openEuler: Bringing new opportunities to the diversified computing era

Wei Xiong
Member of Technical Committee, openEuler

Three Tough Challenges for the OS Industry

1. The fast development of chipset brings tough challenge for the OS development
2. OS needs more aggressive innovations to bring things to be lighter and faster
3. The gap between server/cloud and the embedded system

openEuler's way to resolve

Aggressive

Is Linux good enough?

Linux has been the most successful OS system for the industry. Everything looks fine.

Is it true?

Suppose you have a new chip:

Ensure patch accepted by kernel → Make sure the OS vendor accept → Deliver those features to customer by OS vendor

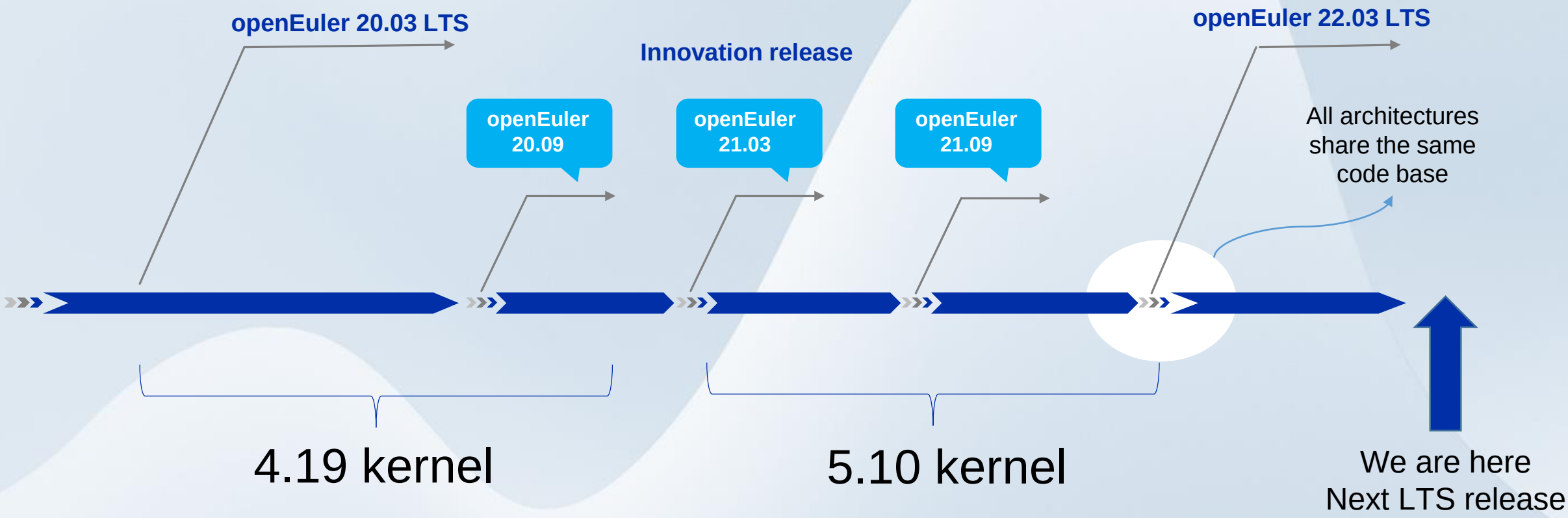
And

- kernel has 4.0, 4.1… 5.1,5.2…, 6.0.6.1…
- OS vendor has 6.0 release, 7.0 release
- 6.0 release has SP1, SP2, SP3…
- A lot of OS vendors

In addition, we will have more chip architectures

x86 **ARM64** **RISC-V**

How does openEuler handle the new scenarios?



- openEuler has more aggressive release cycle to meet the aggressive chipset development, and arranges innovation releases to cover immature features
- Up to 22.03 LTS, x86, ARM64, RISC-V can share the same code base
- openEuler can contain features whose mainline is not accepted or need long time for acceptance

