EMCO: Project Overview and Architecture
Agenda

• Drivers for Edge Computing
• How EMCO is addressing challenges
• Architecture overview
• Features at high level
• OpenAPIs and brief overview
• Use cases
• Q&A’s
Drivers
- Latency/Physics
- Bandwidth/Economics
- Context/Proximity
- Privacy/Legal

Trend: Geo Distributed Computing trend with Edge-computing
Geo-Distributed Computing - few use cases

**5G Use case**
- Large Number of sites
- Computing (Apps across sites) – MEC
- Multiple tenant applications along with operator CNFs.
- Workload types - VMs, VNFs, CNFs, CNAs and Functions (FaaS)
- Note: K8s is becoming choice of workload orchestrator in each cluster

**Universal CPE Use case**
- SDEWAN CNF
- Traffic Hub
- NW Analytics

**Enterprise (Retail) use case**
- SDEWAN CNF
- SASE frmwrk
- SASE Services

Multi Edge/Cloud computing scale is similar (or even higher) to Hyper-scalers’ scale
Now Telcos, MSPs and Enterprises need @scale Orchestration and Automation solutions
E2E Edge Stack

- **OSS and BSS**
- **5G Slicing Orchestrator**
- **Service Orchestrator**

**Distributed Applications & Network functions (CNF, VNF, Apps)**

**Infrastructure Orchestrator**

**Edge/Telco Extensions**
- K8s, Hypervisor, runtimes
- Linux
- Private Cloud / On Prem K8s cluster

**Edge/Telco Extensions**
- K8s, Hypervisor, runtimes
- Linux
- VMs
- Public Cloud K8s Cluster

**Edge/Telco Extensions**
- K8s, Hypervisor, runtimes
- Linux
- Edge K8s Cluster
Service Orchestrator – Big Picture

- One Click deployment of complex applications & network services across multiple K8s clusters
- Comprehensive Status monitoring of deployed complex applications
- One Service Orchestrator for both CNF/CNA, VNF/VMs
- Self Service Portal for multiple tenants
- Comprehensive Analytics platform for Day2 operations
- App Centric infrastructure configuration (Service Mesh, SDWAN, L2/L3 switches)
Needs/Requirements – Preparation

Registration of Clusters
- Cluster labels
  (Example: Cell tower Edge, CO Edge etc.)
  Needed for identifying multiple clusters
- Cluster specific configuration
  (Few: ISTIO CA provisioning; Virtual/Provider network preparation)

Tenant registration
- Ability to use tenant specific OAUTH2 servers for authenticating tenant admins
- Tenant level isolation via RBAC rules
- Logical Cluster provisioning across multiple selected clusters
- Logical Cluster user and permission provisioning
Needs/Requirements – Application deployment design

Service Orchestrator

Cluster Provisioning
Tenant Provisioning
Logical Cluster Provisioning

App/Service onboarding
Define placement policies
Define Placement constraints

App Onboarding
(Complex Apps & Network Services)

Multiple deployment profiles to ensure same APP can be instantiated multiple times

Placement policies to replicate and distribute workloads across clusters

Placement constraints:
Affinity and Anti-Affinity;
Platform capabilities;
Latency; Cost
Requirements – Workload Customization & Connectivity management

No changes to helm charts/K8s description of applications

Each deployment may have its own customization

Connectivity intent provisioning
- Enabling inter-micro service communication within or across clusters
- Enabling communication to external entities
  - With/Without Mutual TLS
  - Multi Cluster DNS management

Dynamic provisioning with LCM of Applications

Extensible framework to add new capability controllers
Requirements – Operations

**Service Orchestrator**

- **Cluster Provisioning**
- **Tenant Provisioning**
- **Logical Cluster Provisioning**

### Cluster Provisioning
- App/Service onboarding
- Define placement policies
- Define Placement constraints

### Tenant Provisioning

### Logical Cluster Provisioning
- Workload Customization
- Automation of connectivity and security
- Continuous App monitoring
- Analytics & Closed loop automation
- Day2 Config Controllers

