The Standards People

Insights for Edge Software Developers

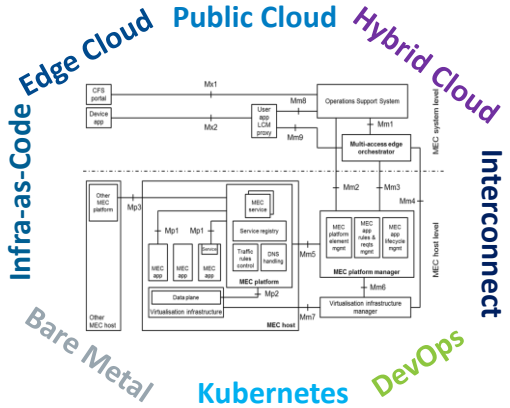
Presented by: **Oleg Berzin**
Equinix, Co-chair of the Linux Foundation Edge Akraino Project

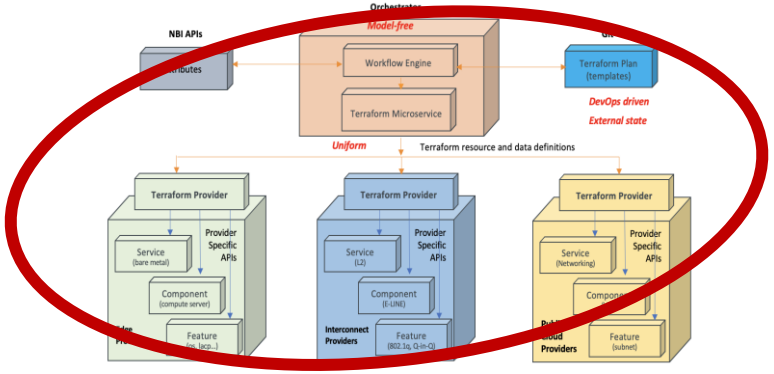
For: **everyone**

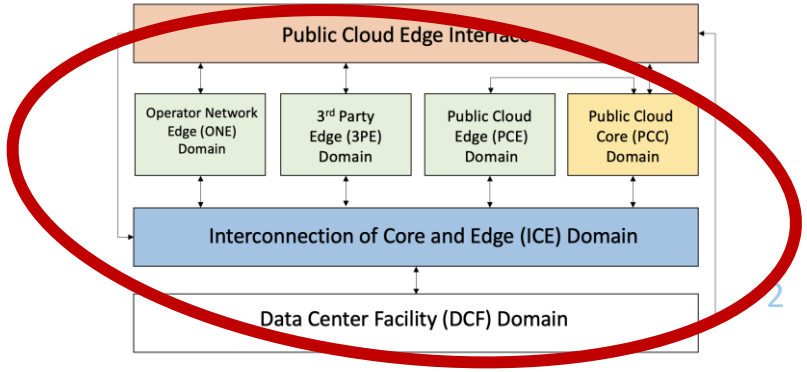
Episode #7 – DevOps MEC INfrastructure Orchestration

In this episode …

- We will learn how:
  - Major industry trends influence MEC infrastructure
  - Infrastructure-as-code can be used to apply DevOps framework to building and testing MEC infrastructure
  - Open-source projects help build tools for orchestrating MEC infrastructure and services