It looks like not aggressive enough

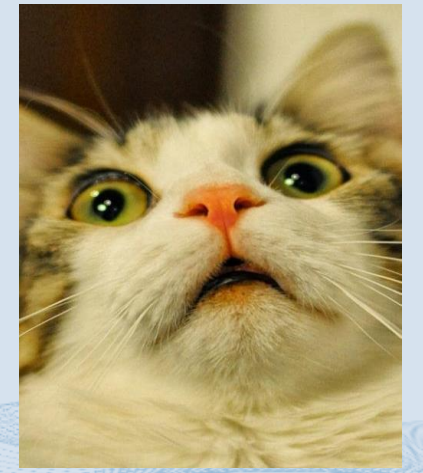
More complicated

Suppose you are a cloud admin who is in charge of 100,000 machines

You will face

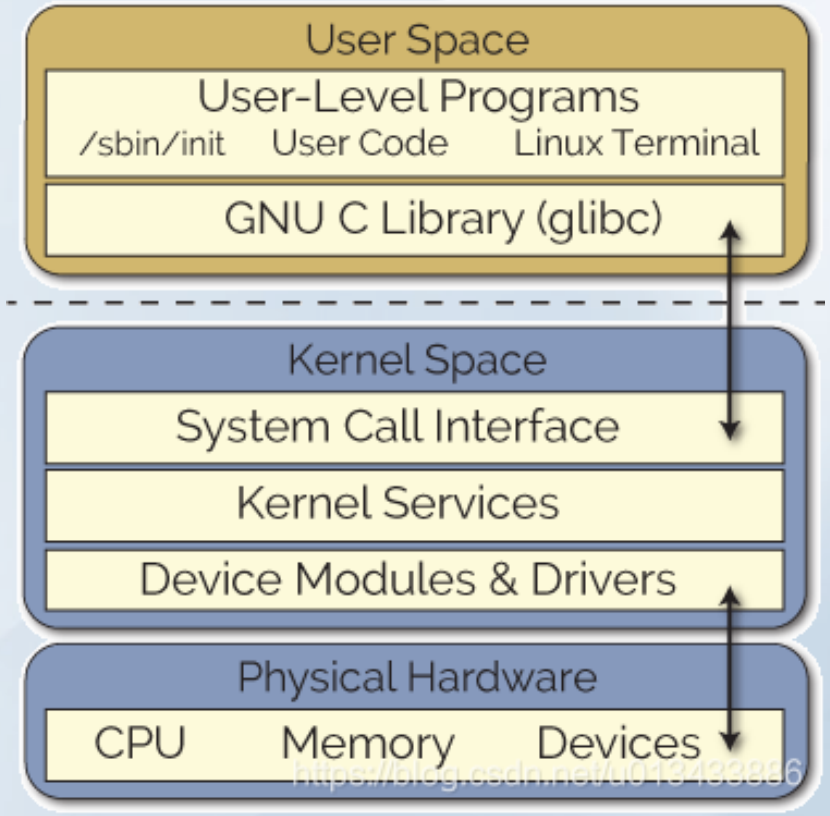
- CPU, GPU, DPU, TPU, xPU…
- 40+ new CVEs every week
- OS release update per month or even per week

They all need you to upgrade KERNEL.



The kernel faces great challenges from chip vendors NOW

Kernel is the interface of hardware and software



Is it enough for chip vendor to put code into kernel? Actually, it's much more complicated.

What makes it such a big mess?

The Root Cause:

Kernel is highly coupled and unchangeable

Changing anything means changing everything

Idea from the openEuler Community

Make everything in kernel as a service, we call it **KaaS**

Redesign a driver framework to make driver more isolated.

Use eBPF to make those modules to be flexible and reconfigurable.

The idea is to create a more lightweight and streamlined kernel.

However,
an OS encompasses more than just the kernel.

Make VM and container lighter and faster

Rebuild basic components of IT infrastructure.

Container →  iSulad

Virtualization → **StratoVirt**

- Lighter
- Faster
- Component-based
- More secure
- …

Make virtualization and container light enough, build a new infrastructure from IoT to cloud.

Is the systemd daemon too big and heavy?