#### Analytics framework
- Metric collection across clusters & apps
- Long term central store (Time Series)
  - Training framework
  - Closed loop policy management

#### Day 2 Configuration
- Configuration of apps/network-functions that are already deployed.
- Various types of configurations (CR based, RESTful based or Netconf/yang based)

**Continuous monitoring of complex Application (Across clusters, apps and micro-services)**

**Comprehensive report on the application status**
EMCO – Edge Multi Cluster Orchestrator

EMCO is an implementation of Service Orchestrator
Addressing majority of requirements; Extensible architecture allows new automation requirements

- **Cluster Registration Controller** registers clusters by cluster owners
- **Distributed Application Scheduler** provides simplified, and extensible placement; tenant mgmt; LCM implementation
- **Hardware Platform Aware Controller** enables scheduling with auto-discovery of platform features/capabilities; Others: Cost, Power Savings, Latency aware… (WIP)
- **Distributed Cloud Manager** presents a single logical cloud from multiple edges
- **Traffic Connectivity controller** auto-configure service mesh (ISTIO) and security policy (NAT, firewall), DNS and SLB entities of edges - WIP
- **Day2 generic configuration** configures Day2 configuration of any app/network function via templates & configs - WIP
- **Resource Synchronizer & Monitoring** synchronizes resources across multiple edge/cloud platforms and then monitors the status of deployed resources
EMCO Features

• Extensible Architecture
  • Placement controllers
  • Action controllers
• Intent based architecture
• On-demand instantiation of applications on K8s clusters
• Intelligent selection of clusters to place the workloads
• Tenant Isolation using logical clouds
• Customization of resources in the applications based on clusters
EMCO Features Contd.

- Monitoring of resources deployed by EMCO
  - Notification framework for per app, per cluster, per resource
- Automation of service mesh and other connectivity & security infrastructure
- Dependency and order of priority of application deployments between clusters
- Update and Rollbacks
  - Update applications, resources for existing applications and/or add/delete clusters for applications
- On-demand scale-out of the applications
EMCO Features Contd.

• Nodus Support ([https://github.com/akraino-edge-stack/icn-nodus](https://github.com/akraino-edge-stack/icn-nodus))
  • Network Configuration Management (NCM) Controller
  • OvnAction Controller
  • SFC Controllers
• Referential Integrity
• Service Discovery Controller
EMCO Features (Under development)

• Rsync Plugin Architecture (available in 21.12 release)
  • K8s Cluster
  • WIP/Planned Plugins
    • Fluxv2
    • Azure Arc
    • Google Anthos
  • Others…. 
EMCO OpenAPI’s

• API Concepts
  • Projects
  • Composite Apps, Composite Profiles
  • Deployment Intent Group
  • Generic Placement Intent
  • Controllers
  • Clusters
  • Logical Clouds
  • Traffic controller, Nodus, HPA, ….

• Link to the API

https://gitlab.com/project-emco/core/emco-base/-/raw/main/docs/swagger-specs-for-APIs/emco_apis.yaml
Test Usecases

- vFW
- Prometheus
- Free5gc
vFW Use Case - Standalone

**Connectivity Info**
- virtlet containers

**Definitions**
- **Unprotected Network**
  - packetgen
  - firewall
  - sink

- **Protected Network**
  - protected-network

**Professionals**
- vFW chart
- vFW profile

**Templates**
- firewall
- unprotected-network
- protected-network
- onap-network

**Sub-charts**
- packetgen
- sink
vFW – set up clusters and controllers

clm
- cluster APIs

ncm
- network intents
- scheduler

ovn4k8s controller

mongodb

ectd (appcontext)

orchestrator
- composite app APIs
- controller
- register API
- generic placement
- controller
- deployment intent
- scheduler

ovnaction
- workload / interface APIs
- ovnaction controller

rsync

(edge01, edge02)

test script

(ovnaction, rsync)
vFW – set up the composite app and intents
vFW – apply the network intents

