bubble RAN

WELCOME TO THE WEBINAR

Accelerate network automation with BubbleRAN MX-PDK

Discover & learn with concrete use cases & hands-on examples optimizing on xApps, rApps, and network slicing.

Carlos Risma
Solution Architect

Alessandro Pacini
Telecom Cloud Engineer

Ali Yaghoubian
PhD Student Eurecom

Practical information

1. **Chat is for questions, our team is available to reply**
2. **Dedicated Q&A in the end**
3. **This webinar will be recorded**
4. **Please fill in a short questionnaire, link is in the chat**

BubbleRAN

AUTOMATION

INTELLIGENCE

OPEN ECOSYSTEM

- Breakthrough technologies in the area of **5G/6G, O-RAN and AI-RAN**
- Accelerate innovation by making **advanced telecom infrastructure** and software accessible, flexible, and feature-rich
- Enable building **scalable, extendable and customizable** 5G Open RAN and 6G ready networking solutions built on both open-source and industrial grade software stacks
- Create an **open ecosystem** for 5G open RAN including a developer portal with SDKs, APIs, and documentations


bubble
RAN

BubbleRAN - Products

MX-PDK

For R&D and PoC use-cases

5G O-RAN platform at Scale
Multi-Vendor 3GPP & O-RAN Compliant
Cloud-Native Automation
Data-lake and Observability
Slicing, and AI Frameworks

MX-AI

For AI powered R&D and PoC use-cases

5G AI-RAN platform
Multi-agent AIFabic and AI Agents inventory
Agentic toolkits and blueprints
Open Ecosystem with xApps/rApps/Agents/Datasets

MX-PRO

For Private 5G use-cases

Industry-grade private 5G
3GPP Compliant
Large set of add-on options
Use-cases: Campus, Maritime, Airport, Harbour, Factories, and Warehouses

MX-ORS

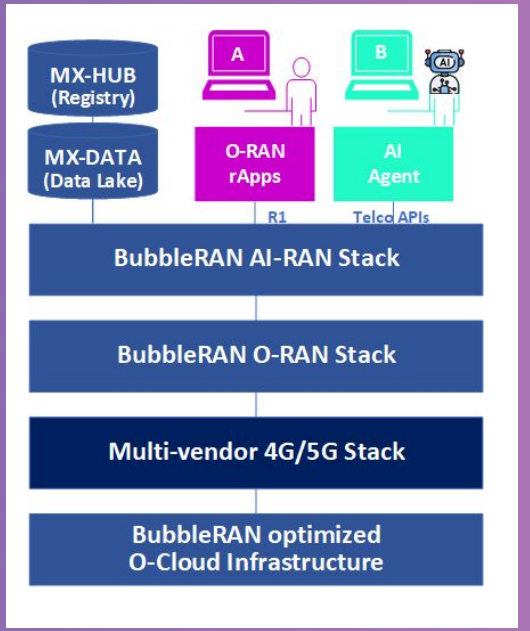
For Learning & upskilling use-cases

5G O-RAN Emulation Platform
Edge Intelligence
Multi-source Observability Stack
Hyper-Scale Hybrid Emulated Networks

BubbleRAN - MX-PDK

SOFTWARE STACK

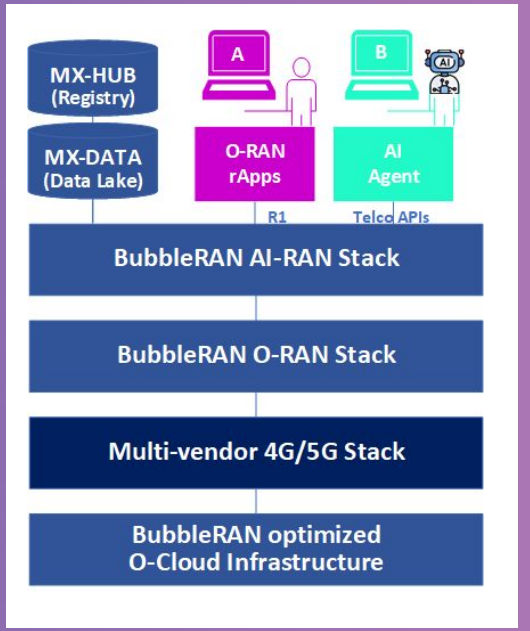
- Kubernetes-based O-Cloud infrastructure
- Industrial-grade & Open Source 5G Stack
- O-RAN compliant environment with SMO/RIC/xApps/rApps
- AI-RAN Powered Multi-agent Network Optimizer
- Edge Applications and Services
- Open Ecosystem of xApps/rApps/Agents



BubbleRAN - MX-PDK

ARCHITECTURE & FEATURES

- Deployment on control-plane nodes and additional infrastructure components
- On top of the control-plane nodes, BubbleRAN SMO enabling zero-touch lifecycle automation
- Policy-driven long term control facilitated through rApps leveraging the Non-RT RIC
- Data Lake & Observability Stack
- Heterogeneous set of gNB CU and DU

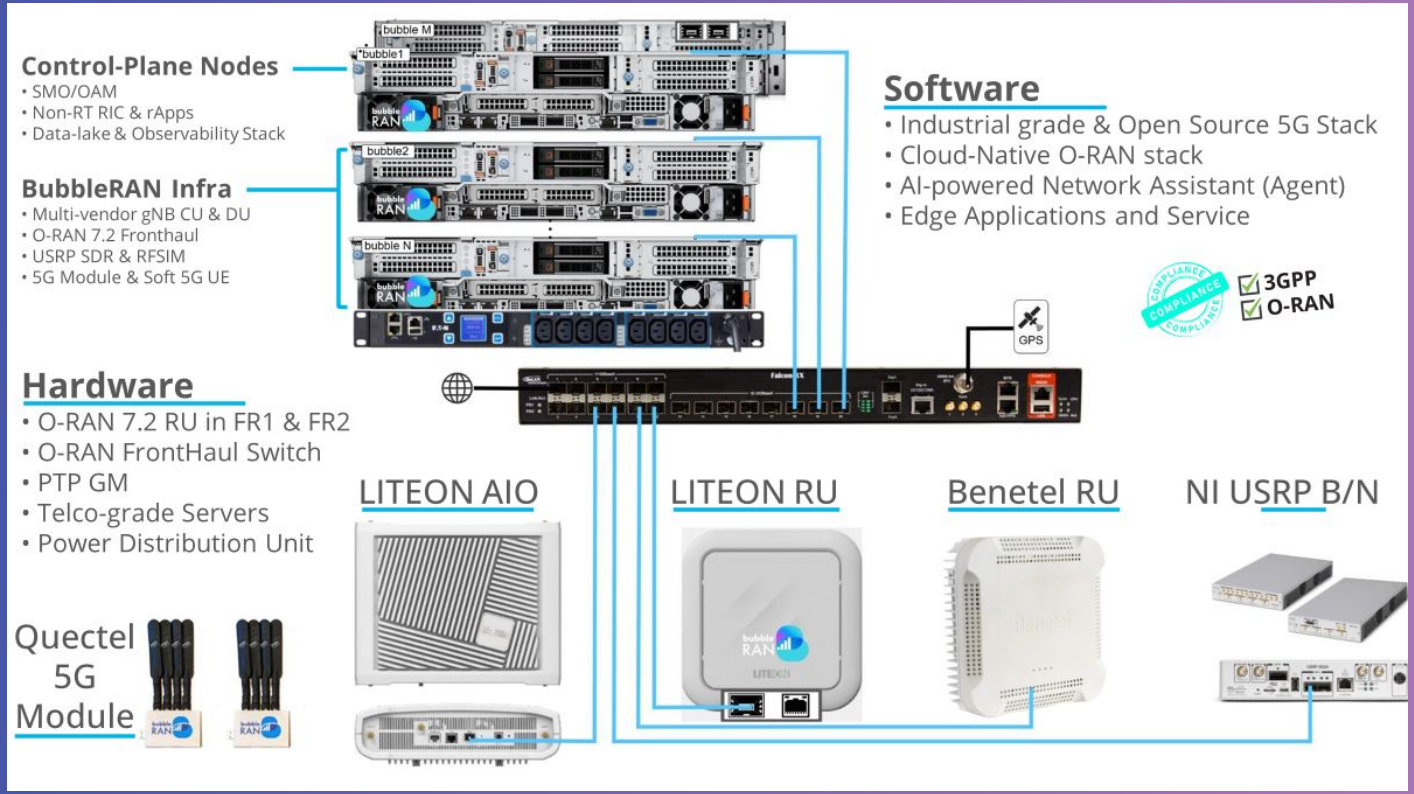


BubbleRAN - MX-PDK

HARDWARE EQUIPMENT

- O-RAN 7.2 RU | FR1 & FR2 | Indoor and outdoor
- O-RAN 7.2 FrontHaul Switch
- PTP GM
- Telco-grade Servers
- Power Distribution Unit
- Quectel’s 5G modules for UE integration
- SDRs such as Amarisoft and USRP

BubbleRAN - MX-PDK



Let’s talk about you…

And you want:

- Learn concrete use cases for network automation optimization
- Discover how you can efficiently use our SDKs in your work and projects
- Follow step-by-step guidance to simplify your own developments
- Fuel your interest in future networks and exciting topics of xApps, rApps and Network Slicing

You are in the right place!

Agenda of the webinar

4:10PM **Network Automation by Carlos Risma**

- Intent-driven cloud-native end-to-end network automation

4:25PM **xApp Automation by Alessandro Pacini**

- Network Observability by visualizing, monitoring, and analyzing metrics

4:40PM **rApp Automation by Ali Yaghoubian**

- Enabling Non-Real-Time Control and Management with rApps

4:55PM **Slice Automation by Carlos Risma**

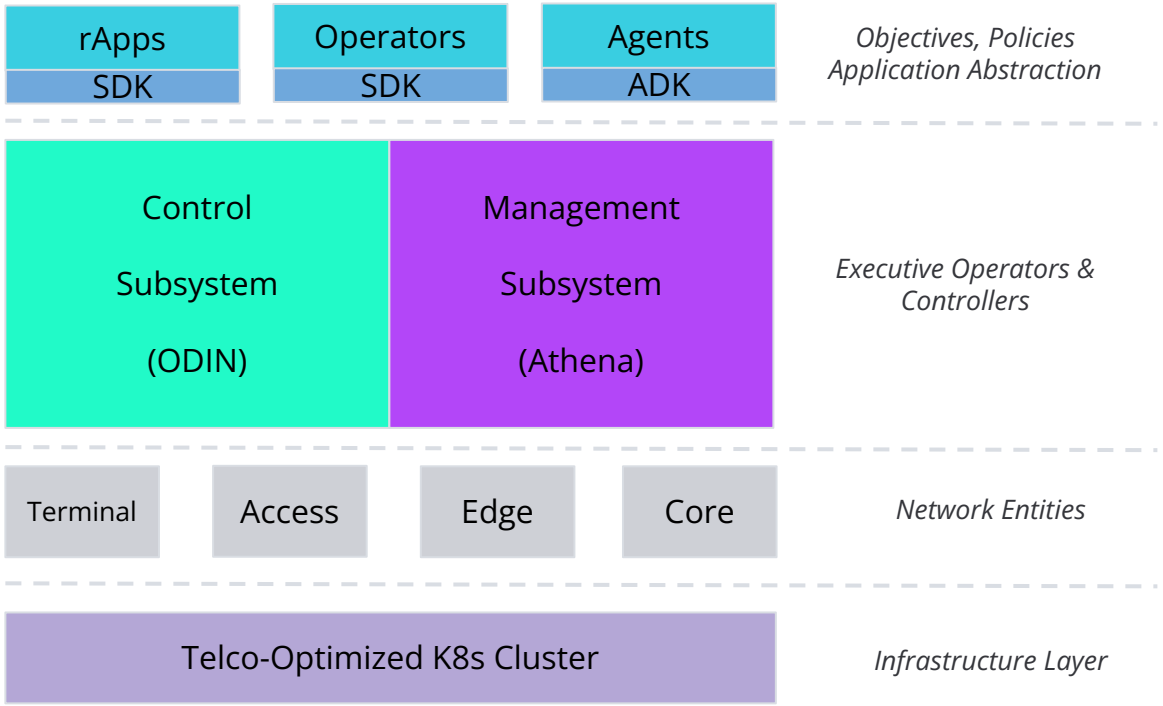
- Defining end-to-end 3GPP & O-RAN network slices through Service Profiles