Rebuild the No.1 process in Linux system

sysMaster: use Rust to rewrite process 1,

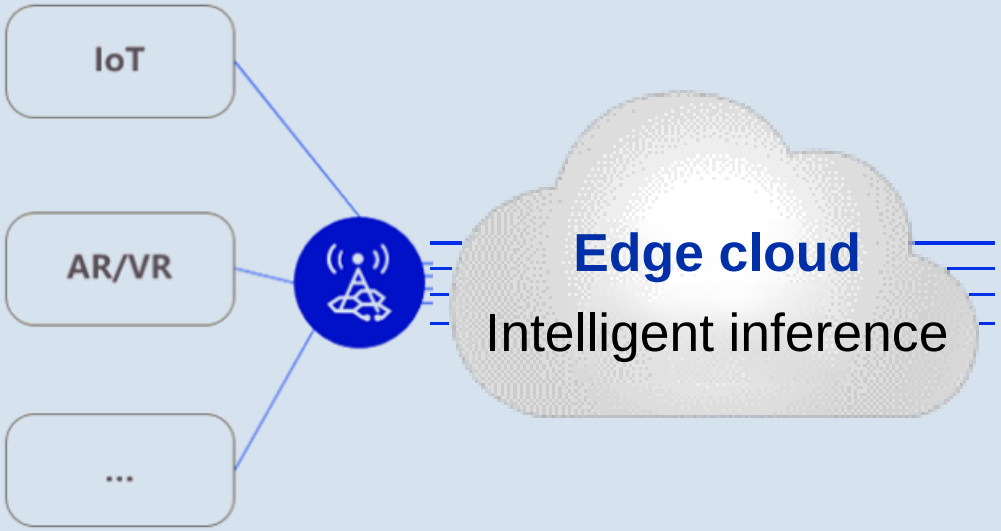
make it lighter and safer

What we have done is not only for replacement of the system, but for a bigger vision

A universal OS platform

The gap between server and embedded

WindRiver
WRLinux/Yocto



CentOS/koji
SUSE/OBS
Ubuntu

Conventional
Different departments often establish independent physical networks and independent server resources.

New
Through the technical means of server virtualization and network virtualization, resources are integrated and diversified teaching methods are provided, which truly helps the teaching and material research of schools.

The gap between server and embedded

When the industry is divided in to two parts

- **Everyone will have to pay double efforts to do the compliance work, especially for chipset vendors**
- **Applications cannot be deployed freely among different machines**

But OS is OS

Actually, no matter for an embedded or a server OS

Every OS is a collection of PACKAGEs.
Alternatively, different OSs can be thought of as distinct compilations of packages.

So, we come up with an idea!

openEuler's Idea

openEuler is **a Linux distribution and a compose system** to make OS for different scenarios, we call it "tenon/socket structure" which is used by Chinese traditional buildings.

Components of OS

Compose system

What is openEuler?

openEuler is an OS platform which builds different OSs

Below is our idea

To summarize:

- **Optimize OS components for lighter, modular designs that run on multiple hardware platforms.**
- **openEuler uses these components to create tailored OS solutions for different scenarios.**

Some tips

openEuler does not abandon those "old" components,

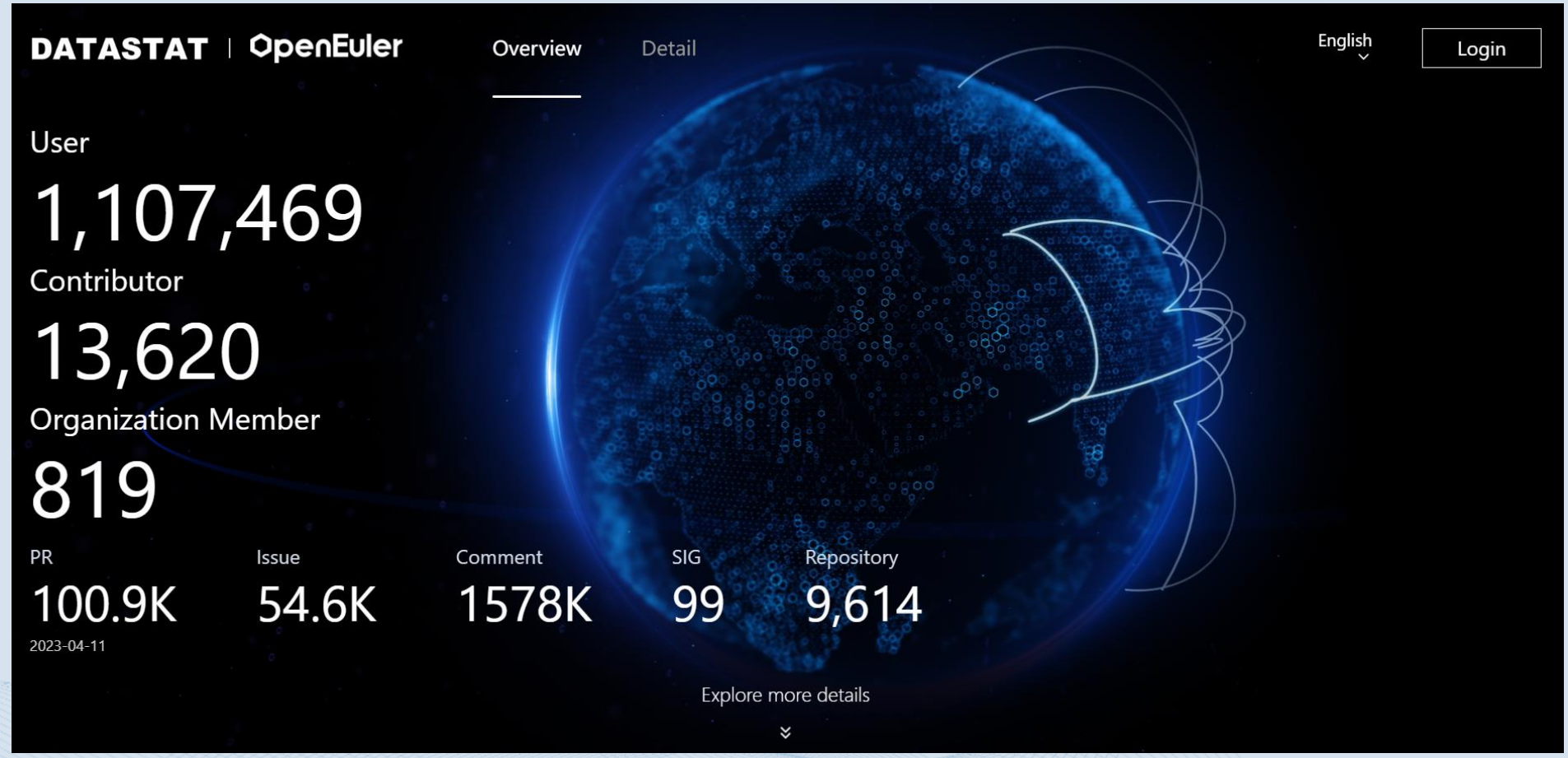
openEuler only provide a second choice.