1. test script

2. ovn4k8s controller
   - scheduler
   - controller register API

3. ectd (appcontext)
   - mongod

4. orchestrator
   - composite app APIs
   - generic placement
   - deployment intent
   - scheduler

5. rsync
   - ovnaction
     - workload / interface APIs
     - ovnaction controller
vFW – instantiate a specific deployment intent

1. Test script

2. Schedule

3. ovn4k8s controller

4. ovnaction controller

5. rsync

6. orchestrator

7. ectd (appcontext)

- clm cluster APIs
- ncm network intents scheduler
- mongodb
- ovn4k8s controller
- generic app APIs placement
- deployment intent
- register API
- workload / interface APIs
- edge01
- edge02
EMCO Journey

- V1 version of the EMCO API are called k8splugin under multicloud-k8s project in ONAP.
- V2 incubated as a subproject, ONAP4k8S, under ONAP since 2019.
- Garnered following of several companies in ONAP.
- Moved under the Intel OpenNESS umbrella/repo in late 2020.
- Renamed EMCO, source drops provided back to ONAP users. Several releases of EMCO were made under OpenNESS.
- ONAP community formally requested that EMCO return to open governance March 2021 - **LF Leadership suggested LFN**.
- Intel approved move back to LF May 2021; formation commenced July ’21.
- TSC established August 24, 2021 (11 Representatives from different companies).
- LFN TAC approved EMCO as a sandbox project under LFN umbrella on September 22nd.
- Active open source EMCO development moved to LFN under gitlab on October 1st.
EMCO under LFN

- **Project name:** EMCO
- **Project creation date:** July 20, 2021 (Formation KickOff)
- **Project license:** Apache 2.0
- **Multiple company plans to contribute to code base**
  - Intel, Aarna, Cango, Verizon, Tech Mahindra
- **EMCO Gitlab project:**
  - Includes emco-base and emco-gui
  - Emco-base contributed by Intel and emco-gui by Aarna Networks
  - Repos: [https://gitlab.com/project-emco/ui/emco-gui](https://gitlab.com/project-emco/ui/emco-gui)  [https://gitlab.com/project-emco/core/emco-base](https://gitlab.com/project-emco/core/emco-base)
- **Release Schedule:**
  - Releases Planned for Sept (seed code) and Dec 2021
    More detail on Roadmap slide below
EMCO Integrations

Part of these commercial solutions
Part of AMCOP solution from Aarna: [https://www.aarananetworks.com/amcop](https://www.aarananetworks.com/amcop)
Commercial Support

Few Blueprints in LFE/Akraino use EMCO for Multi Cluster Orchestration
free5GC deployments using EMCO by Aarna networks

ONAP uses EMCOv1 to Onboard/design network services and deploy on K8s clusters

ONAP Slicing Orchestrator uses EMCOv1 for Day 2 LCM
Thank You!!
Backup
EMCO Vision

- Be a comprehensive geo-distributed Cloud native application orchestrator
- Be a Multi-Party and Multi-Cloud Orchestrator
- Be an orchestrator for Network services and Enterprise applications
- Be an orchestrator for convergence of Network services and Enterprise applications
- Be an orchestrator for Distributed Clouds with Edge-computing
What is not in the scope of EMCO?

<table>
<thead>
<tr>
<th>EMCO does not expose ETSI and Tmforum APIs.</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMCO does not deploy workloads in non-K8s environments</td>
</tr>
<tr>
<td>EMCO CNF/App configuration is limited to K8s CR based apps/CNFs. It does not support NetConf, CLI and other mechanisms as of now.</td>
</tr>
<tr>
<td>EMCO does not include Analytics stack</td>
</tr>
</tbody>
</table>

Few distributions of third-party service orchestrators leveraging EMCO combine other projects such as ONAP CDS, ONAP DCAE to address brownfield deployments.