5G Super Blueprint Reference Architecture Seattle 2022

An Industry 4.0 Proof of Concept

Version 1.0

Table of Content

List of Figures

List of Tables

Preface

This Document defines the process and the steps performed in the 5G Super Blueprint Reference Architecture 2022.

Version Information

Title	5G Super Blueprint Reference Architecture 2022
Document Ver. No.	Version 1.0
Approved By	
Drafted by	
Doc. Classification	

Version History

Version #	Date	Author(s)	Approved by	Description

5G Super Blueprint Contributors

- Below are the list of contributors of 5G Super Blueprint Reference Architecture 2022.

5G SUPER BLUEPRINT CONTRIBUTORS		
Abhijit Patil – US Navy	Haseem Ahmed - Kratos Defense	Pano Xinos - The Linux Foundation
Amar Kapadia - Aarna Networks	Heather Kirksey - The Linux Foundation	Parthiban N – Wavelabs
Ben Posthuma – GXC	Jason Hunt – IBM	Ranny Haiby - The Linux Foundation
Bharath Kumar – HPE	Jeff Schranz – GXC	Rob Edwards -MATRIXX Software
Bob Heinemann – MITRE	Jill Lovato - The Linux Foundation	Sandeep Panesar – Turnium
Bob Monkman – Intel	Kader Khan – Wavelabs	Satish Sadagopan – IBM
Brandon Wick - Aarna Networks	Kenny Paul - The Linux Foundation	Shanmuga Sundaram – Wavelabs
Byung-Woo Jun - Ericsson Software Technology	Lincoln Lavoie - University of New Hampshire	Sridhar Rao - The Linux Foundation
Casey Cain - The Linux Foundation	Louis Illuzzi - The Linux Foundation	Sundar Nadathur – Intel
Cedric Thienot – Firecell	Martial Ngueko - AT&T	Suresh Balaram – Wavelabs
Daniel Nilsson – TietoEVRY	Martin Skorupski - High Street Technologies	Timo Perala – Nokia
Dave Armbrust – MITRE	Muddasar Ahmed – MITRE	Tunde Okunola – Kaloom
Ganesh Venkatraman – Kaloom	Neil Hoff - US Navy	Victanand Karip – HCL
Hardik Jain – GXC	Oleg Berzin – Equinix	Yogendra Pal - Aarna Networks

Table 1: 5G Super Blueprint Contributors

5G SUPER BLUEPRINT PARTICIPATING COMPANIES & ENTITIES

- Below shown are the 5G Super Blueprint Reference Architecture 2022 participated companies and its entities.

Figure 1: 5G Super Blueprint Participating Companies & Entities

1. Introduction

1.1 Purpose of the Document

- The LF Networking 5G Super Blueprint is a community-driven integration of multiple open-source initiatives coming together to show end-to-end industry use cases demonstrating implementation architectures for end users. It builds upon a long-running 5G Cloud Native Network Proof of Concept and has been showcased at several industry events including the Open Networking Summit, Open Compute Project Summit, and KubeCon + CloudNativeCon. The 2022 version was shown at the ONE Summit 2022 event.
- This document details the build and the integration steps for Visual Inspection IOT use case and each Proof-of-Concept component. This document also lists the issues encountered, workarounds, mitigations, and guidance for furthering the next evolution of the Proof-of-Concept.
- The 5G Super Blueprint is an open industry initiative and all are welcome to join. Learn more here: https://lfnetworking.org/5g-super-blueprint/.

2. Abbreviations

#	Abbreviations	Expansions
1	DCIP	Data Center Infrastructure Provider
2	IOL	Interoperability Testing Lab
3	RAN	Radio Access Network
4	UPF	User Plane Function
5	VM	Virtual Machine
6	vDCO	Virtual Data Center Operator
7	PPE	Personal Protective Equipment
8	MVI	Maximo Visual Inspection
9	IEAM	IBM Edge Application Manager
10	SDO	Secure Device Onboarding
11	IOT	Internet of Things
12	eMBB	Enhanced mobile broadband
13	URLLC	Ultra-Reliable Low-Latency Communication

3. Use Cases

3.1 eMBB (Enhanced mobile broadband)

- eMBB use cases are data-intensive use cases that require high bandwidth. Such cases include cloud and UHD, 8K video streaming, immersive gaming (including AR and VR) gaming, video analytics, immersive event experience, and telemedicine.
- One of the significant ways 5G will deploy across large areas is through fixed wireless access (FWA), which will leverage 5G technologies like beamforming and higher-spectrum bands to deliver wireless broadband to previously unreached coverage areas.
- FWA will make a significant impact on global markets, both developing and developed, such as in the U.S., where sparsely populated, rural areas currently lag far behind cities in broadband access. FWA can deliver speeds comparable to or exceeding those of current fiber-based networks, thus creating a platform for eMBB over vast coverage areas using spectrum bands unavailable to 4G. As 5G eMBB becomes widely available, it can deliver several sub-use cases, such as:
  - **Hot spots:** eMBB can enhance broadband access in densely populated areas, boosting indoor and outdoor coverage in high-rise buildings and crowded city centers.
  - **Broadband everywhere:** Technologies like FWA can offer consistent coverage around the world with minimum speeds of 50 Mbps.
  - **Public transportation:** Broadband access on high-speed trains and other modes of public transport are examples.
  - **Smart offices:** eMBB can deliver high-bandwidth connections to hundreds of users in environments with heavy data traffic.
  - **Large-scale events:** Concerts and sporting events may be served by eMBB, enabling high data rates where hundreds of thousands of people are gathered.
  - **Enhanced multimedia:** eMBB can provide seamless, high-definition video streaming, mobile TV and real-time content over broad coverage areas.
- 5G eMBB will be driven by "the growth in user-generated content and our expectations of being able to stream what we want, where we want and when we want without needing to log onto a Wi-Fi network." Multimedia streaming and entertainment, however, constitute only some of the needs eMBB could meet. Important business use cases include mobile

cloud computing and connected remote smart offices.

- The technical requirements for such enhanced mobile broadband capacity and access are high, and it may be several years before the development of mobile technology and cellular infrastructure can bridge the gap. Still, with 5G networks debuting in cities around the world and public & privately funded FWA initiatives expanding coverage to underserved regions, it seems that the age of seamless 5G eMBB is rapidly approaching.

3.2 URLLC (Ultra-Reliable Low-Latency Communication)

3.2.1 INDUSTRIAL AUTOMATION

- Industrial automation is a key application for URLLC features. Some industrial processes have extremely tight Key Performance Indicators (KPIs) for 5G communications links between sensors, actuators and controllers. Example use cases in this category include the following:
  - Motion control
  - Industrial Ethernet
  - Control-to-control communication
  - Process automation
  - Electric power generation and distribution
- URLLC is one of the enabling technologies in the fourth industrial revolution. In this new industrial vision, industry control is automated by deploying networks in factories. Typical industrial automation use cases requiring URLLC include factory, process, and power system automation.
- Use cases involve communication transfers enabling time-critical factory automation that are required in many industries across a wide spectrum that includes metals, semiconductors, pharmaceuticals, electrical assembly, food, and beverage.

