes. You can store TARS service data in Cache, DB, or a file system.

TARS framework supports Tars protocol, TUP, SSL, HTTP 1/2, protocol buffers, and other customized protocols. Tars protocol is IDL-based, binary, extensible, and cross-platform. These features allow servers to use different languages to communicate with each other using RPC. And TARS supports different RPC methods like synchronous, asynchronous, and one-way requests.

TARS has multiple functions to governance services and can integrate other DevOps Tools such as Jenkins as the IEC Type 4.

For release 4, we update TARS to version 2.4.13, which supports multiple new features, such as API Gateway named TarsGateway, that support to transfer of HTTP protocol to TARS protocol.

We provided a serial of TARS API to the Akraio API map. Including application management APIs and service management APIs.

And last year, on March 10th. TARS evolved to a non-profit microservices foundation under the Linux Foundation umbrella.



The TARS Foundation is a neutral home for open source Microservices projects that empower any industry to quickly turn ideas into applications at scale. It's not only TARS but a microservices Ecosystem. It will solve microservices problems with the TARS characteristics, such as:

- Agile Development with DevOps best practices.
- Built-in comprehensive service governance
- Multiple languages supported
- High performance with Scalability

Then there is a summary of what we have done in release 4.

We focus on building the infrastructure and virtual classroom application, which I mentioned before.

We deploy the AR/VR Blueprint in the environments: Parserlabs, and use Jenkins to make CI/CD available in the Blueprint.

For release 4 we update TARS to version 2.4.13 which supports multiple new features, such as API Gateway named TarsGateway that support to transfer of HTTP protocol to TARS protocol.

Then we have passed the security scan and the validation lab check. The logs are shown here.

For the coming Release 5. We are planning to deploy IEC Type 4 AR/VR on Kubernetes by K8STARS.

K8STARS is a convenient solution to run TARS services in Kubernetes.



It maintains the native development capability of TARS and automatic registration and configuration deletion of name service for TARS.

It supports the smooth migration of original TARS services to Kubernetes and other container platforms.

The K8STARS is a non-intrusive design, no coupling relationship with the operating environment.

Service can be run as easy as one command like:

```
kubectl apply -f simpleserver.yaml
```

If you are interested in TARS Foundation or TARS, please feel free to contact us via email tars@tarscloud.org

Welcome to join us to build the IEC Type 4 AR/VR blueprint and microservices ecosystem together.