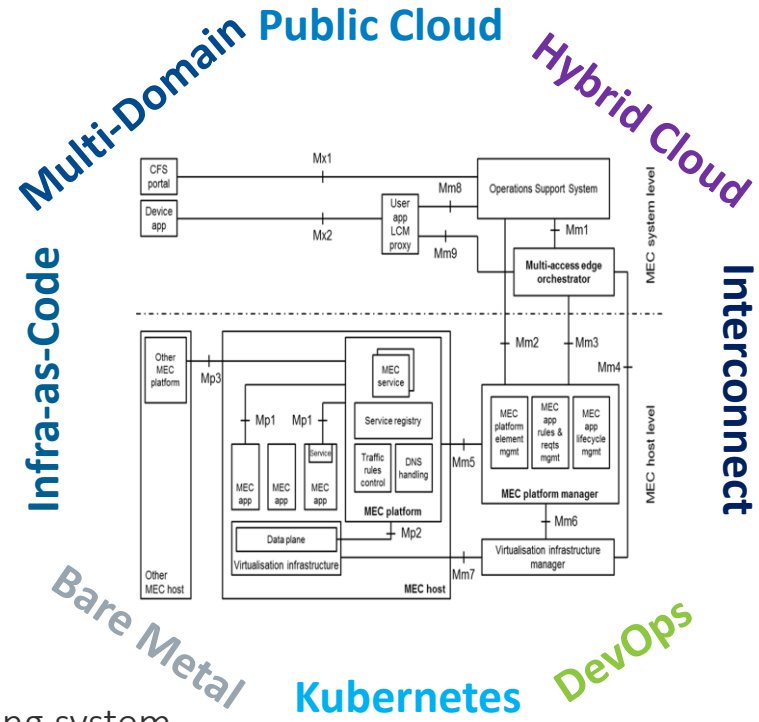






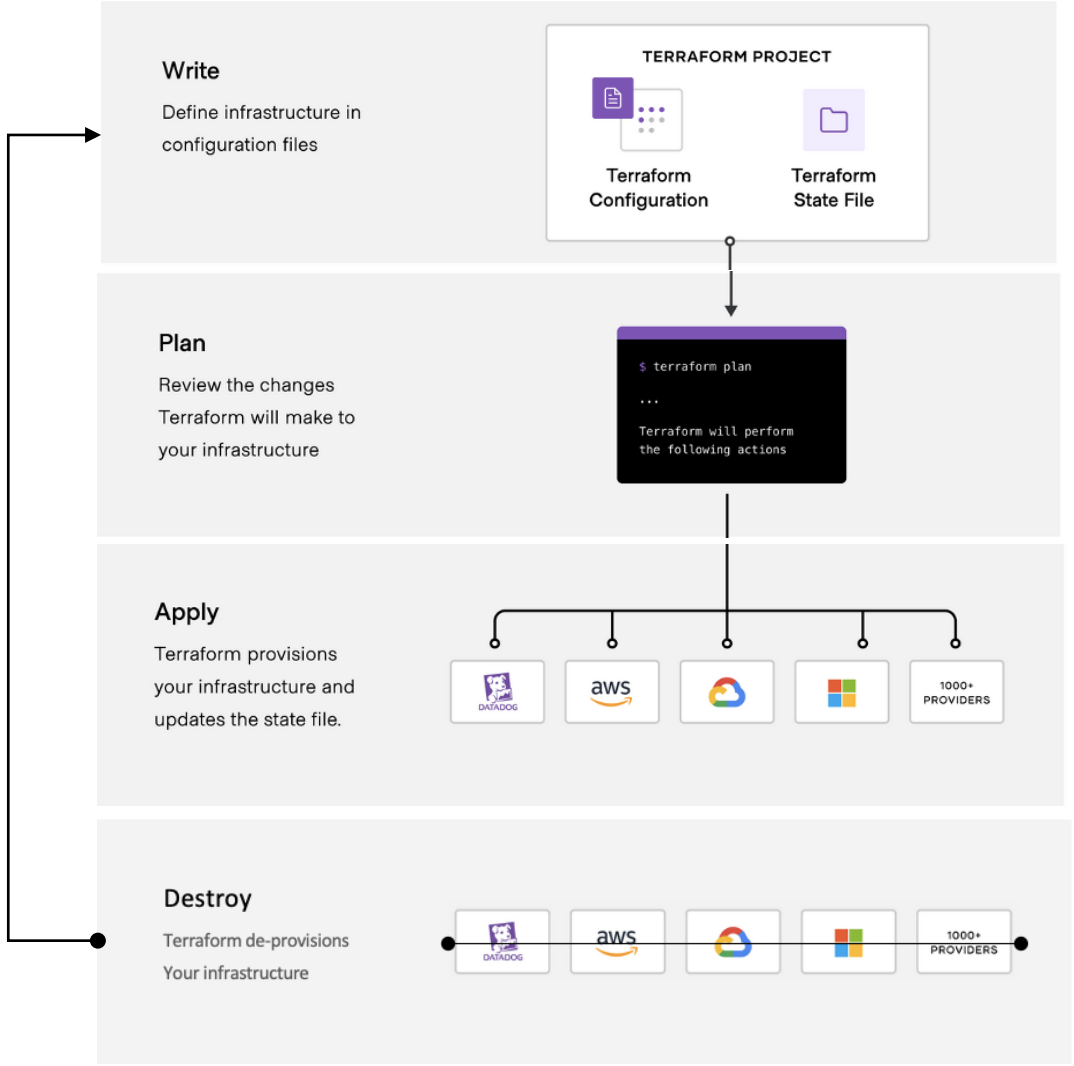
Major industry trends influencing MEC infrastructure