3.2.2 GROUND VEHICLES, DRONES, ROBOTS

- 5G will support communications with and among ground vehicles, drones, and robots. Automated guided vehicles are common in factory applications, and they have requirements for coverage and mobility in addition to URLLC. Robots include fixed and mobile applications, with varying KPIs for communications links depending on the specific application. Example use cases in this category include the following:

- Mobile and industrial robots
- Connectivity for the factory floor
- Factories of the future, including automated guided vehicles

3.2.3 TACTILE INTERACTION

- Tactile interaction refers to a level of responsiveness that works at a human scale. For example, remote health care or gaming applications may require very low round-trip times to convince human senses that the perceived touch, sight, and sound are lifelike.
- These use cases involve interaction between humans and systems, where humans wirelessly control real and virtual objects, and the interaction requires a tactile control signal with audio or visual feedback. Robotic controls and interaction include several scenarios with many applications in manufacturing, remote medical care, and autonomous cars. The tactile interaction requires real-time reactions on the order of a few milliseconds. Example use cases in this category include the following:
  - Tactile internet
  - Extreme real-time communications

3.2.4 AUGMENTED REALITY (AR) AND VIRTUAL REALITY (VR)

- Augmented Reality (AR), Virtual Reality (VR) and Mixed Reality (MR) bring higher bandwidth requirements in addition to URLLC constraints. The main difference between AR and VR is in the uplink requirements is, VR needs low-data-rate pose estimates from the headset, while AR requires images of the view experienced by the user.
- These use cases are the critical IoT applications that will have very high demands on reliability, availability, and low latency, with lower demands on the volume of data, but significantly higher business value. These use cases also fall into the category of mission-critical Machine-Type Communication (MTC).
- The mission-critical MTC is envisioned to enable real-time control and automation of dynamic processes in various fields, such as industrial process automation and manufacturing, energy distribution, and intelligent transport systems. These use cases and applications feature interactions across all categories, human-to-human, human-to-machine, and machine-to-machine.

3.2.5 EMERGENCY, DISASTERS AND PUBLIC SAFETY

- The use cases in this category require robust and reliable communications in case of natural disasters such as earthquakes, tsunamis, floods, and hurricanes. The use cases may require accurate position location and quick communication exchanges between users and systems. Energy efficiency in user battery consumption and network communications are critical in these use cases. Public safety organizations require enhanced and secured communications with real-time video and the ability to send high-quality pictures.

3.2.6 URGENT HEALTH CARE

- This use case is envisioned around applications involving remote diagnosis and treatment. There is a need for remote patient monitoring and communications with devices measuring vital signs such as ECG, pulse, blood glucose, blood pressure, and temperature. The remote treatment and response based on monitored data can be life critical for a patient, requiring an immediate, automatic, or semi-automatic response.
- The URLLC features are used for two aspects they are,
  - Remote surgical consultations
  - Remote surgery
- Remote surgery is about applications in a mobile scenario in ambulances, disaster situations and remote areas requiring precise control and feedback communication mechanisms for surgeons in terms of low latency, high reliability, and tight security.
- In a remote surgery scenario, the entire treatment procedure of patients is executed by a surgeon at a remote site, where hands are replaced by robotic arms. In these two cases, the communication networks should be able to support the timely and reliable delivery of audio and video streaming.

3.2.7 INTELLIGENT TRANSPORTATION

- The realization of URLLC can empower several technological transformations in the transportation industry, including automated driving, road safety, and traffic efficiency services. These transformations will get cars fully connected such that they can react to increasingly complex road situations by cooperating with others rather than relying on their local information. These trends will require information to be disseminated among vehicles reliably within an extremely short time duration.

3.3 SLICING

- Below are the common use cases of the 5G network slicing, which serves almost all the common applications of the 5G network spectrum.

3.3.1 PERFORMANCE

- 5G network slicing supports enhanced mobile broadband (eMBB), which aims at maximizing the network speeds and data rates while having an acceptable QoS, including reliability and packet-error rates.
- 5G network slicing can also enable network providers to serve various clients by providing a tailor-made network for maximum performance. For instance, a 5G network provider can use network slicing techniques to provide a minimal latency slice to time-sensitive applications such as autonomous vehicle management, which typically require latency under 5 milliseconds.

3.3.2 CAPACITY

- Several other applications, such as surveillance and security systems, need very high network bandwidths, capacity, reliability, and quality of service support. The throughput from surveillance cameras can constantly be over 1.5 megabytes per second without any packet drops. Network providers can aggregate multiple processing devices and connectivity to meet the needs of these types of use cases through slicing.
- Industrial applications and IoT can require massive machine-to-machine connections that need multiple connections with varying latency and bandwidth. All these growing needs can be addressed using network slicing as it allows both upscaling and downscaling of the network resources on the 5G network spectrum.

3.3.3 SECURITY AND IDENTITY MANAGEMENT

- Cybersecurity is now an absolute priority for all organizations and the 5G network has been built with security as a priority. Therefore, all firms require a highly secure and reliable network spectrum to minimize the risks associated with network security breaches such as data sniffing, ransomware, and data leaks, among others.
- 5G network slicing can also help vertical industries provide various levels of security and identity management. It can also cater to the growing device-user identity management and

related lifecycle management.

4. 5G Super Blueprint Reference Architecture Lab Architecture

- The below architecture is designed in the Kaloom lab that overviews the proof of concept system of the 5G SBP. The main elements of the network are 5G RAN, CORE, UPF, and IOT App Server. Free 5G Core was used as an open source 5G core with Kaloom virtual UPF while GenXComm's GXC 5G RAN Radio Unit (RU), Distributed Unit(DU), and Central UNit (CU) provided radio access network capabilities.
- For IOT application IBM IOT Device (camera) and IOT application software to manage edge devices and learning engine were used. Here the **IBM IOT Dev** is located in Kaloom's lab in Montreal and **IBM IOT APP** is located in the IBMs lab in Dallas.

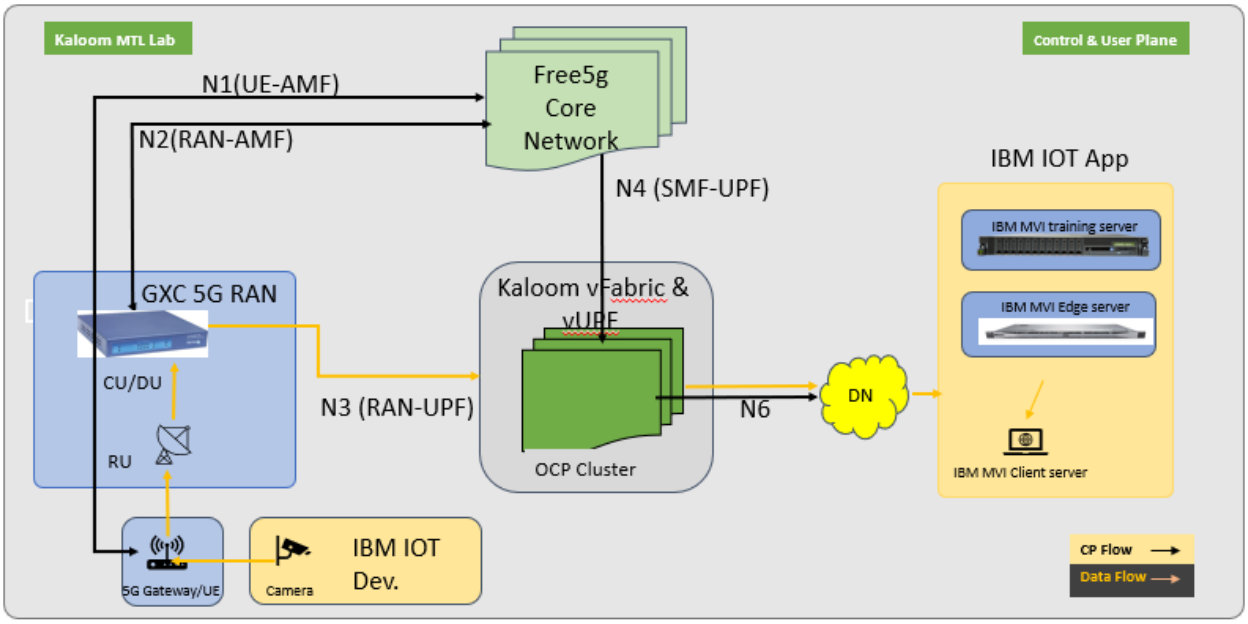

Figure 2: Proof of Concept Lab Architecture

5. Build and Integration Components

5.1 5G Core

- Free 5GC
- Kaloom vUPF

5.2 5G RAN using GXC RAN Overview

- The Role of GXC RAN in the Reference Architecture is to provide radio access network in 5G system, with the following functionality.
  - Radio Unit (RU)- 5G NR ORAN compatible n78 radio unit is capable of 100MHz of bandwidth
    - Indoor Radio 5G NR SA
    - 100MHz of Bandwidth
    - 24 dBm, 4T4R
    - 1.6Gbps throughput support
    - IEEE1588v2 support
- Central Unit (CU) + Distributed Unit (DU) is layer 2, 3 is ORAN compatible software for 5G system. Key features include support for 3GPP Release 16 features.