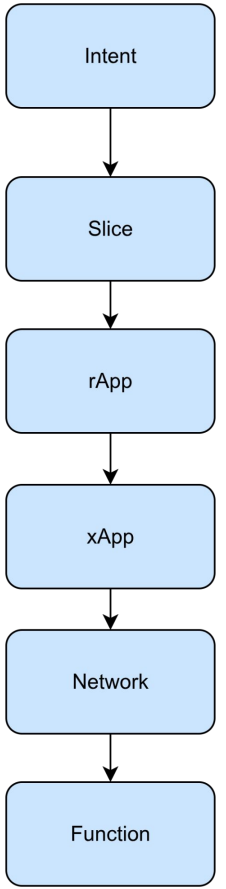
5:10 **Dedicated Q&A moderated by Navid NIKAEIN**

01

Network Automation

BubbleRAN's Operator Plane



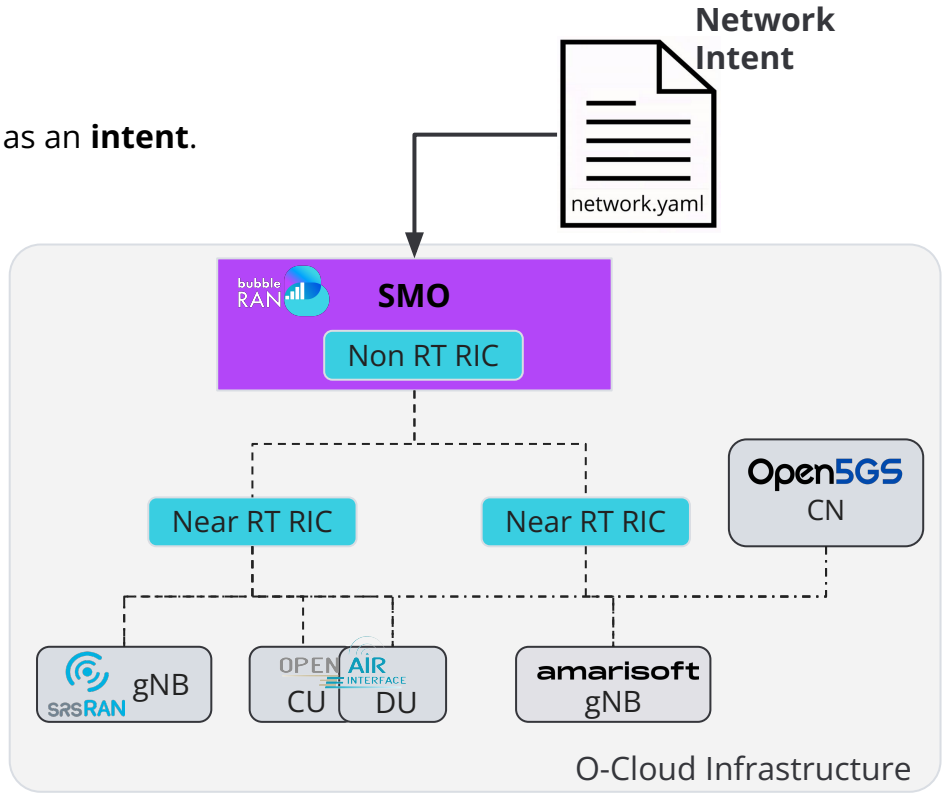


Zero-touch Network Automation

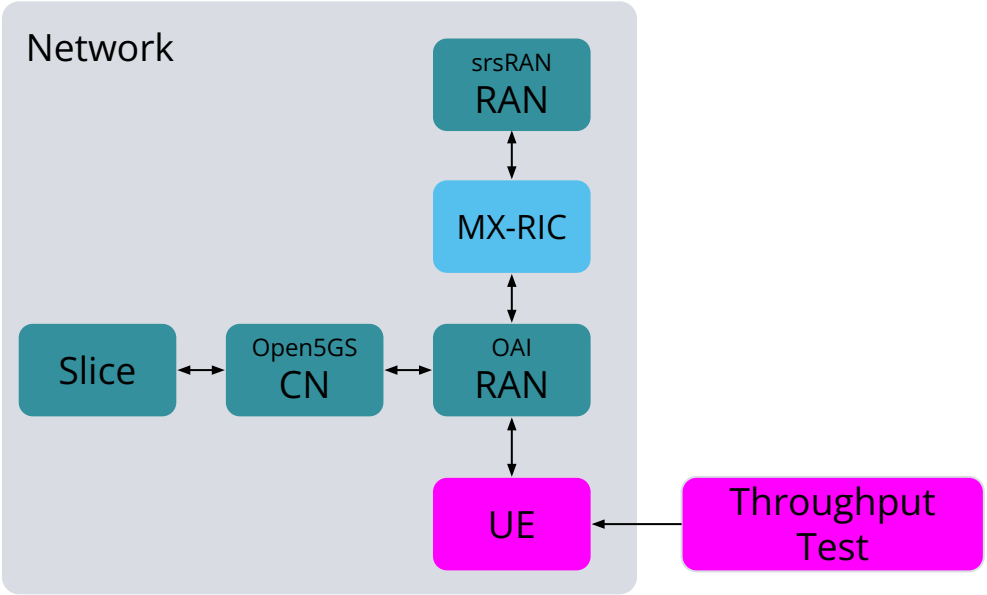
The **desired state** of your network is defined in a file as an **intent**.

Features:

- **Extendable**
  - Building blocks for each 5G domain (CN, RAN and Terminal) → **Modular design**
- **Multi-vendor**
- **O-RAN aligned SMO** orchestrates RAN and Core automation
- **Intent-driven**
  - Files define **what you want** as opposed to how to create a network → **Declarative**



Zero-Touch 5G SA



- Design a simple network with **O-RAN stack**
- Multi Vendor
  - **Open5GS & OAI & srsRAN**
- Day-2 operations
- UE in the loop

Zero-Touch 5G SA

Network

Network Intent

```yaml
apiVersion: athena.trirematics.io/v1
kind: Network
metadata:
    name: bubbleran
    namespace: trirematics
spec:
    slices:
    core:
    edge:
    access:
    dns:
```

Zero-Touch 5G SA

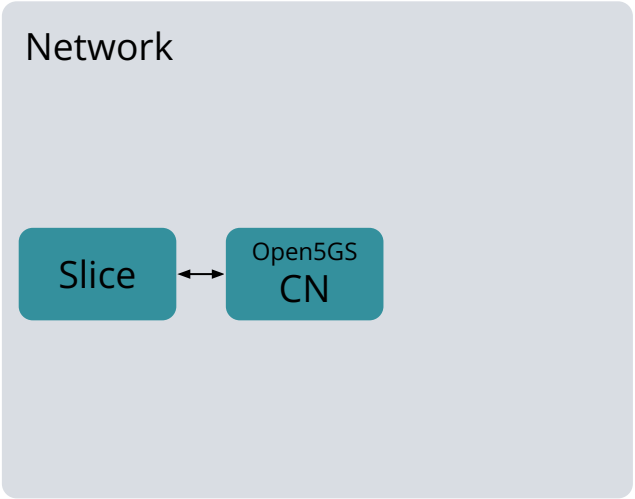
Network

Slice

Network Intent

```
slices:
    -   plmn: "00101"
        dnn: "internet"
        network-mode: "IPv4"
        service-type: eMBB
        differentiator: 0x000000
        ipv4-range: "12.1.1.0/24"
        ipv6-range: "2001:db8:1::/64"
```

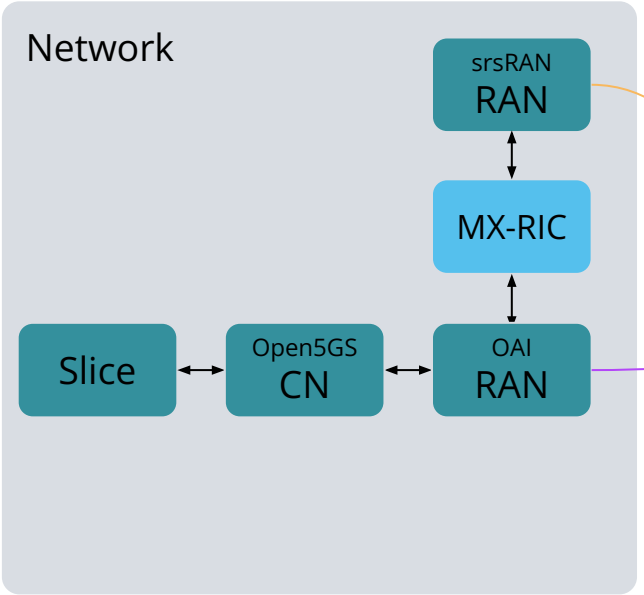
Zero-Touch 5G SA



Network Intent

```
core:
    -   name: ogs
        stack: 5g-sa
        model: open5gs/5gc
        identity:
            region: 0
            cn-group: 4
            cn-id: 5
```