- Public cloud driven edge computing
  - Edge computing infrastructure & resources are increasingly provided by public clouds (e.g., AWS Outposts)
- Hybrid infrastructure
  - Most practical deployments of edge infrastructure and applications are hybrid in nature, where an application deployed at the edge needs services residing in the core cloud(s) to function
- Multi-Domain
  - Individual service domains (e.g., edge, cloud, network fabric) present their own APIs and/or other provisioning methods, making end-to-end deployment challenging
  - A Multi-domain orchestration solution is required to handle edge, cloud and interconnection in a uniform and consistent manner.
- Interconnection and Federation
  - Need for efficient and performant interconnection and resource distribution between edge and cloud as well as between distributed edges proximal to end users.
- Bare Metal orchestration
  - Many orchestration solutions assume that the bare metal hardware and basic operating system resources are available for the deployment of virtualization and application/services layers.
- Developer-centric capabilities
  - End-to-end infrastructure activation, application deployment, configuration and interconnection
  - Continuous Integration (CI)/Continuous Delivery (CD) environments for DevOps



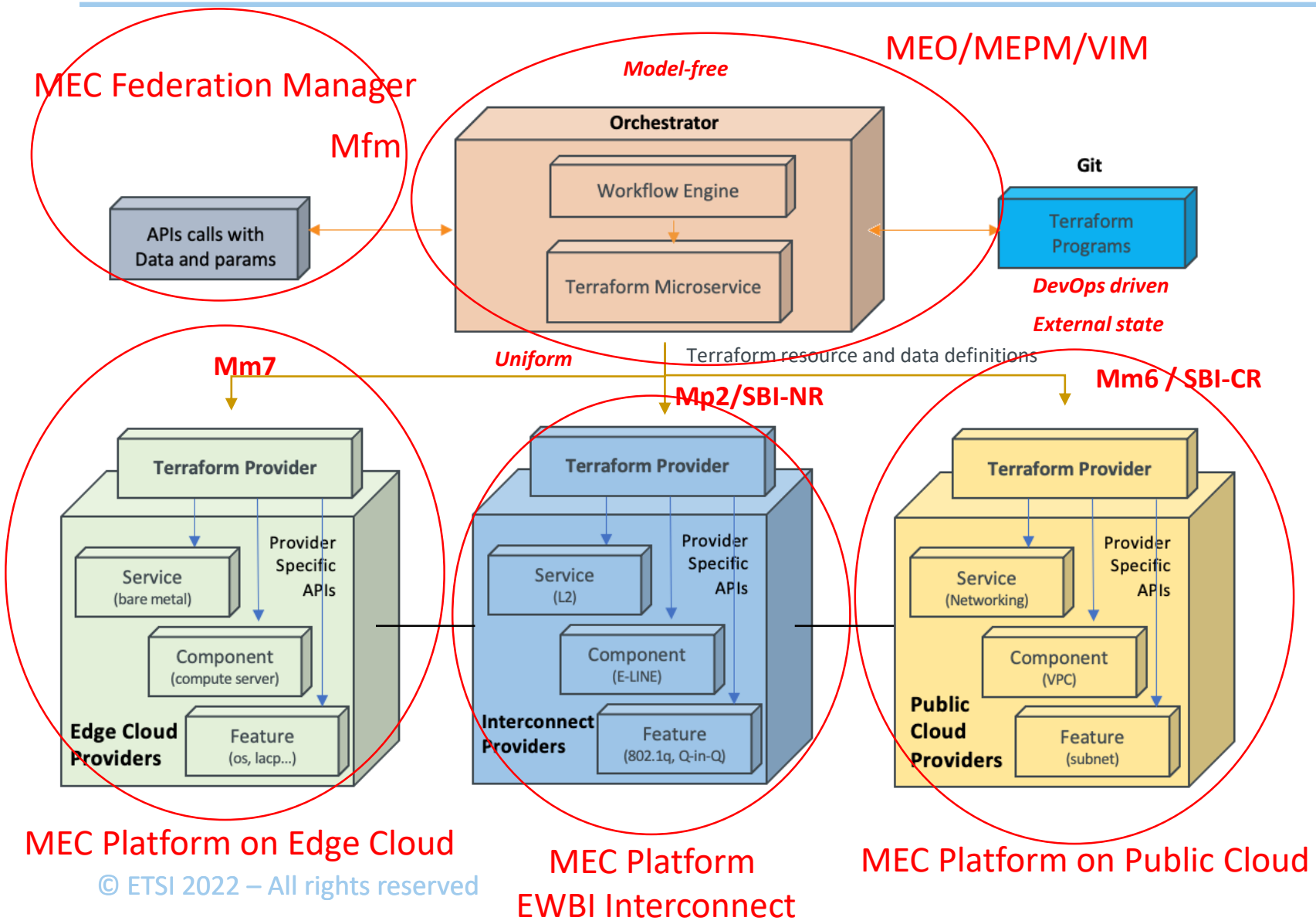
How Infrastructure-as-Code works: Terraform

- Terraform is an infrastructure as code tool that lets you define both cloud and edge resources in human-readable configuration files that you can version, reuse, and share.
- You can then use a consistent workflow to provision and manage all of your infrastructure throughout its lifecycle.
- Terraform can manage low-level components like compute, storage, and networking resources, as well as high-level components like Software as a Service (SaaS) features.