5.3 IBM-Maximo Visual Inspection (MVI) IOT Application

- IOT applications and use cases split data collection and data processing. IOT devices such as motion or heat sensors, IP cameras collect and provide data while Application servers in or near edge compute facilities process the data and provide actionable insight for humans or other upstream/downstream processes.
- In this proof of concept, IBM demonstrated proof of concept using IOT camera and IBM IOT edge server and learning engine using 5G Network. Traffic from IOT devices was routed to via Kaloom vUPF to local edge compute nodes where IBM Edge Server and Model Training with local loop option implemented over Kaloom vUPF.
- The main objective of IBM's part in this Reference Architecture is to show how IOT devices can be used to improve safety at factory floor or improve product quality or process. This work was demonstrated in a factory, when employees enter a designated area, they must be wearing proper Personal Protective Equipment (PPE) such as a hard hat, protective face mask, etc., A solution is needed to monitor the designated area and issue an alert only when an employee has been detected, entering the area without wearing a hard-hat / face mask, else no alert is issued. We achieve this using IBM's Maximo Visual Inspection (MVI) an AI-based video analytics solution.

6. Integration Components (Steps for Setting Up the 5G Super Blueprint)

- In this section a detail of hardware and software used for this demonstration is provided.

6.1 Demonstration HW/SW Summary Bill of Materials

Requirement	
Components	**Version**
Openshift	4.10
5G Core	Free5GC 3.2.1 on Ubuntu 20.04
Kaloom Software	KSDF 2.5 RC2
MVI training Server HW spec: (2 GPUs)	Dell PowerEdge R650 56Core, 512GB RAM, 2x960GB SSD, 2 Intel(R) Xeon(R) Gold 6348 CPU @ 2.60GHz, 2x Nvidia Tesla T4 GPU
MVI Edge server HW spec: (1 GPU)	Dell PowerEdge R650 64Core, 512GB RAM, 2x960GB SSD, 2 Intel(R) Xeon(R) Gold 6338N CPU @ 2.20GHz Processor, 1 Nvidia Tesla T4 GPU
OS software for all participating nodes in the Kaloom Fabric	
RHEL	RHEL 8.6

6.2 5G RAN System

- Below is the block diagram of the RAN system used in the 5G SuperBluePrint Reference Architecture 2022.
- There are two end user devices: Samsung Phone and a 5G UE are connected over the air to the 4x4 Radio Unit.
- The 5G UE transports the data from the camera over the 5G airwaves to the Radio Unit.
- The Radio Unit is connected to the gNB (DU + CU) for the data, and with the clock source for the synchronization.

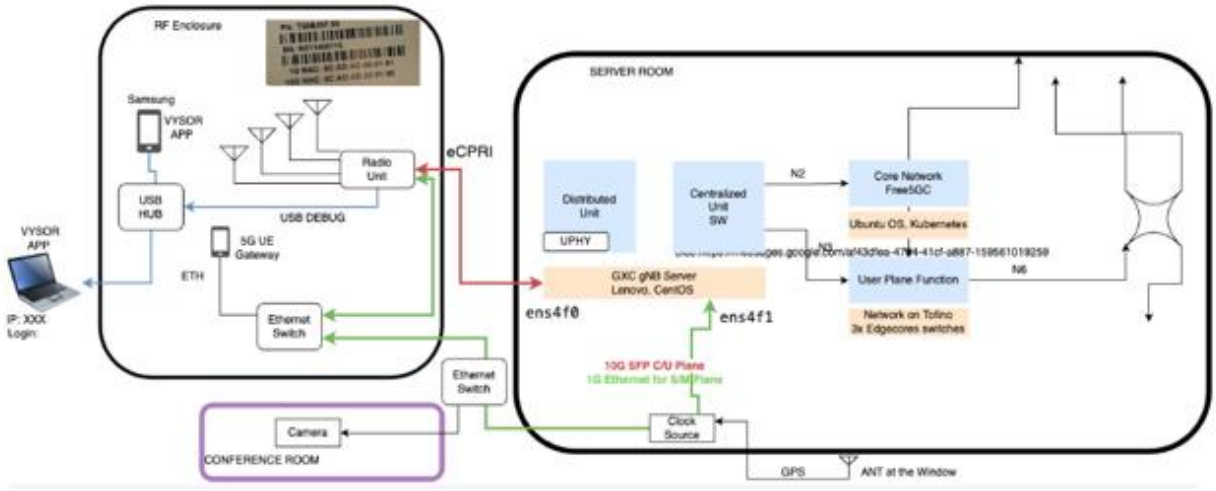

Figure 3: RAN System

6.2.1 LIST OF COMPONENTS FOR RAN

- Below are the list of RAN components.

Item	Type	Description
1	Hardware	RU (5G Stand Alone n78)
2	Hardware	RU Antenna (4x4)
3	Hardware	RU Power Supplies
4	Hardware	RU Power Cables
5	Software	FlexRAN - v21.11
6	Software	CU/DU L2/L3 - vGXC5G3.1
7	Hardware	DU Hardware Accelerator
8	Hardware	DU HW

Table 2: List of RAN components

- Contact information for procurement of GXC RAN
  - Technical point of contact: Hardik Jain : hardik@gxc.io
  - Sales point of contact: Jeff Schranz: sales@gxc.io

6.3 MONTREAL LAB AT KALOOM

- Kaloom has configured the part both from the Switching and the UPF perspective with the payload and given to Aarna networks, where they integrated it with their solution.

- The below diagram shows the end-to-end connectivity for the full network path provided by Kaloom integrating with Aarna Networks on the Cloud Native 5G Super Blueprint proof of concept.

Note: *Kaloom provides the environment and OCP Cluster to Aarna Network.*

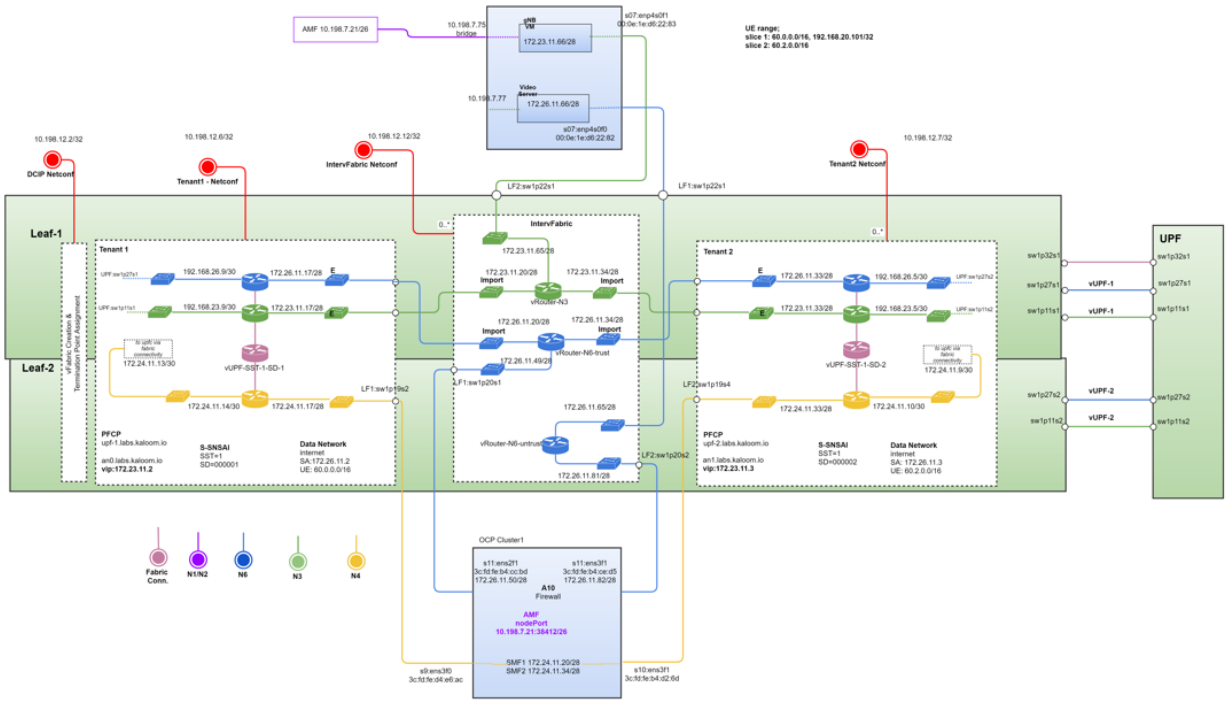

Figure 4: End-to End Connectivity of full Network path

- All in green in the above diagram are the physical nodes that are available in the Kaloom fabric and it has an UPF & two switches.