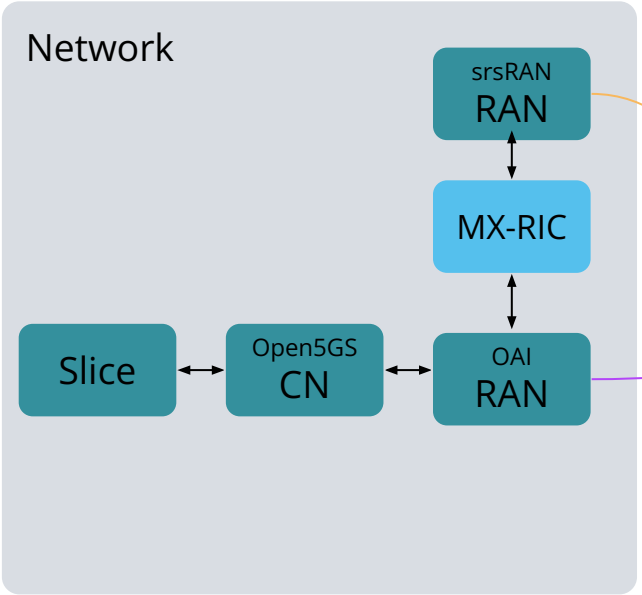
Zero-Touch 5G SA



Network Intent

```
edge:
    -   name: ric
        stack: 5g-sa
        model: mosaic5g/flexric
access:
    -   name: oai-gnb
        stack: 5g-sa
        model: oai-ran/monolithic-gnb
        identity:
            an-id: 50
        radio:
            device: rf-sim
        cells:
            -   band: n78
                arfcn: 641280
                bandwidth: 40MHz
                subcarrier-spacing: 30kHz
                tdd-config:
                ...
        controller: ric.bubbleran
        core-networks:
            - ogs.bubbleran
    -   name: srs-gnb
        stack: 5g-sa
        model: srs/monolithic-gnb
        ...
        controller: ric.bubbleran
        core-networks:
            - ogs.bubbleran
```

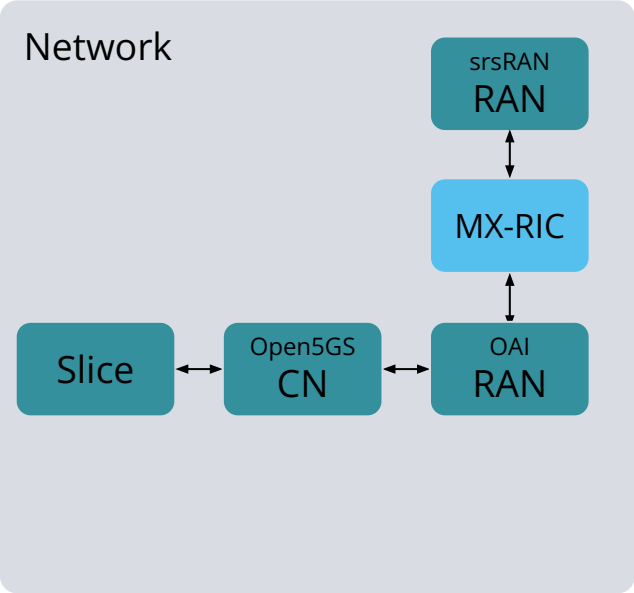
Zero-Touch 5G SA



Network Intent

```
edge:
    -   name: ric
        stack: 5g-sa
        model: mosaic5g/flexric
access:
    -   name: oai-gnb
        stack: 5g-sa
        model: oai-ran/monolithic-gnb
        identity:
            an-id: 50
        radio:
            device: rf-sim
        cells:
            -   band: n78
                arfcn: 641280
                bandwidth: 40MHz
                subcarrier-spacing: 30kHz
                tdd-config:
                ...
        controller: ric.bubbleran
        core-networks:
            - ogs.bubbleran
    -   name: srs-gnb
        stack: 5g-sa
        model: srs/monolithic-gnb
        ...
        controller: ric.bubbleran
        core-networks:
            - ogs.bubbleran
```

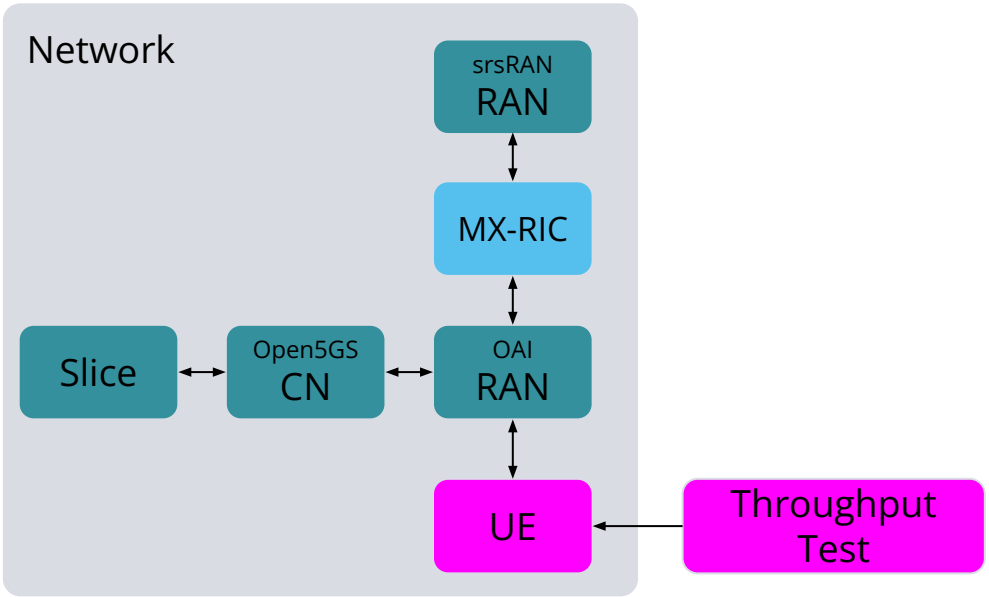
Zero-Touch 5G SA



- Deploy the network
- Check network status

```
brc install network simple-sa-oran.yaml
```

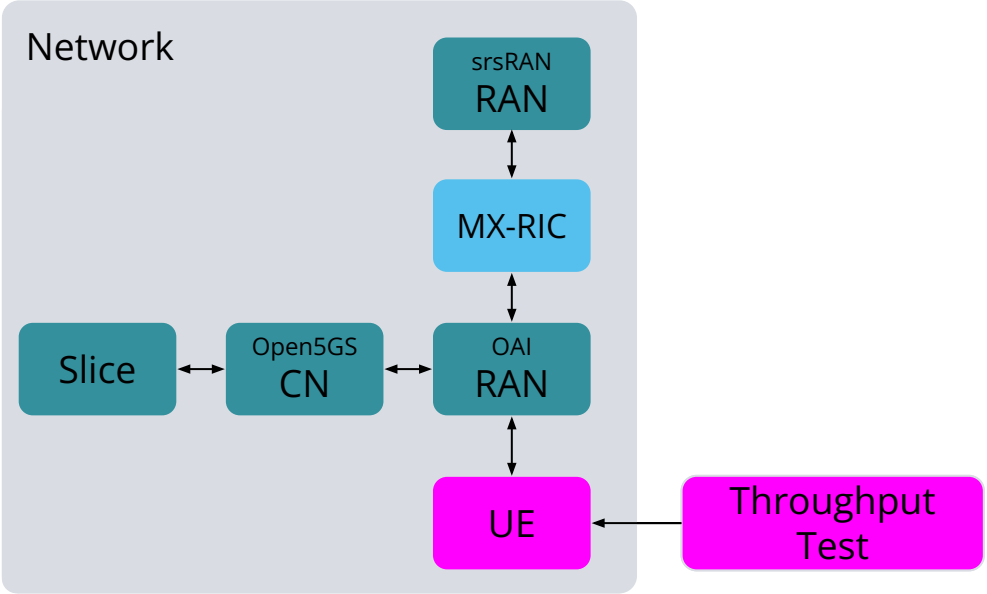
Zero-Touch 5G SA



Network Intent

```yaml
apiVersion: athena.trirematics.io/v1
kind: Terminal
metadata:
    name: ue1
    namespace: trirematics
spec:
    vendor: oai
    stack: 5g-sa
    model: terminal/nr-rfsim
    preferred-access: oai-gnb.bubbleran
    target-cores:
        - ogs.bubbleran
    identity:
        imsi: "001010000000001"
        ...
    slice:
        dnn: "internet"
        ...
    radio:
        bands:
            - n78
    readiness-check:
        ...
```

Zero-Touch 5G SA

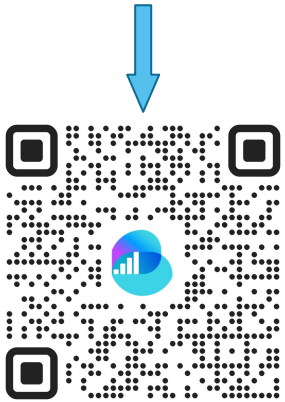


- UE in the loop
- Checkout the UE messages in Wireshark
- RTT test with the UE
- Throughput test with the UE

```
brc test rtt ue1 gateway
brc test rtt ue1 8.8.8.8 -- -c 8
brc test throughput dl gateway -- -t 10
```

Inventory

- **Compose** any network with these **building blocks** from our supported vendors !
- Check out various deployment examples in our **open documentation**



Vendor	Kind	Composition Model
BubbleRAN	RIC	*mosaic5g/flexric*
	xApp	*Monitoring, RC, LLC & CCC* ***(listed in next section)***
	Database	*mosaic5g/xapps-sdl*
	Terminal	*terminal/nr-rfsim* *terminal/quectel* *terminal/external*
	RAN	*bubbleran/liteon-aio*
OAI	RAN	*oai-ran/monolithic-gnb* *oai-ran/monolithic-gnb-ru* *oai-ran/cu-du* *oai-ran/cu-du-ru*
	CN	*oai-cn/epc* *oai-cn/minimal*
srsRAN	RAN	*srs/monolithic-gnb*
Open5GS	CN	*open5gs/5gc* *open5gs/epc* *open5gs/full*
Amarisoft	RAN	*amarisoft/ran-e2* *amarisoft/basic-ran*
	CN	*amarisoft/basic-cn* *amarisoft/cn-with-ims*

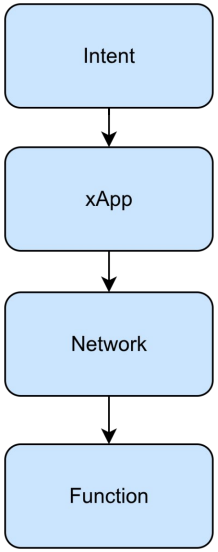
Summary

- **Intent-driven**, **zero-touch automation** across RAN, Core, and Terminal
- **Declarative intents** define desired network realized through SMO
- **Multi-vendor**, **modular design** enabling full 5G SA deployments
- End-to-end **lifecycle automation** built on **3GPP & O-RAN** principles