Source: https://www.terraform.io/intro

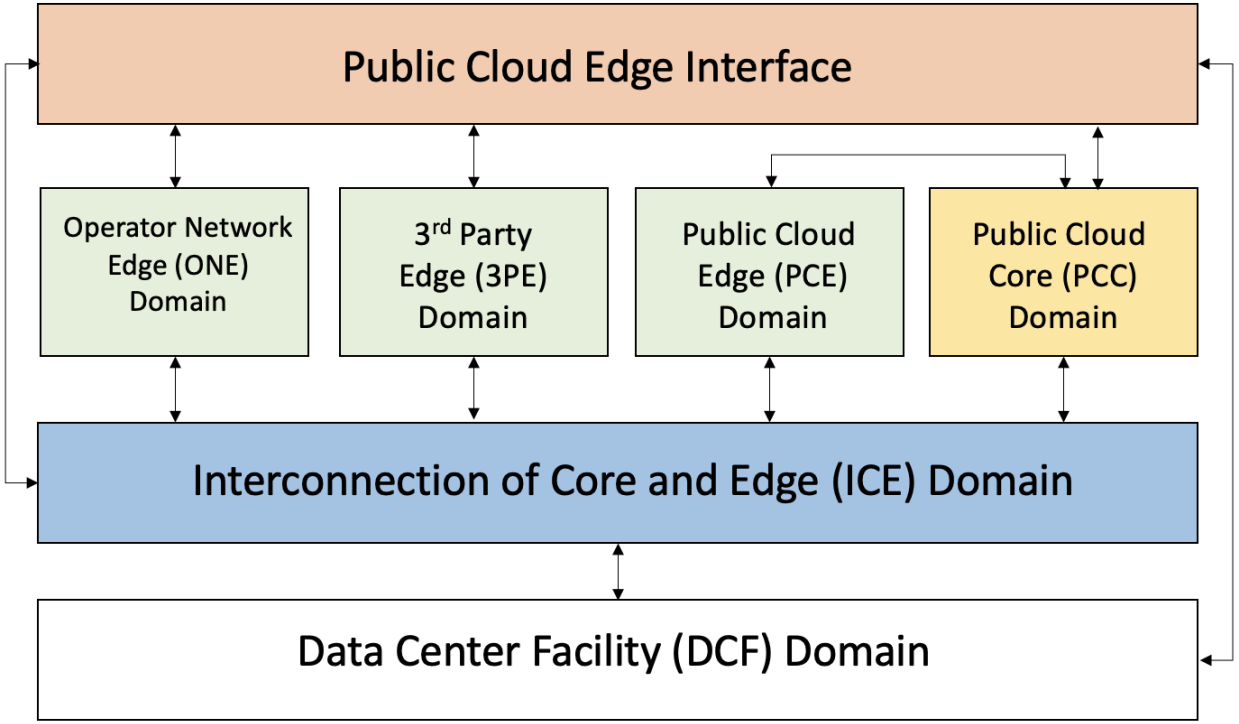
Leveraging Terraform for orchestrating MEC infrastructure



- **Uniform** - use of the same infrastructure orchestration methods across public clouds, edge clouds and interconnection domains.
- **Model-free** – the orchestrator does not need to understand the details of the individual infrastructure domains (i.e., implement their models). It only needs to know where to retrieve the Terraform plans for the domain in question and execute the plans using the specified provider..
- **DevOps driven** – the Terraform plans can be developed and evolved using DvOps tools and processes.
- **External state** – the state of infrastructure resources created by the orchestrator is stored outside of the orchestrator itself, making it stateless with respect to the infrastructure