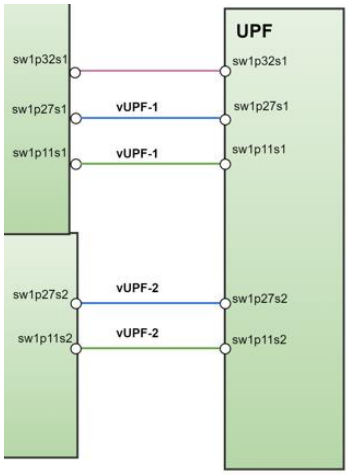

Figure 5: UPF & Switches

- Tenant 1, IntervFabric, and Tenant 2 are the virtual V-Fabric that is created.
- IntervFabric is used to allocate all the traffic between the two different UPF and each UPF instance can share it.

- Aarna Networks and Nextgen Firewall runs on the OCP cluster.

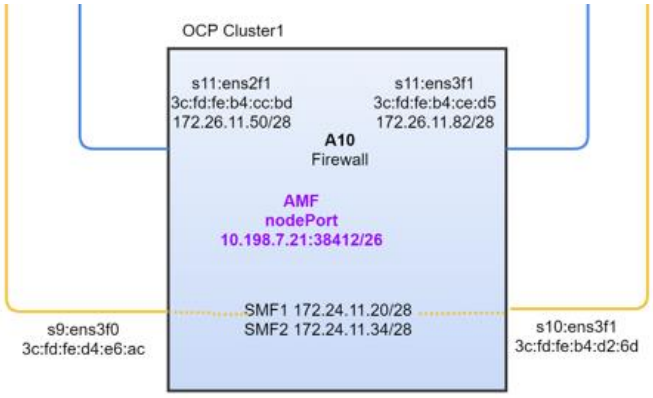

Figure 6: OCP Cluster

- Information is pushed by the firewall (Yellow line in the OCP Cluster) through OCP Cluster, but for the N6 traffic it was first untrusted and trusted as an internal network. Only one N6 at the firewall is sent as untrusted to the nodes.

VCO-Fabric-LFN.out

- Below are the slices that are used,
  - vco-lfn-intervfabric

VCO-LFN-Intervfabric.out

  - vco-lfn-slice1

VCO-LFN-Slice1.out

  - vco-lfn-slice2

VCO-LFN-Slice2.out

6.3.1 VFABRIC-CREATION OF A VUPF

- The configuration for the UPF on the vFabric is split between internal and external environments.
- Below attached word file shows how to get the Operational data of the vFabric.

Get-Operational Information.docx

Note: *Ensure the vRouter must exist on the vFabric before adding the vRouter in the Internal*

Environment.

6.3.2 VFABRIC CREATION OF A VUPF - INTERNAL ENVIRONMENT

- The configurations of the vFabric internal environment make use of the SPs of the UPF equipment. System Ports are ports connected between the UPF nodes and Leaf Nodes and will be used for internal UPF communication only.
- To configure the internal vFabric environment for the vUPF, follow the below flow diagram.

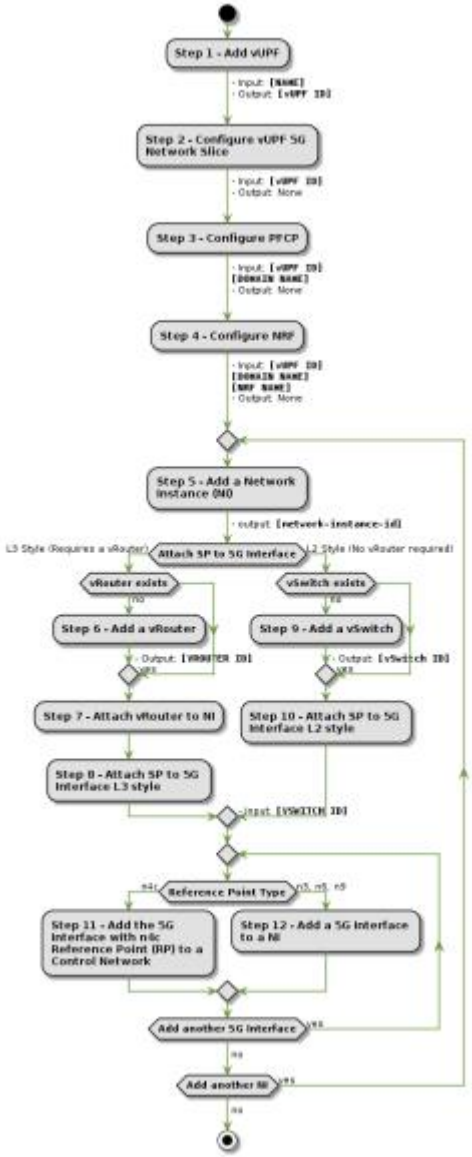

Figure 7: Configuring the Internal vFabric

6.3.3 CONFIGURE VFABRIC EXTERNAL ENVIRONMENT

- The configurations of the vFabric external environment make use of the TPs assigned to the vFabric and allow the internal network environment (vUPF configurations) to communicate with

everything outside the vFabric network.

- The vDCO must run the following commands to configure the vFabric's external network environment.

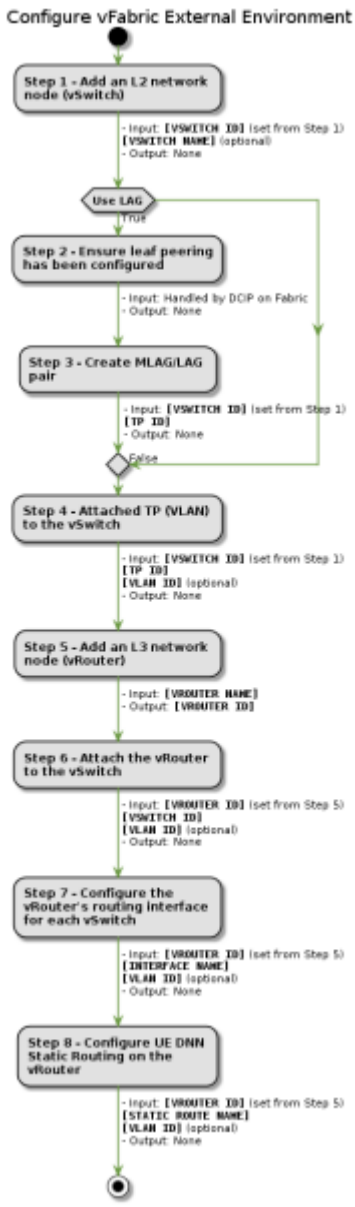

Figure 8: Configuring the External vFabric

6.3.4 SAMPLE CONFIGURATION OF A VFABRIC

- The following is a sample configuration where a vUPF is configured on a vFabric Slice.