02

xApp Automation

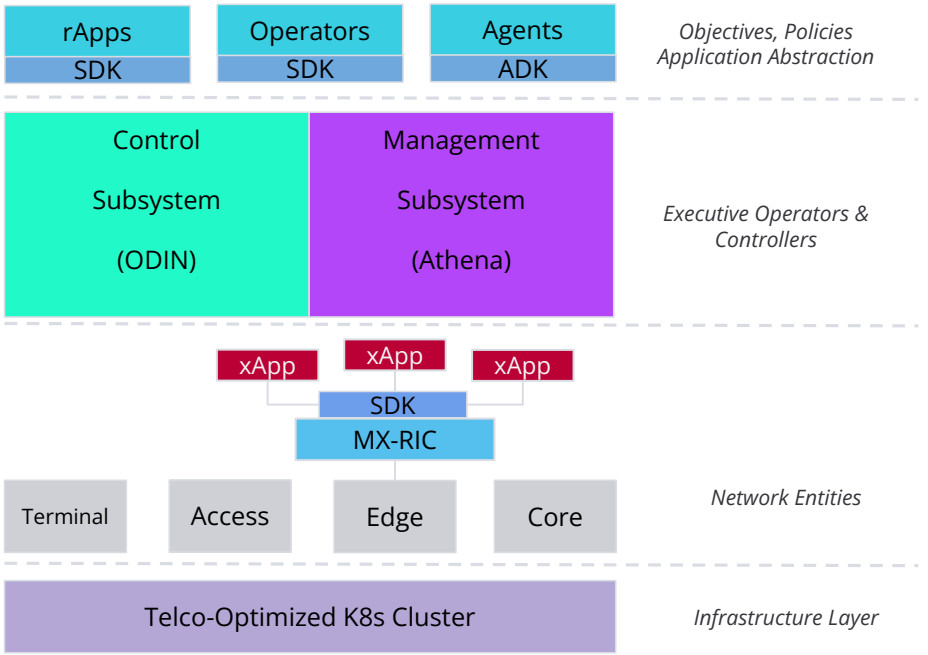
Network observability by visualizing, monitoring, and analyzing metrics



MX-RIC: Ultimate Power for the Open RAN

MX-RIC is BubbleRAN's Near-Real Time RIC, offering:

- **"Multi-X" Versatility:**
  - Supports multiple RAT (4G/5G), platform (x86/ARM), language (C/Python), version (E2AP V1-V3) and vendor ecosystem (LITEON, Benetel)
- **Carrier-Grade Performance & Control:**
  - Ultra-low latency (<1ms) for real-time applications.
  - Deep network visibility with a rich set of standard (O-RAN KPM/RC/CCC/LLC) and custom Service Models (e.g., MAC, RLC, PDCP)



Intelligent Automation with xApps

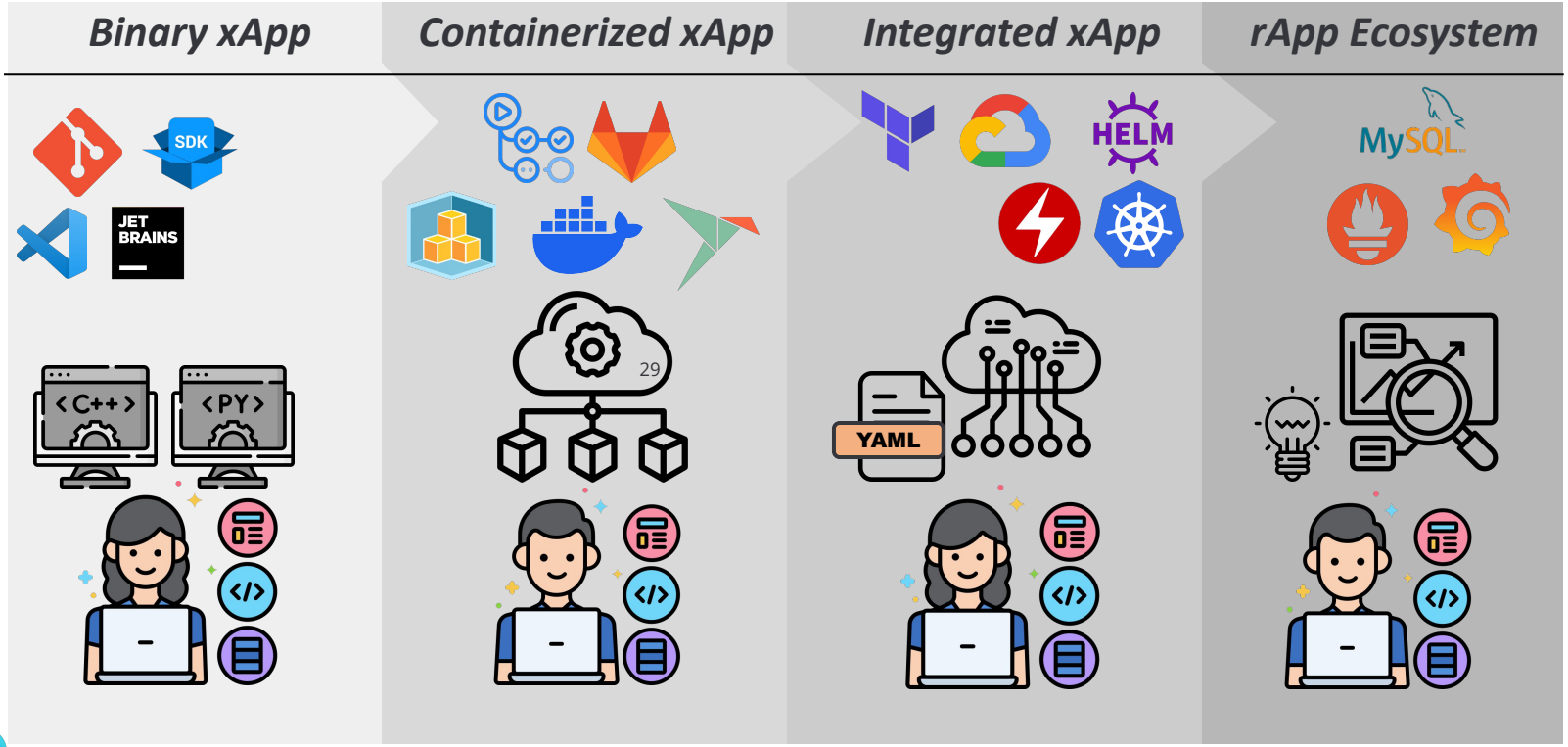
Deploy and Automate everywhere.
BubbleRAN gives you total freedom, seamlessly running powerful xApps in any environment—from containerized to bare metal xApps.

Ready-to-Go xApp Toolkit.
Instantly access a catalogue of applications to supercharge your network.

xApp	Description	Language
monitoring-[oran/custom]	Monitoring xApp for O-RAN and/or custom SM	C/Python
handover-rc-ctrl	Static handover control xApp with ORAN RC SM	C/Python
interactive	Easily test your advanced use case by developing your own xApp (Slicing, ISAC)	C/Python

BubbleRAN's Open-source xApp SDK

xApp Lifecycle



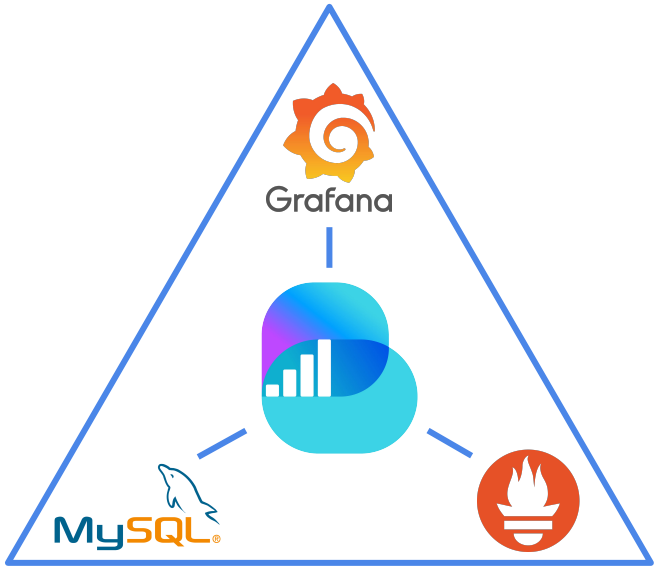
MX-PDK: Total Observability, Simplified

All data in one place.
Fuses deep 5G stack metrics with cloud-native infrastructure data from multiple data sources:

- Prometheus
  - Cluster nodes statistics
  - Cluster status, pod and deployments
  - Power consumption per node and cluster service
- MySQL
  - xApp metrics service models

Dynamic dataset collection for intelligence.
Provides the rich, unified data set required to fuel advanced AI and analytics.

Observability Stack (Demo ready).
A complete, easy-to-deploy observability solution perfect for live showcases with Grafana.



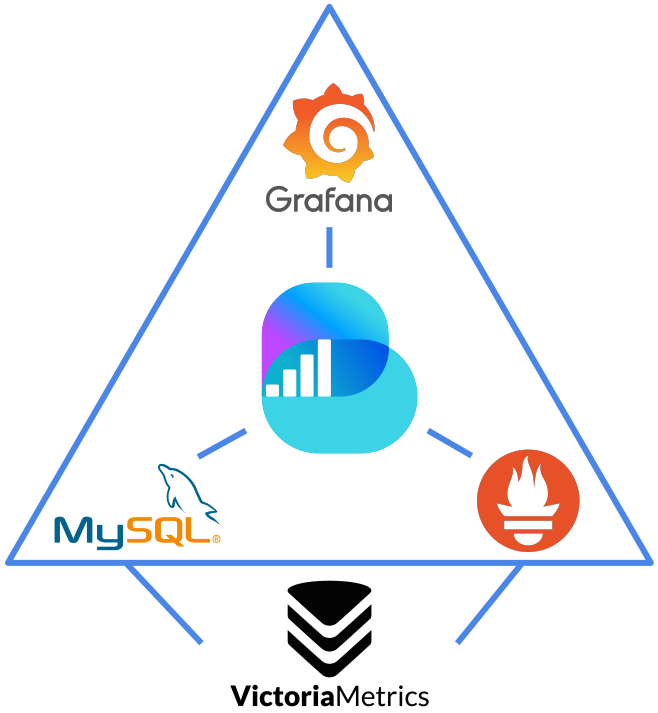
MX-PDK: Total Observability, Simplified

All data in one place.
Fuses deep 5G stack metrics with cloud-native infrastructure data from multiple data sources:

- Prometheus
  - Cluster nodes statistics
  - Cluster status, pod and deployments
  - Power consumption per node and cluster service
- MySQL
  - xApp metrics service models

Dynamic dataset collection for intelligence.
Provides the rich, unified data set required to fuel advanced AI and analytics.

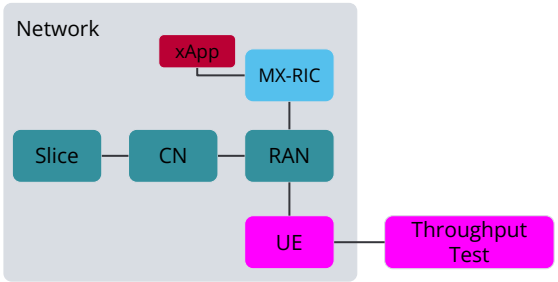
Observability Stack (Demo ready).
A complete, easy-to-deploy observability solution perfect for live showcases with Grafana.



Let’s monitor your 5G network: (I) Deploy a monitoring xApp

Plug and play.
xApps can either be integrated into existing network deployments or developed alongside them.