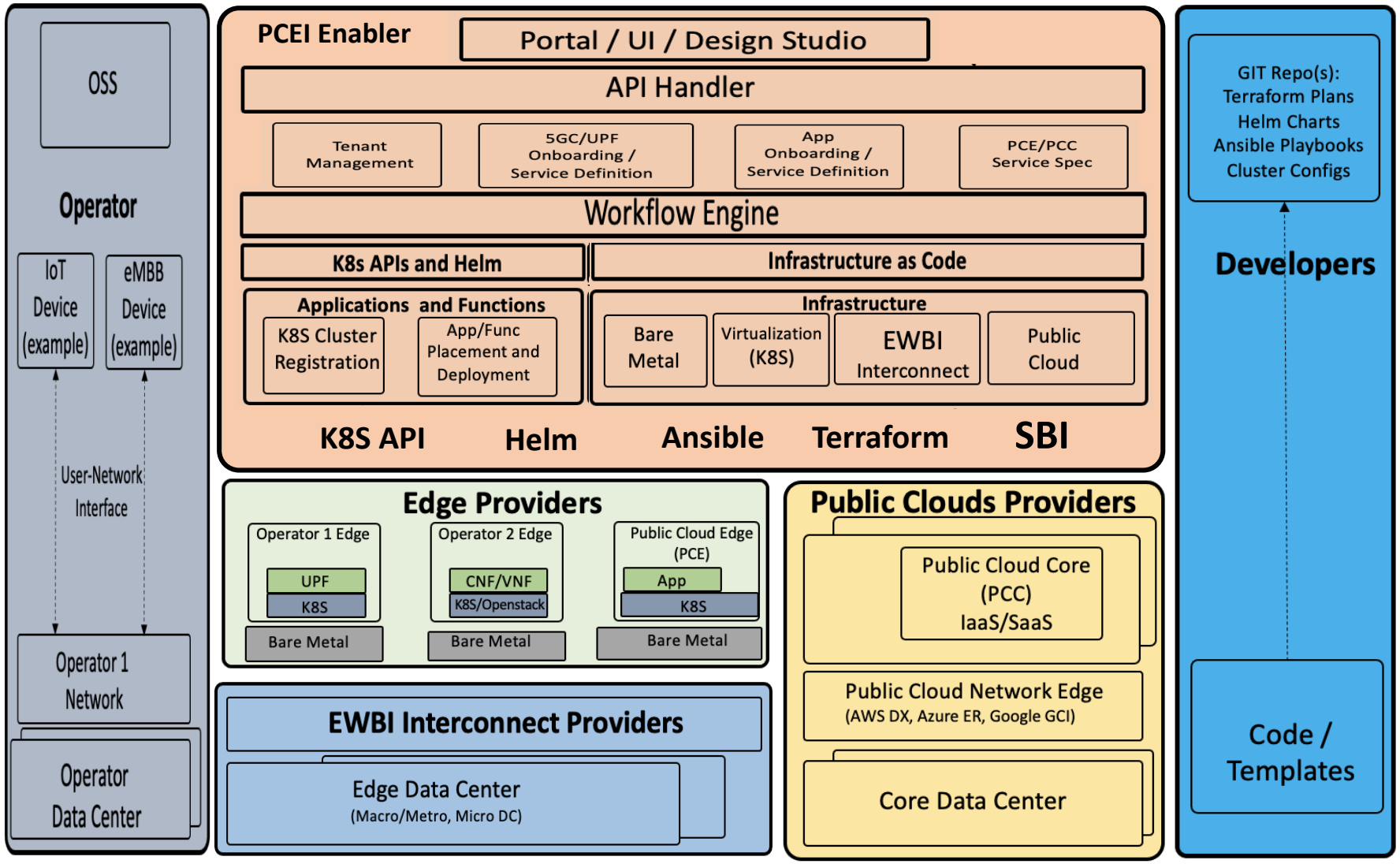
Leveraging open-source projects for building and testing MEC infra and deploying MEC services