Column 1	Column 1 Contd..
Sample Creation of a vUPF --- ldap: - name: ldap-server.kaloom.io	- name: L2:N3-to-Private5G mtu: 9000 ipv4: - address: 172.23.11.17

```
            host: 10.127.16.2
            protocol: ldap
            port: 389
            bind_dn:
           "cn=admin,dc=kaloom,dc=io"
            bind_password: "!Password123"
            base_dn:
           "o=vfabric,dc=kaloom,dc=io"
            filter:
           "(objectClass=organizationalPerson)"
lag:
   - lagName: LAG-N3
     tp1: L1-sw1p10s1-25G
     tp2: L2-sw1p10s1-25G
     lacpMode: active
     lacpShortTimeout: "true"
     lacpMinLink: 1
   - lagName: LAG-N4
     tp1: L1-sw1p10s2-25G
     tp2: L2-sw1p10s2-25G
     lacpMode: active
     lacpShortTimeout: "true"
     lacpMinLink: 1
   - lagName: LAG-N6
     tp1: L1-sw1p10s3-25G
     tp2: L2-sw1p10s3-25G
     lacpMode: active
     lacpShortTimeout: "true"
     lacpMinLink: 1
interfaces:
      - id: L2:N3-to-Private5G
        name: L2:N3-to-Private5G
        enabled: true
        description: external access to N3
        tp:
        - name: LAG-N3
          vlan: 3113
      - id: L2:N4-to-Private5G
        name: L2:N4-to-Private5G
        enabled: true
        description: external access to N4
        tp:
        - name: LAG-N4
          vlan: 3114
      - id: L2:N6-Private5G
        name: L2:N6-Private5G
        enabled: true
        description: external access to N6
        tp:
        - name: LAG-N6
 vlan: 3116
            prefix: 28
            ipv6:
            - address: fd23:12:10::1
            prefix: 120
      - name: L2:N6-Private5G
            mtu: 9000
            ipv4:
            - address: 172.26.11.17
            prefix: 28
            - address: 172.26.11.49
            prefix: 28
            ipv6:
            - address: fd26:12:10::1
                  prefix: 120
                  pfcp:
                   - node: upf.kaloom.io
                   ref: n4c
                   ignoreUnsupportedIe: "false"
                   ipv4: 172.24.11.12/30
                   ipv6: fd24:11:0::12/127
                   supportedFeatures:
                   - bucp: "false"
                   ddnd: "false"
                   dlbd: "false"
                   trst: "false"
                   ftup: "true"
                   pfdm: "false"
                   heeu: "false"
                   treu: "false"
                   empu: "false"
                   pdiu: "false"
                   udbc: "false"
                   quoa: "false"
                   trace: "false"
                   frrt: "false"
                   pfde: "false"
                   router:
                   - name: vupf-router
                  nrf:
               - nrf: nrf.kaloom.io
                   networks:
                   - node: dn1.labs.kaloom.io
                   default: "false"
                   description: N6Interface-DN1
                   router:
                   - name: vupf-router
                   servingIP:
                   - ref:
                   - n6
                   servingIP4: 172.26.11.2
                   servingIP6: fd26:11:0::2
```

<table>
<tr>
<td>vupfs:
- name: Private5G_vUPF
description: vUPF to slice UPF functionality
routing:
- name: vupf-router
enabled: "true"
lookingGlass: "false"
interfaces:
- name: L2:N4-to-Private5G
mtu: 9000
ipv4:
- address: 172.24.11.17
prefix: 28
ipv6:
- address: fd24:12:10::1
prefix: 120</td>
<td>name: N6Interface-DN1
systemPort:
- tp: UPF-sw1p5s1-100G
network: 192.168.26.6/30
name: N6Interface-DN1
- node: an0.labs.kaloom.io
default: "false"
description: NI Interface for N3
router:
- name: vupf-router
servingIP:
- ref:
- n3
servingIP4: 172.23.11.2
servingIP6: fd23:11:0::2
name: N3Interface-AN0
systemPort:
- tp: UPF-sw1p13s1-100G
network: 192.168.23.6/30
name: N3Interface-AN0</td>
</tr>
</table>

Table 3: Sample Configuration of a vFabric

- Below attached word files are examples of "Kaloom-Configure-vUPF" and "Kaloom-Create-vFabric".

Kaloom-ConfigurevUPF-example

Kaloom-Create-vFabric-example.doc

6.3.5 DCIP RESOURCE MANAGEMENT OF THE FABRIC

- Data Center Infrastructure Provider (DCIP) who owns and/or manages the Fabric equipment, such as Fabric switches/servers, also referred to as Kaloom system nodes and allocates Fabric resources to a Virtual Data Center Operator (vDCO) in the form of a Virtual Fabric (vFabric).
- It is achieved through the Fabric, which provides the means to connect all the user equipment, such as servers (E.g., Application or Storage) and networking equipment (E.g., Switches or Routers) together as part of one network, irrespective of the data center scale.
- The Fabric also provides the means to configure and monitor the condition and status of the Fabric equipment.
- The DCIP distributes the data center's hardware resources to one or many Virtual Data Center Operators (vDCOs) through a process called slicing.
- Slicing allows the DCIP to distribute through the Fabric and data center hardware resources to vDCOs by first creating Virtual Fabrics (vFabrics) and then allocating those resources to the vFabric

in the form of Termination Points (TPs).

- TPs are network port channels on Leaf switches connected to different user equipment such as servers.
- The same TP cannot be assigned to two different vFabrics and TPs of the same server should be allocated to the same vFabric. This provides the vDCO privacy, isolation, and protection not only from other vFabrics but also from the DCIP, where the TPs are already allocated to the vFabric, which does not know how those TPs are used or how the vFabric network is set up.
- Port channelization subdivides the bandwidth of an interface into multiple channels where each channel represents a Termination Point (TP) that is assigned to a vFabric.
- A Port cannot be channelized if already allocated to a vFabric Slice.

Note: *The Fabric must be configured to authenticate against an LDAP Server. It is assumed the LDAP server for the Fabric is configured during the deployment.*

6.3.6 FLOW DIAGRAM

- The below diagram shows the typical sequence of events to slice the fabric and allocate network port channels to existing slices.