Changes? Only when needed.
xApps follow a declarative schema as part of the network configuration. Objects are updated only to match the desired network state.

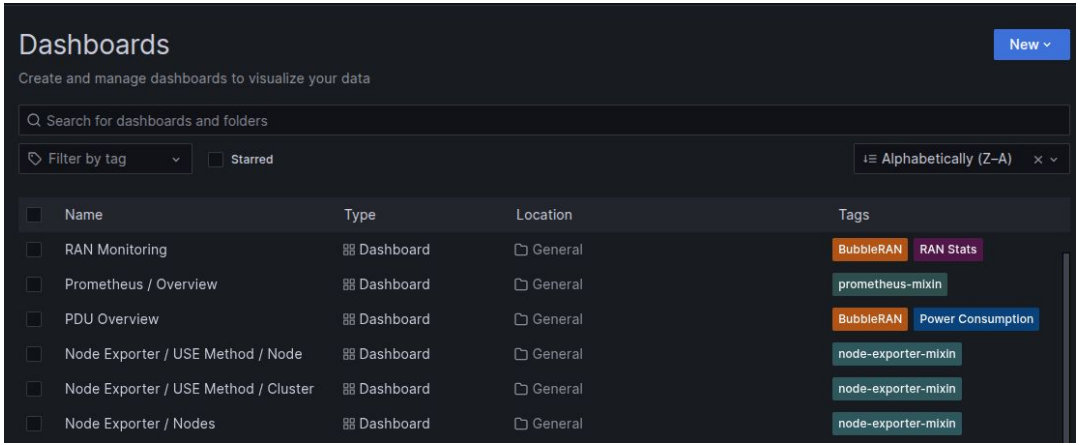



```
    edge:
...
        -   name: monitoring
            stack: 5g-sa
            model: mosaic5g/monitoring-c
            profiles:
                - rlc-sm
                - pdcp-sm
                - mac-sm
                - gtp-sm
                - slice-sm
                - kpm-sm
                - database
        -   name: sdl
            stack: 5g-sa
            model: mosaic5g/xapps-sdl
...
```

Let’s monitor your 5G network: - Done

Let’s access Grafana to see the result.
It embeds a set of dedicated dashboards targeting and mixing the different data sources, both network and cluster related.

Not enough? Create your dashboard.
Advanced use cases requires impactful GUI. Grafana allows fully customizable interface, though which you can build your own!



Summary

Our **MX-RIC provides total freedom**, integrating seamlessly with multiple RAT (4G/5G), vendors or hardware.

BubbleRAN's xApp ecosystem lets you deploy **ready-to-go xApps** from the catalogue or **build your own** with a powerful SDK to easily automate the RAN.

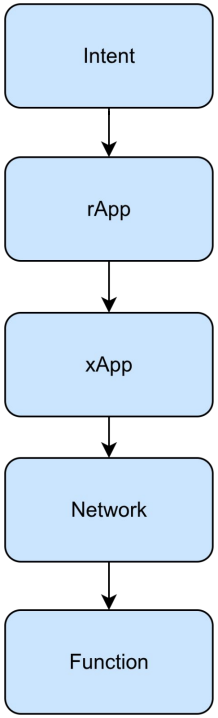
MX-PDK drives smarter RAN automation by unifying **cloud monitoring** with granular, **real-time data** collected by **xApps**.

xApp category	Available
Key Performance Monitoring	Standard (e.g., O-RAN KPM, RC) and custom Service Models (e.g., MAC, RLC, PDCP)
RAN Control	Slicing, Handover, Service Level Agreement (SLA), Modulation and Coding Scheme (MCS)
Low Layer Control	Sensing, Object Detection
Cell Configuration and Control	Bandwidth Part (BWP)

03

rApp Automation

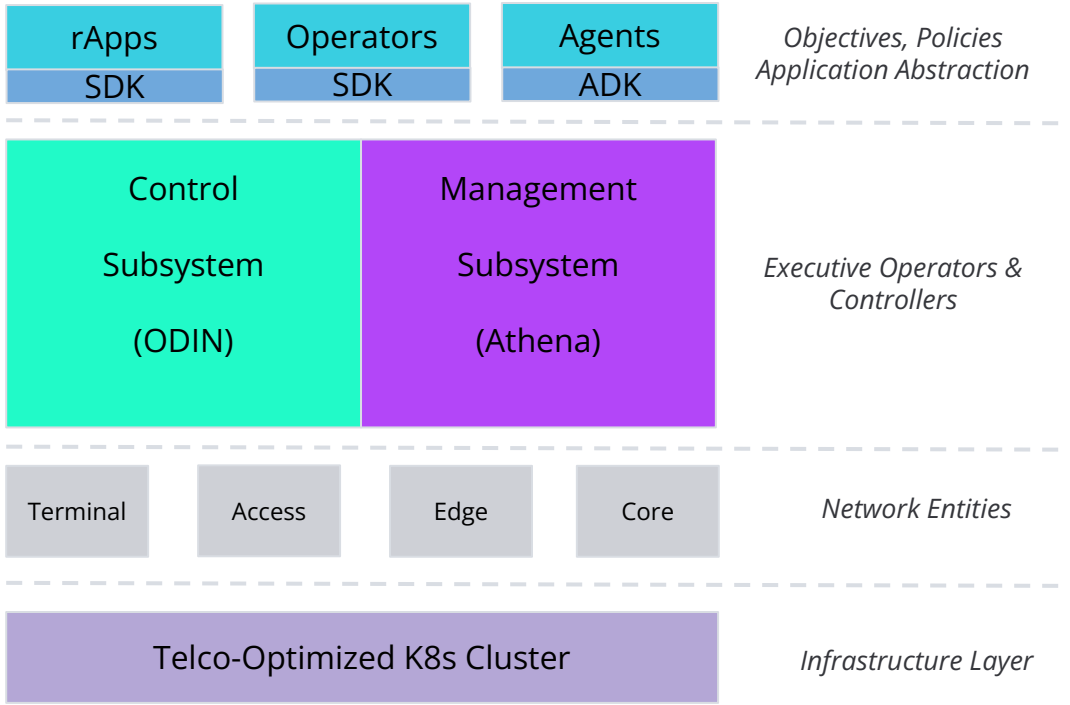
Enabling Non-Real-Time Control and Management



BubbleRAN’s Non RT RIC

A logical function within **SMO** that enables non-real-time control and optimization of RAN elements and resources, AI/ML workflow including **model training and updates**, and **policy-based guidance** of applications/features in Near-RT RIC.

Gateway of the control plane to the operator plane for complex management+control operation



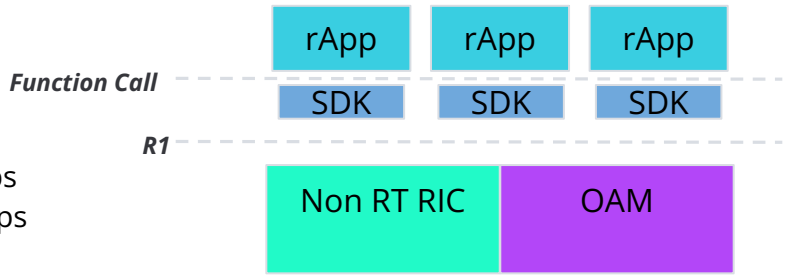
Non-RT RIC Interfaces and rApp SDK

Input: R1 Interface (K8s Custom Resource)

- The **rApp SDK** is a Python library that simplifies the *design, development, onboarding* of rApps within the ODIN platform.
- It acts as a high-level interface—**aligned** to the **R1 interface**—enabling rApps to interact with the platform without dealing directly with low-level APIs (CRs).
- Supported Services (as of now)
  a. Monitoring Services
  b. A1-related Services
  c. RAN OAM-related Services
- Installation: pip3 install br-rapp-sdk
- Repository: https://github.com/bubbleran/rapp_sdk

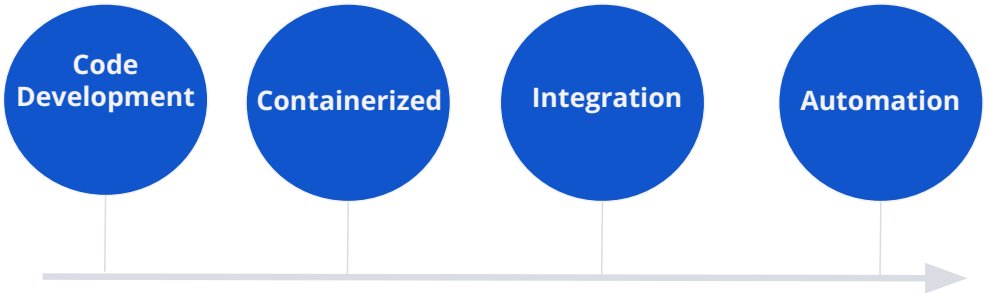
Output

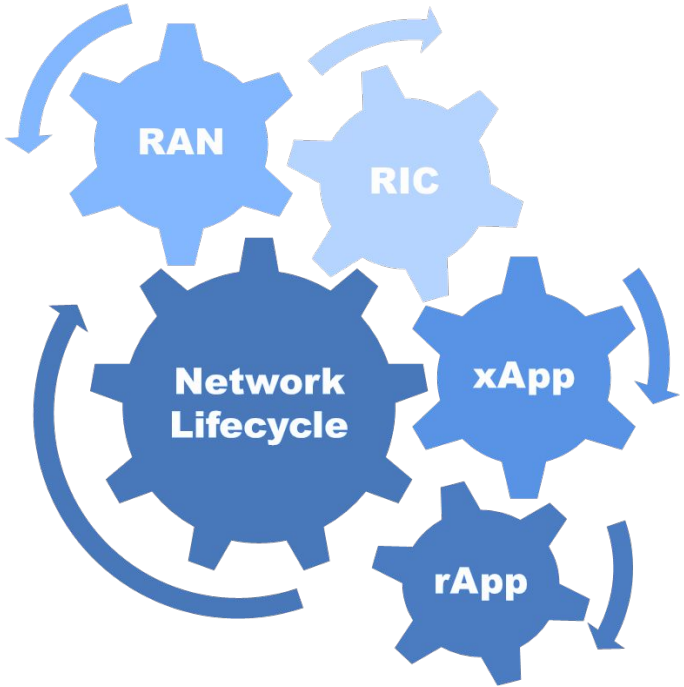
- **A1 Interface** for communication with Near-RT RIC and xApps
- **OAM Interface** for managing the lifecycle of rApps and xApps