Akraino Public Cloud Edge Interface (PCEI) Blueprint

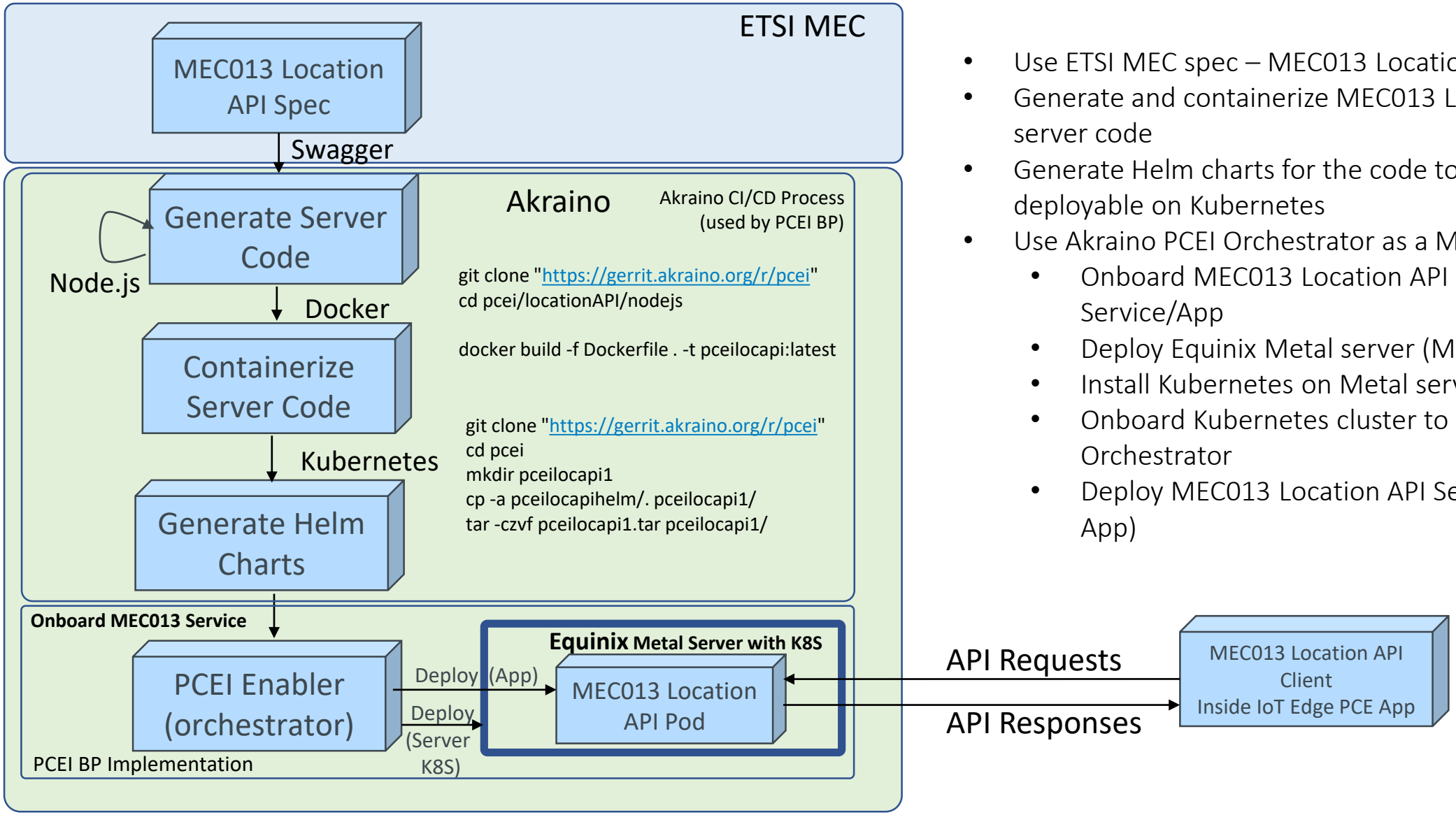


The purpose of Public Cloud Edge Interface (PCEI) Blueprint is to develop a **set of**, **orchestration functionalities**, **edge capabilities and open APIs** for enabling Multi-Domain Interworking across the Operator Network Edge, the Public Cloud Core and Edge, the 3rd-Party Edge as well as the underlying infrastructure such as Data Centers, Compute Hardware and Networks.

Enabling DevOps MEC Infra Orchestration with Akraino PCEI



Enabling DevOps MEC service implementation and deployment with Akraino PCEI: MEC Location API



- Use ETSI MEC spec – MEC013 Location API
- Generate and containerize MEC013 Location API server code
- Generate Helm charts for the code to make it deployable on Kubernetes
- Use Akraino PCEI Orchestrator as a MEO/MEPM to:
  - Onboard MEC013 Location API server as a Service/App
  - Deploy Equinix Metal server (MEP/MEC Host)
  - Install Kubernetes on Metal server
  - Onboard Kubernetes cluster to PCEI Orchestrator
  - Deploy MEC013 Location API Service (as MEC App)

Conclusions and further deepening

What we have learned:

- Major industry trends influencing MEC infra
- The concept of Infra-as-code and how it can be used to apply DevOps framework to building and testing MEC infra
- Akraino Public Cloud Edge Interface (PCEI) blueprint and how it enables orchestration of MEC infra and services

Interested to learn more?

- See Akraino PCEI WiKi page
- Watch Akraino PCEI Release 5 Demo
- Follow the next episodes of the MEC TECH Series ☺

Enjoy the