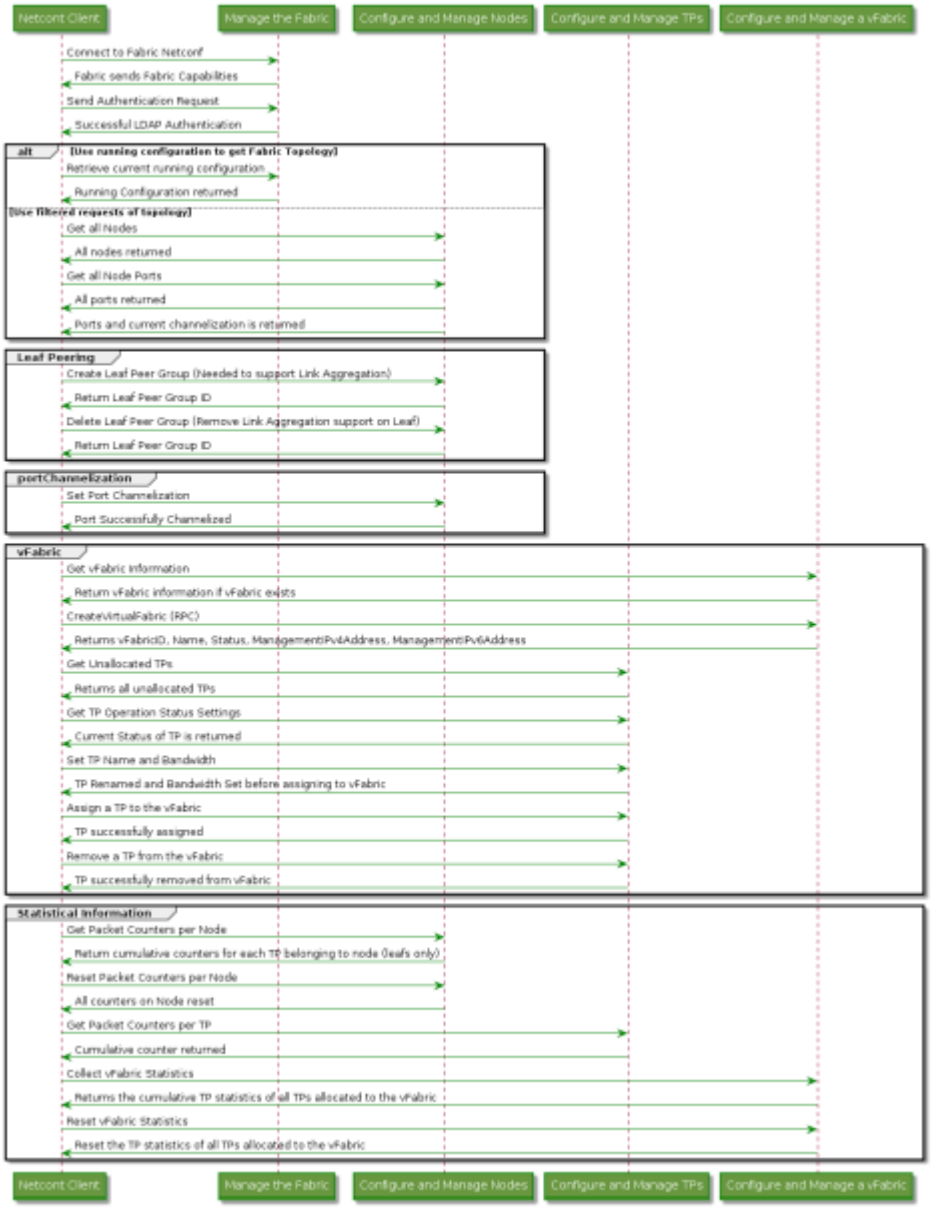

Figure 9: Slicing the fabric and allocate network port

6.3.7 SAMPLE DCIP CONFIGURATION

- Below the YAML file is an example of how the topology can be built to perform all the configurations of the Fabric and assign Termination Points to a vFabric called Private5G (i.e., NPN).

```
Fabric Resource Management
        ---
        fabric: 212
        LeafPeerGrp:
         - node1: Leaf-1
         node2: Leaf-2

    Leaf:
    - name: Leaf-1
```

```
      vfabric: Private5G
      switchport:
      - port: sw1p10s1
        name: L1-sw1p10s1-25G
        fec: FIRE-CODE
        enableFec: true
        bw: 25G
        enable: UP
      - port: sw1p10s2
        name: L1-sw1p10s2-25G
        fec: FIRE-CODE
        enableFec: true
        bw: 25G
        enable: UP
      - port: sw1p10s3
        name: L1-sw1p10s3-25G
        fec: FIRE-CODE
        enableFec: true
        bw: 25G
        enable: UP

  - name: Leaf-2
      vfabric: Private5G
      switchport:
      - port: sw1p10s1
        name: L2-sw1p10s1-25G
        fec: FIRE-CODE
        enableFec: true
        bw: 25G
        enable: UP
      - port: sw1p10s2
        name: L2-sw1p10s2-25G
        fec: FIRE-CODE
        enableFec: true
        bw: 25G
        enable: UP
      - port: sw1p10s3
        name: L2-sw1p10s3-25G
        fec: FIRE-CODE
        enableFec: true
        bw: 25G
        enable: UP
  - name: UPF
      portChannelization:
      - port: sw1p5
        ChannelBreakout: CHANNEL1_4
        MirrorConfigOnCabledPeer: true
      - port: sw1p13
        ChannelBreakout: CHANNEL1_4
        MirrorConfigOnCabledPeer: true
      vfabric: Private5G
```

```
switchport:
- port: sw1p5s1
   name: UPF-sw1p5s1-100G
   enable: UP
- port: sw1p13s1
   name: UPF-sw1p13s1-100G
   enable: UP
```

Table 4: Sample DCIP Configuration

6.4 IBM

- The roles of MVI Training Server and MVI Edge Server are given below,
  - Deep learning models must be trained to identify a person wearing a hard hat and face mask. This is accomplished using IBM's MVI Training Server.
  - The models need to be containerized and deployed to the edge cluster.
  - The 5G enabled smart camera communicates with the MVI edge server through the 5G network. All the inferencing happens on the MVI edge server.
- The below diagram briefs the IBM's MVI servers (Training Server and Edge Server)

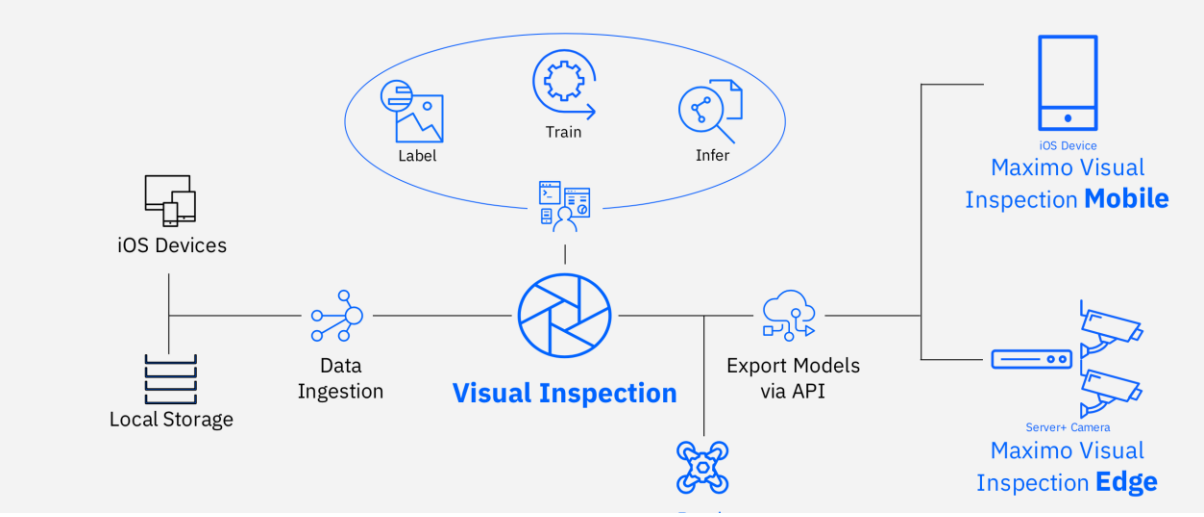

Figure 10: IBM's MVI servers

6.4.1 MVI TRAINING SERVER

- IBM's MVI makes computer vision with deep learning more accessible to business users. It includes an intuitive toolset that empowers subject matter experts to label, train, and deploy deep learning vision models, without coding or deep learning expertise.
- It includes the most popular deep learning frameworks and their dependencies, and it is built for easy and rapid deployment and increased team productivity. The above diagram gives an overall view of MVI components.

6.4.2 MVI EDGE SERVER

- The MVI edge server is an inference engine to seamlessly publishes AI models. This offering is provided as a part of the MVI offering and can be installed on inference servers. It helps customers distribute computer technology to their production lines, plants, and enterprises.
- MVI is delivered as part of the Maximo Application Suite with two key components.
  - **Visual Inspection Training** (aka MVI Training Server): Labeling, Model Training, Validation, and Dataset Management.
  - **Visual Inspection Edge** (aka MVI Edge Server): Inspection Definition, Processing Pipeline, and Line Integration Point.

6.4.3 INTEGRATING MVI TRAINING SERVER WITH MVI EDGE SERVER

- By integrating MVI Edge with Maximo Visual Inspection, users can collect or inspect images from input sources, such as fixed cameras, that are positioned at the edge of manufacturing activity.
- In MVI Edge, user collect or inspect images by creating inspections. Inspections are the central configuration components in MVI Edge.

6.4.4 MVI TRAINING SERVER WORKFLOW

- IBM MVI is a video and image analysis platform that makes it easy for subject matter experts to train and deploy image classification and object detection models. Below are the steps to build a hardhat detection model using MVI.

Step.1 Create a set of videos with individuals wearing a hardhat. Make sure to include varied scenarios with different lighting conditions.

Step.2 Create Datasets and import the images or videos that are created in step 1.

Step.3 Create two object detection models, one for hardhat and another for the facemask.

Step.4 Label the images based on the model that is created.

Step.5 Once the labeling is done, train the model. The training time depends on the size of the data, type of model, and additional options selected.