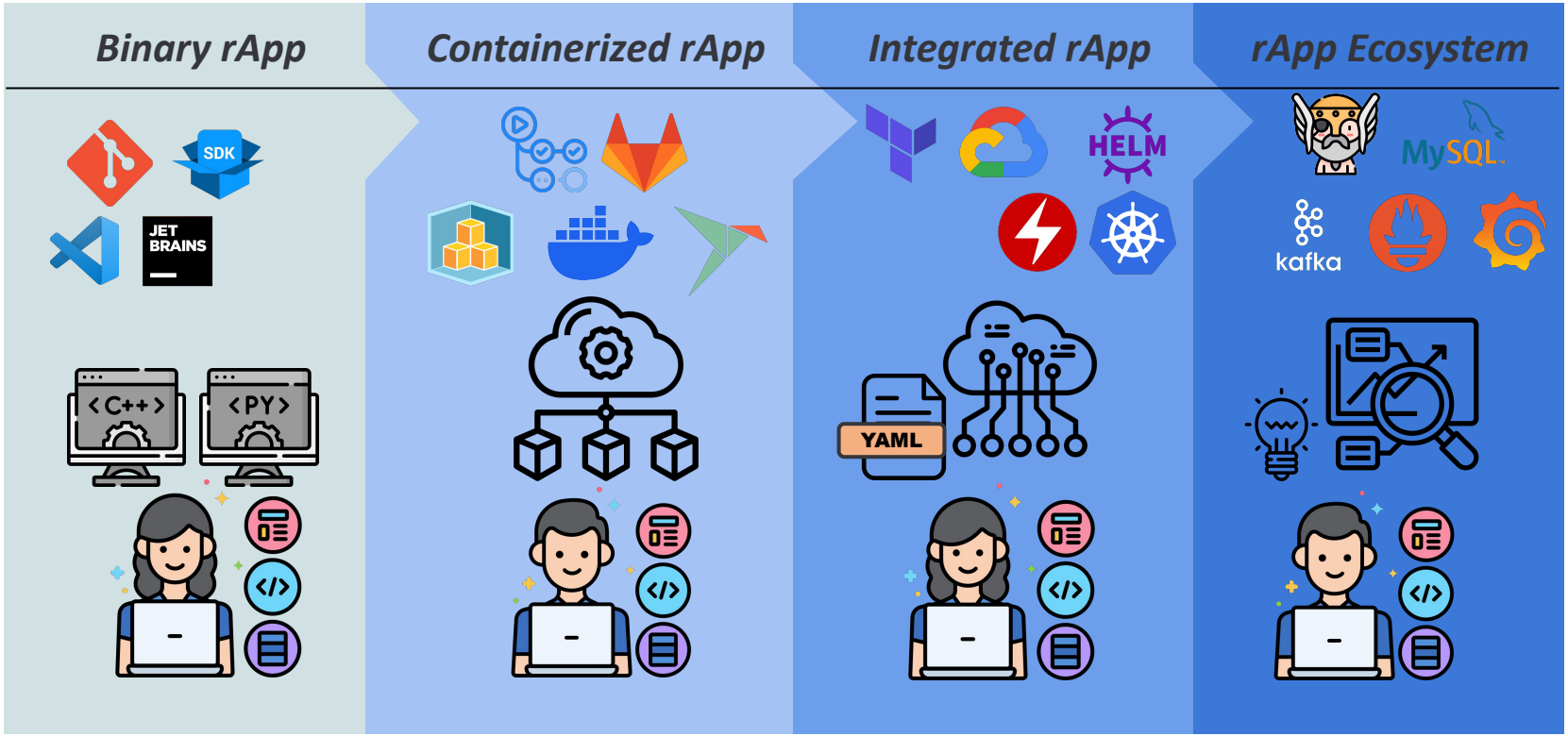
rApp Lifecycle

- Accelerating innovative rApp development
- Delivered as containerized, cloud-native applications
- Enabling dynamic rApp operations through the SMO
- Leveraging the R1 interface for seamless integration
- Bridging xApps and rApps for unified intelligence
- Ensuring a smooth transition from RAN to data-driven insights





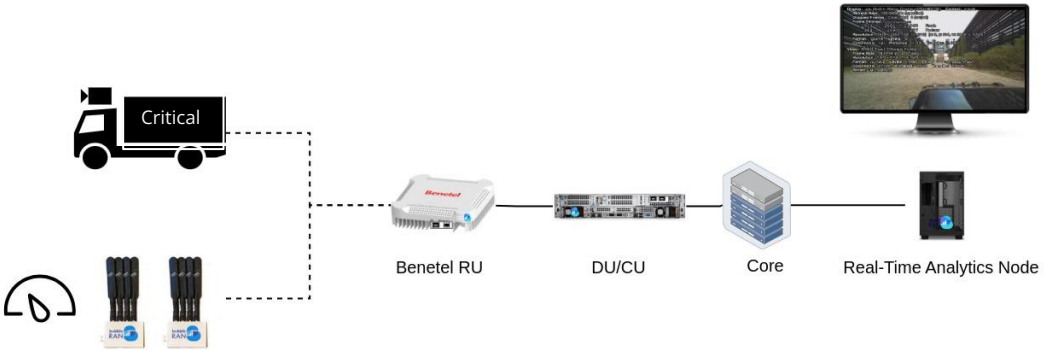
rApp Lifecycle



BubbleRAN's rApps

rApp category	Available	Planned
Network Deployment	TDD Reconfiguration, Network Planning, Spectrum Allocation	Energy Saving rApp
Network Evolution & Optimization	Slice Enforcement, SLA Provisioner	Load Balance Handover rapp, Digital Twin-assisted Recommender rApp
Network Monitoring	SMs-based rApp, Sensing,	Intent-based Monitoring rApp
Network Automation	Network, Slice, Digital Twin	Green, Security
Network Intelligence	Orchestrator Agent, OAM Agent, RIC Agent, Observibility Agent	API Agent, Anomaly Detection Agent, Slice (SO) Agent

Demo: Mission-Critical V2X Uplink Video



Demo: Mission-Critical V2X Uplink Video

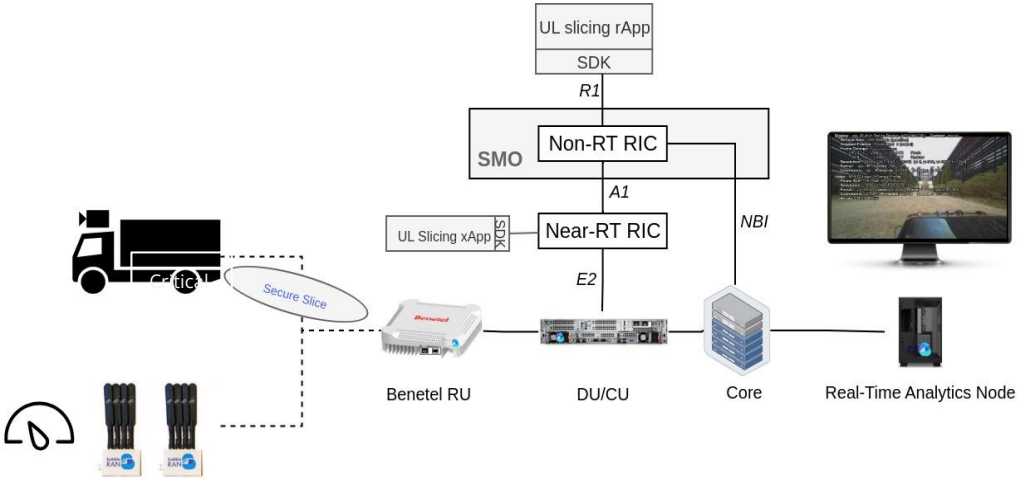
Demo: Mission-Critical V2X Uplink Video

UL Slicing rApp

- Maps IMSI to RAN_UE_ID or RNTI.
- Sends an uplink slicing request, targeting a specific UE to be associated with the secure slice.

UL Slicing xApp

- Creates or updates the slice based on a percentage of PRBs.
- Associates the targeted UE with the slice.



Demo: Mission-Critical V2X Uplink Video

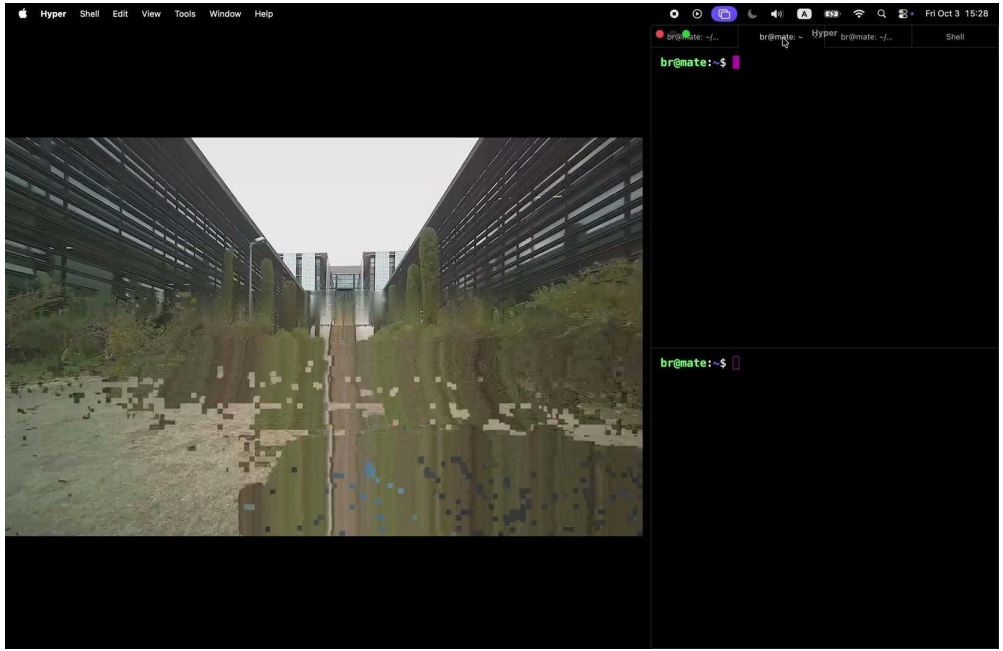
- **Deploy the rApp**

```
brc install rapp rapp.yaml
```

```yaml
apiVersion: odin.trirematics.io/v1
kind: DynamicrApp
metadata:
  name: ulslicing
  namespace: trirematics
spec:
  model: bubbleran/ul-slicing-rapp
  rbac:
    scope: namespaced
    grants:
      - group: odin.trirematics.io
        resources: ["policyjobs"]
        mode: read-write
```

Demo: Mission-Critical V2X Uplink Video

- **Without Slicing**.