More than that!!!

- **A-Tune: AI-based performance tuning tool**
- **KubeOS: OS designed for cloud native**
- **Kernel hot replacement: enhanced cloud OPS technology**
- **BiSheng JDK: JDK optimization for ARM64 and RISC-V**
- **IMA:** support full IMA capability
- **NFS+**: enhanced NFS protocol and implementation, up to 6x speed and more robust
- **kmesh**: high performance for micro services in cloud
- **Gazelle**: high-performance use space network stack
- **AI-OPS**: use AI tech to help DevOps
- …

More than that!!!

300+ projects set up in openEuler community

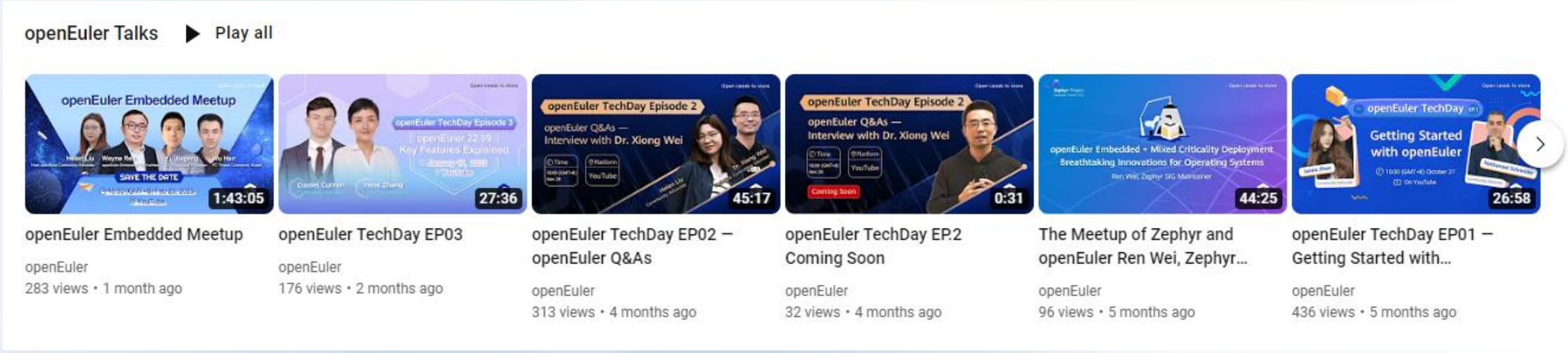


openEuler tries to do something different

openEuler is an open source project under the OpenAtom Foundation

https://www.openeuler.org/en/

openEuler Tech Days and Meetups on YouTube, welcome to Join!



How to engage

@openEuler
https://twitter.com/openEuler

r/openEuler
https://new.reddit.com/r/openEuler/

openEuler
https://www.youtube.com/@openeuler/

openEuler
https://www.linkedin.com/company/86315548/

Official website

LinkedIn newsletter

Join SIGs

Download

Thank you