Step.6 Once the model is trained, deploy the model either locally or on the Edge server.

Step.7 The deployed hardhat model now appears in the Deployed Models tab where the model can be tested either by using the API endpoint displayed or by clicking the Open button and uploading a video to test if the hardhats are being detected.

Step.8 The deployed model (both hardhat and facemask) before deploying it into the production environment, can be tested either using API through GUI or detect if the person wears a hardhat/facemask.

6.4.5 DEPLOYING THE TRAINED MODELS IN THE MVI EDGE SERVER

- IBM MVI edge server is a server that lets you quickly and easily deploy multiple trained models. In MVI Edge, users can download deployed models and deploy them locally. Inspections can use either remotely or locally deployed models. Below are the steps to achieve this.

Step.1 Install MVI Edge on the edge server with the necessary GPUs.

Step.2 Get the necessary Models from the MVI Training Server, the connectivity is achieved through MVI API keys (that are taken from MVI Training Server).

Step.3 Choose the hardhat and facemask models that are already trained from the MVI Training server that is ready for inference.

Step.4 Deploy those models on the MVI Edge server.

Step.5 Choose the input source for the MVI source, for this proof of concept the 5G enabled camera is chosen. The connectivity between the camera and the MVI edge server is through 5G networks as shown in Figure:1, camera is attached to the 5G gateway through which it communicates to the MVI edge server.

Step.6 All set to triggered the inspection.

Step.7 Once trigged, the MVI Edge captures and installs pictures for inference.

Step.8 Based on the inference an alert/message is received stating whether the person who entered the premise wears hardhat/facemask or not.

6.4.6 IBM EDGE APPLICATION MANAGER (IEAM)

- IEAM (An IBM product) provides orchestration and management of individual Edge Nodes and Visual Inspection Edge deployments. But its core components heavily use the Open Horizon - EdgeX project open source project(Below figure).

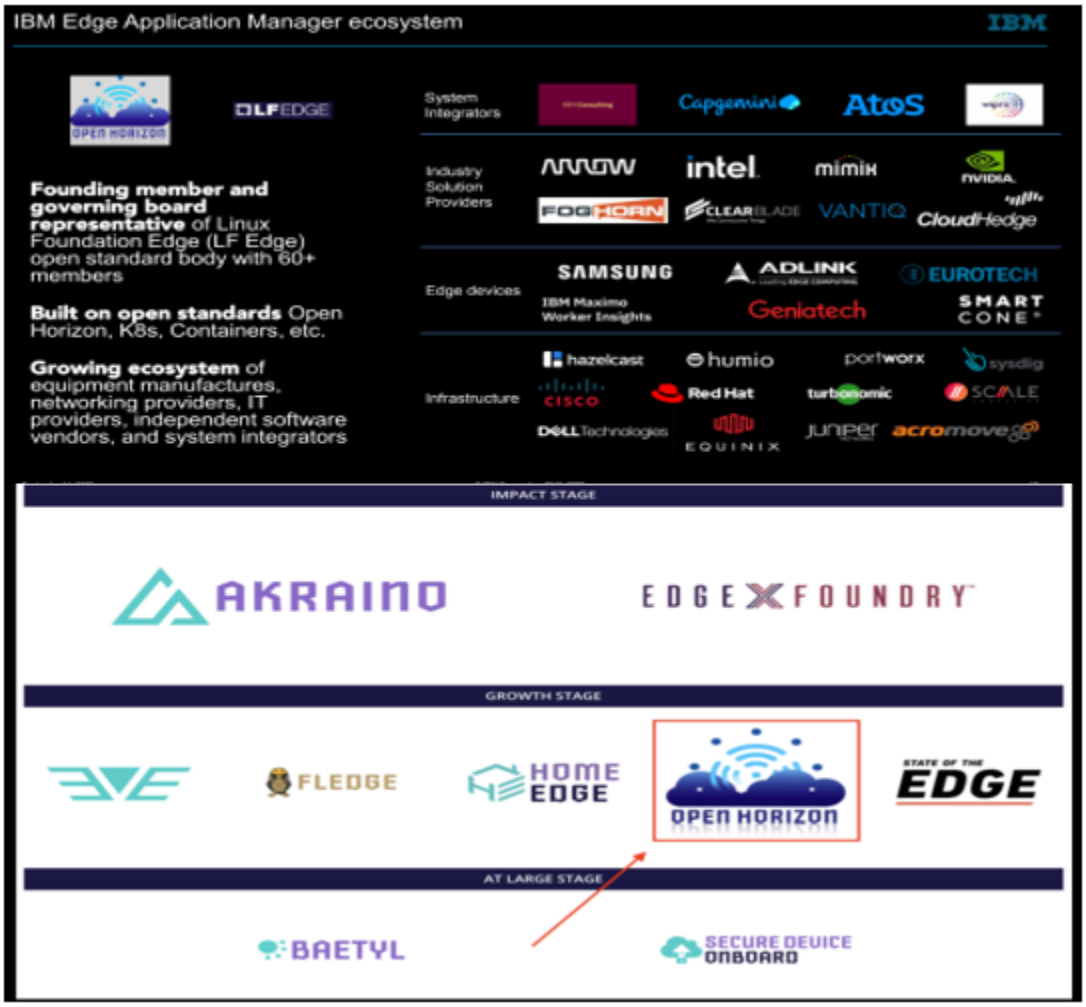

Figure 11: IBM Edge Application Manager Ecosystem

- Many sample and example programs that are available in the Horizon project work with IEAM. For more information about the project, see Open Horizon - EdgeX project. The below figure shows an overview of various components involved in IEAM and how they are interconnected.

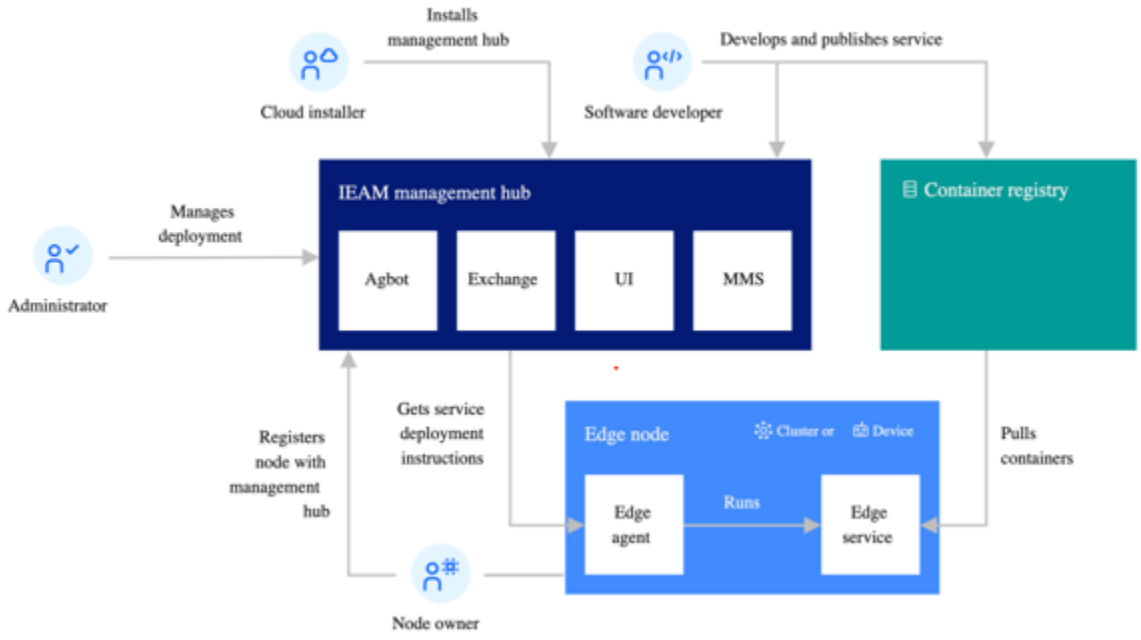

Figure 12: Components involved in IEAM

6.4.7 IEAM - OPEN HORIZON KEY CHARACTERISTICS

- Register an EDGE Server as a NODE and manage it centrally from a point of view of entitled applications that can be executed.
- Connect a container registry as the repository of the applications image available for the entitled nodes.
- Define deployment policies for the node to manage container services and also data synchronization services between the edge and central hub.
- Help to define a zerotouch strategy, the most high-value activities can be performed from the central hub.