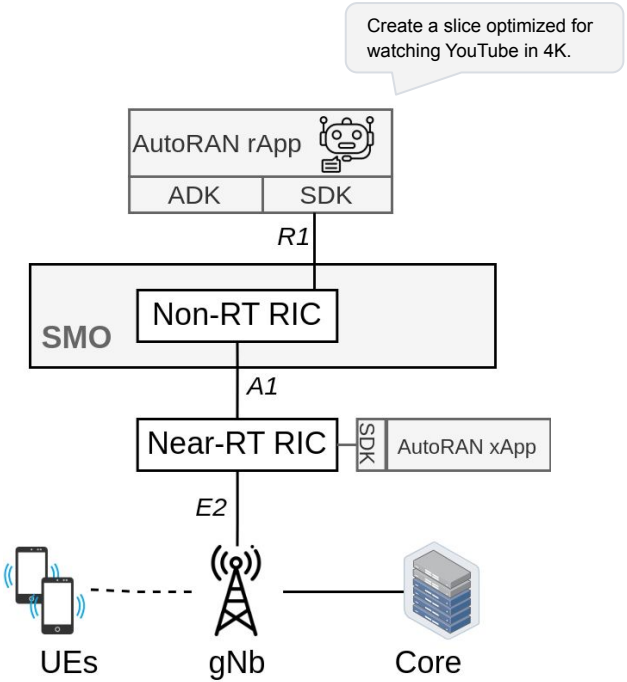
Demo: Intent-driven RAN Automation

LLM-Powered AutoRAN rApp

- Leverages Large Language Models (LLMs) to interpret operator intents and translate them into policy decisions.
- Provides intelligent recommendations by analyzing service needs

AutoRAN xApp

- Automates near-real-time slice provisioning based on dynamic network conditions and operator policies
- Monitors KPIs and adjusts slice parameters proactively to maintain service quality.



OAI Workshop
May 21-22, 2025
Ericsson, Kista, Sweden

Summary

BubbleRAN's Non-RT RIC features a **service-based architecture** that enables rApps to both provide and consume services.

BubbleRAN's rApp ecosystem lets you deploy **ready-to-go rApps** from the catalogue or **build your own** with a powerful SDK and CDK to easily automate the RAN.

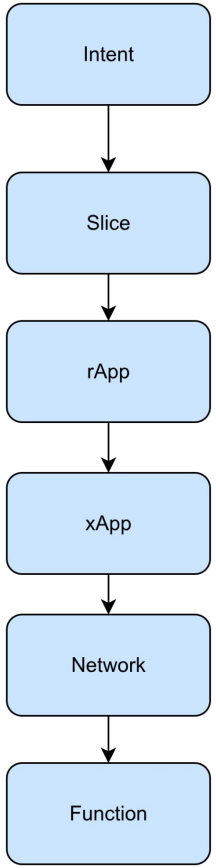
BubbleRAN's MX-PDK acts as a key enabler for **automation** and **intelligence**, which are the core pillars of **autonomous networks**.

rApp category	Available
Network Deployment	TDD Reconfiguration, Network Planning, Spectrum Allocation
Network Evolution & Optimization	Slice Enforcement, SLA Provisioner
Network Monitoring	SMs-based rApp, Sensing,
K8s Operators	Network, Slice, Digital Twin
AI Agent	Orchestrator Agent, OAM Agent, RIC Agent, Observibility agent

04

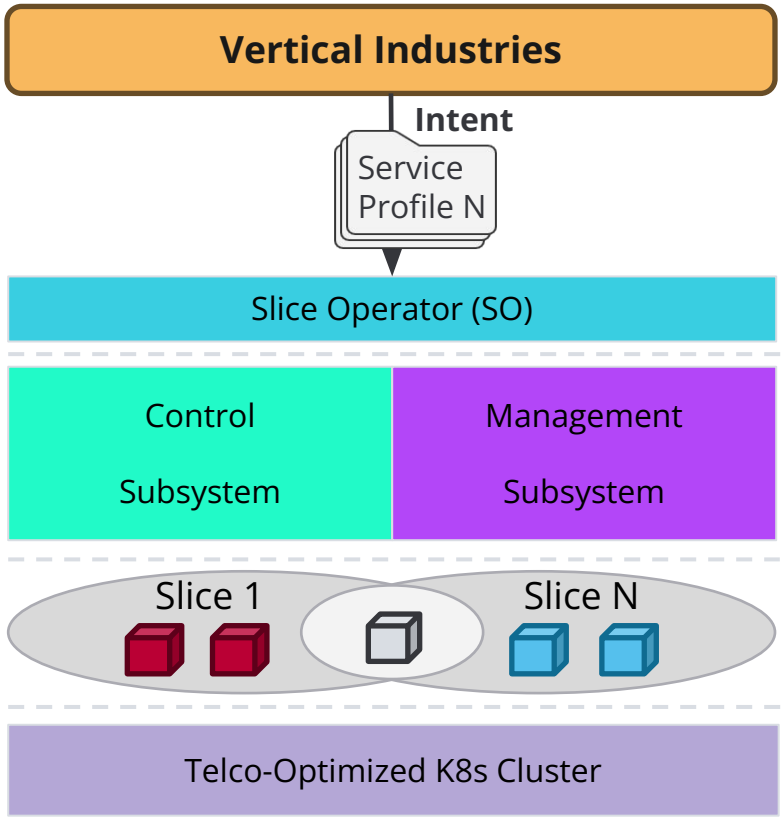
Slice Automation

3GPP & O-RAN *compliant network slicing for end-to-end service guarantees*



BubbleRAN Network Slicing Framework

- **Cloud-native** NSMF/NSSMF built on top of the **MX-Operator** that automates end-to-end slice lifecycle across **RAN & Core**.
- Continuous reconciliation loop keeps slices aligned with requested SLAs.
- **Input**: Service Profile Intent → **Output**: Slice Profile (deployable blueprint).
- A **Service Profile** expresses **what** is needed (latency, bandwidth, area, use-case).



bubble RAN

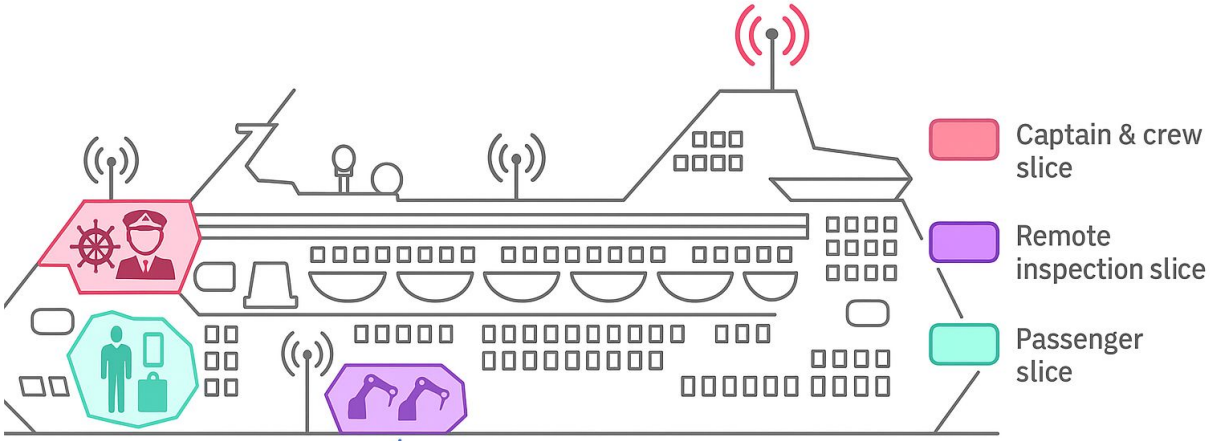
COMPLIANCE COMPLIANCE COMPLIANCE

✓ 3GPP
✓ O-RAN

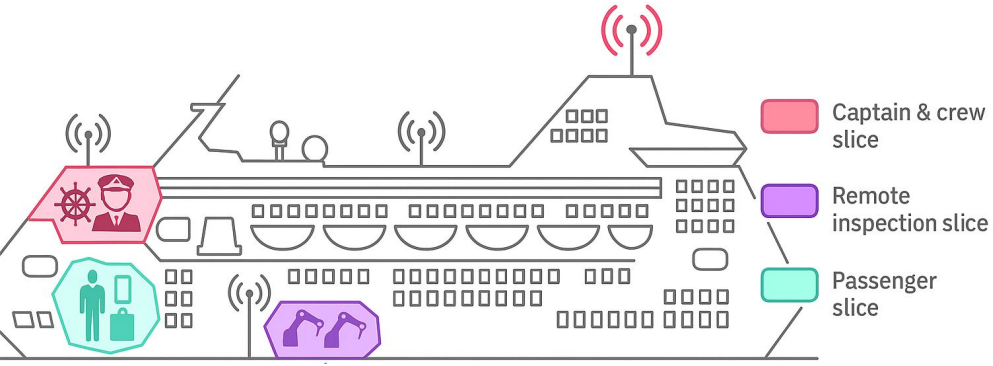
Maritime Use Case

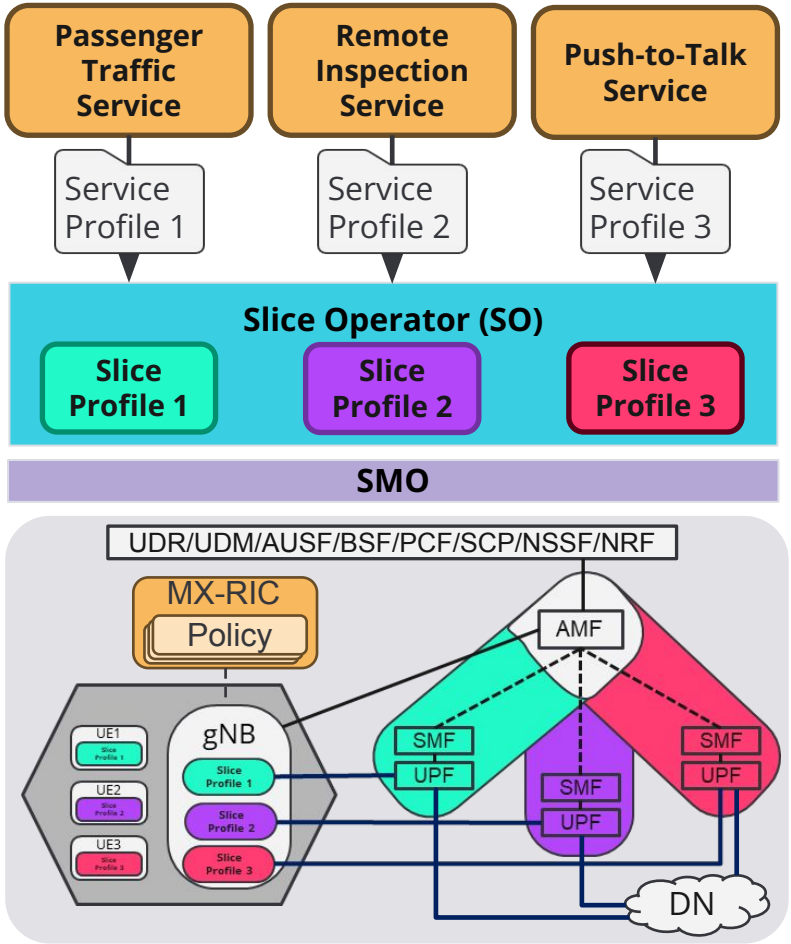
3 different Network Connection for 3 Use Cases :

- Push-to-talk
  - Network connection between captain and the crew
- Passenger Traffic
  - Internet access for the passengers of the ship
- Remote Inspection
  - Sending the video stream from the ship to the shore