6.4.8 IBM'S IEAM ROLE TO MANAGE MVI AT THE EDGE

- IEAM plays a key role in managing the MVI applications (Training server and Edge Server) located in different locations seamlessly. The e2e connectivity which includes IEAM components is depicted in the below figure.

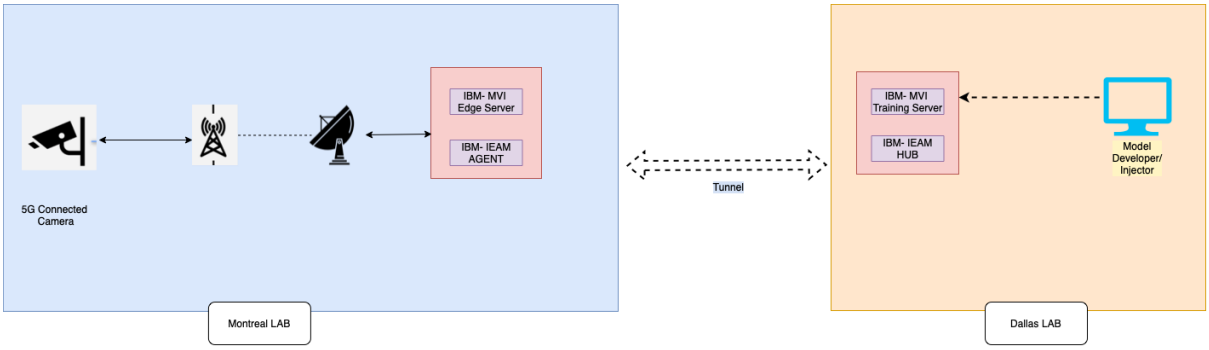

Figure 13: IEAM role to manage MVI at the Edge

- IEAM workflow to deploy the MVI model to the edge is given below.

  Step.1 Create an MVI model and containerize it.

  Step.2 Install the GPU operator on the edge cluster.

  Step.3 Register the edge device and edge cluster as edge nodes to IEAM Hub.

  Step.4 Create helm chart and helm operator service on edge device for the MVI model.

  Step.5 Publish the helm operator service to IEAM Hub.

  Step.6 Create node policy and deploy the edge operator service to the cluster.

6.4.9 ONBOARD EDGE COMPUTING DEVICES WITH SECURE DEVICE ONBOARDING (SDO) AND OPEN-HORIZON(IBM'S IEAM)

- IBM announced that Intel's Secure Device Onboarding (SDO) solution is now fully integrated into Open Horizon and IBM Edge Application Manager and available to developers as a tech preview.
- Both Open Horizon and SDO recently joined the LF Edge umbrella, which aims to establish an open, interoperable framework for edge computing independent of hardware, silicon, cloud, or operating system.

6.4.9.1 How SDO and IEAM (Open-Horizon) works together

- SDO software is installed at the manufacturer.
- An ownership voucher is created and always goes along with the device. I.e., Stays with the device throughout the supply chain.
- The end customer receives the devices and the vouchers and loads the vouchers into the OpenHorizon management hub.
- OpenHorizon uses this voucher to tell the SDO Rendezvous server how to install the OpenHorizon Agent on the corresponding device and configure it for this Management Hub.
- From this point, it's fully automated.
- Now the customers can start connecting the devices.
- On the first power-up, each device reaches out to the "Well-Known" address to the SDO Rendezvous

server, with an identifier corresponding to its specific ownership voucher.

- The SDO Rendezvous server responds with an Open-Horizon Agent install script and configuration details for this specific device.
- From this point, the SDO Rendezvous server is not required anymore.
- The device then automatically installs and configures the Agent for the appropriate Open-Horizon Management Hub.
- This Configuration includes the appropriate Open-Horizon device credentials and deployment pattern or policy.
- The Agent will then reach out to the Management hub and negotiate the appropriate software deployment using normal Open-Horizon procedures.

7. Roadmap

7.1 Aspirational Distributed Lab Architecture

- A distributed lab architecture is envisioned for future 5G Super Blueprint Architectures. Below is an aspirational depiction of a possible architecture.

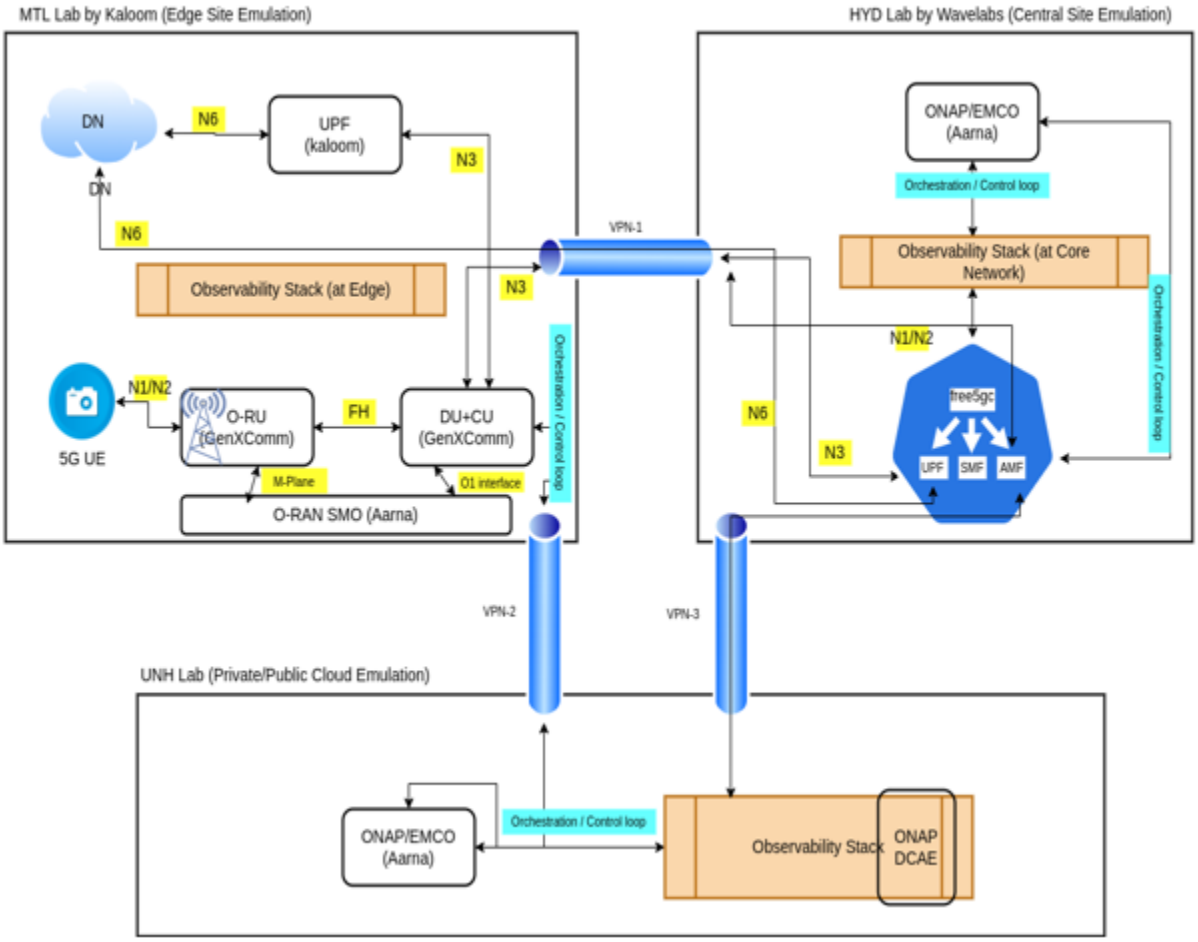

Figure 14: Aspirational Distributed Lab Architecture