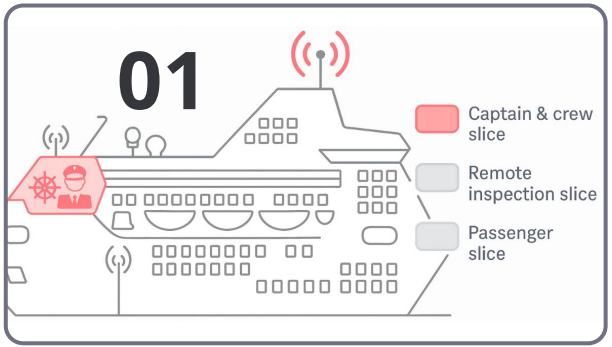
Maritime Use Case

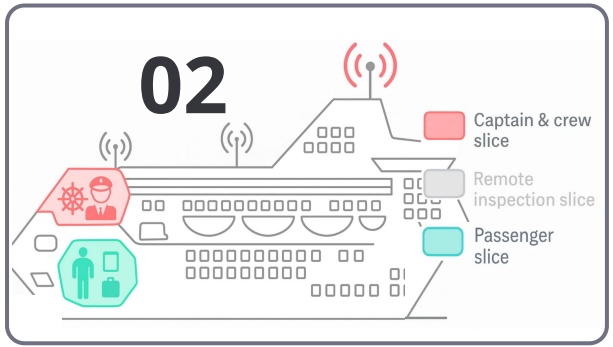


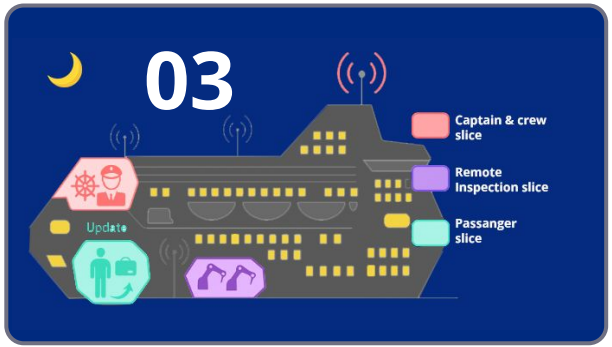


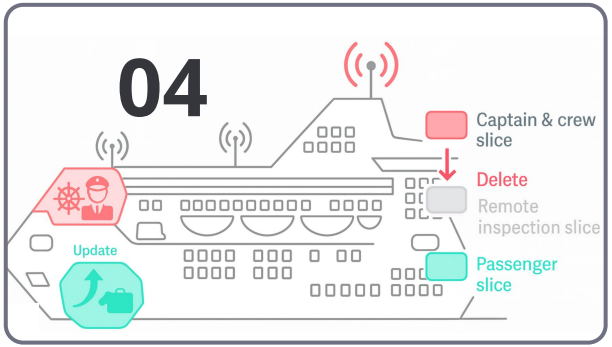
- Shared control-plane NFs across slices
  - gNB, AMF, UDR, UDM,AUSF,BSF,PCF,SCP,NSSF,NRF
- Dedicated SMF+UPF per slice (data-plane isolation)
- UEs → single assigned slice (NSSAI-based)
- Per-slice SLAs

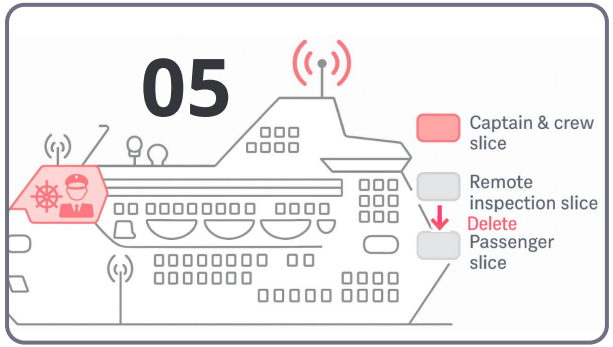
Maritime use case demo











01 **.Ship on shore** → captain-crew slice created

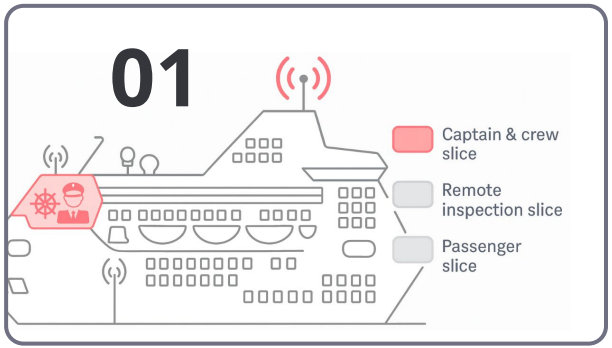
02 **.Ship takes off** → passenger slice created

03 **.At night** → update slice for passengers and create remote inspection slice

04 **.At daytime** → remote inspection is deleted and passenger slice updated

05 **.Ship back to shore** → passenger slice deleted

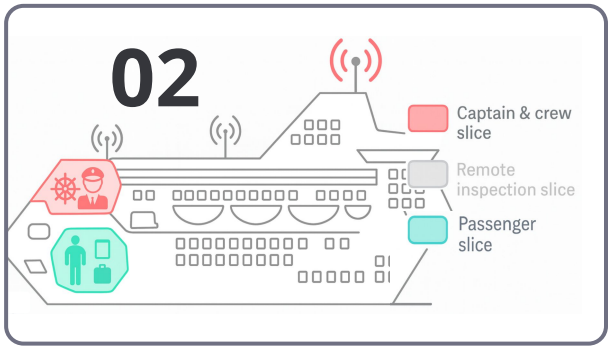
Maritime use case Service Profiles



Push-to-Talk Service

```yaml
apiVersion: athena.trirematics.io/v1
kind: ServiceProfile
metadata:
  name: crew
  namespace: trirematics
spec:
  coverage:
      ...
  data-flows:
    - name: crew
      description: |
        Video chat among the captain and crew.
      data-network: p2p
      traffic-class: burstable
      content-rate:
        downlink:
          maximum: 10Mbps
          average: 10Mbps
        uplink:
          maximum: 10Mbps
          average: 10Mbps
      latency-class: p2p
      payload-types:
        - use-case: video-chat
          application-protocol: rtp
          transport-protocol: udp
          content-type:
            - video/h264
            - audio/opus
```

Maritime use case Service Profiles



Push-to-Talk Service

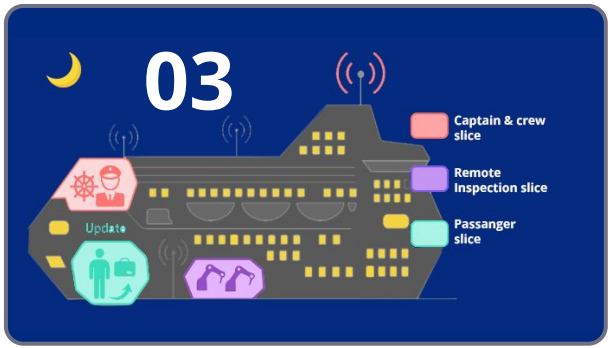
```yaml
apiVersion: athena.trirematics.io/v1
kind: ServiceProfile
metadata:
  name: crew
  namespace: trirematics
spec:
  coverage:
    ...
  data-flows:
    - name: crew
      description: |
        Video chat among the captain and crew.
      data-network: p2p
      traffic-class: burstable
      content-rate:
        downlink:
          maximum: 10Mbps
          average: 10Mbps
        uplink:
          maximum: 10Mbps
          average: 10Mbps
      latency-class: p2p
      payload-types:
        - use-case: video-chat
          application-protocol: rtp
          transport-protocol: udp
          content-type:
            - video/h264
            - audio/opus
```

Passenger Traffic Service

```yaml
apiVersion: athena.trirematics.io/v1
kind: ServiceProfile
metadata:
  name: passengers
  namespace: trirematics
spec:
  coverage:
    ...
  data-flows:
    - name: passengers
      description: |
        Data flows designated for passenger traffic.
      data-network: internet
      traffic-class: burstable
      content-rate:
        downlink:
          maximum: 50Mbps
          average: 50Mbps
        uplink:
          maximum: 20Mbps
          average: 20Mbps
      latency-class: unrestricted
      payload-types:
        - use-case: data-transfer
          application-protocol: http/3
          transport-protocol: tcp
          content-type:
            - application/octet-stream
```

bubble RAN

Maritime use case Service Profiles



Push-to-Talk Service

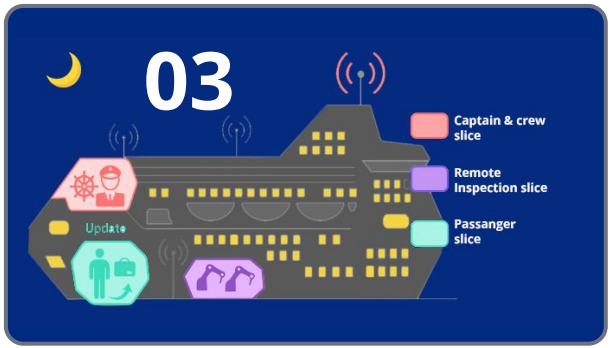
```yaml
apiVersion: athena.trirematics.io/v1
kind: ServiceProfile
metadata:
  name: crew
  namespace: trirematics
spec:
  coverage:
    ...
  data-flows:
    - name: crew
      description: |
        Video chat among the captain and crew.
      data-network: p2p
      traffic-class: burstable
      content-rate:
        downlink:
          maximum: 10Mbps
          average: 10Mbps
        uplink:
          maximum: 10Mbps
          average: 10Mbps
      latency-class: p2p
      payload-types:
        - use-case: video-chat
          application-protocol: rtp
          transport-protocol: udp
          content-type:
            - video/h264
            - audio/opus
```

Passenger Traffic Service

```yaml
apiVersion: athena.trirematics.io/v1
kind: ServiceProfile
metadata:
  name: passengers
  namespace: trirematics
spec:
  coverage:
    ...
  data-flows:
    - name: passengers
      description: |
        Data flows designated for passenger traffic.
      data-network: internet
      traffic-class: burstable
      content-rate:
        downlink:
          maximum: 25Mbps
          average: 25Mbps
        uplink:
          maximum: 10Mbps
          average: 10Mbps
      latency-class: unrestricted
      payload-types:
        - use-case: data-transfer
          application-protocol: http/3
          transport-protocol: tcp
          content-type:
            - application/octet-stream
```

bubble RAN

Maritime use case Service Profiles



Push-to-Talk Service

```yaml
apiVersion: athena.trirematics.io/v1
kind: ServiceProfile
metadata:
  name: crew
  namespace: trirematics
spec:
  coverage:
      ...
  data-flows:
    - name: crew
      description: |
        Video chat among the captain and crew.
      data-network: p2p
      traffic-class: burstable
      content-rate:
        downlink:
          maximum: 10Mbps
          average: 10Mbps
        uplink:
          maximum: 10Mbps
          average: 10Mbps
      latency-class: p2p
      payload-types:
        - use-case: video-chat
          application-protocol: rtp
          transport-protocol: udp
          content-type:
            - video/h264
            - audio/opus
```

Passenger Traffic Service

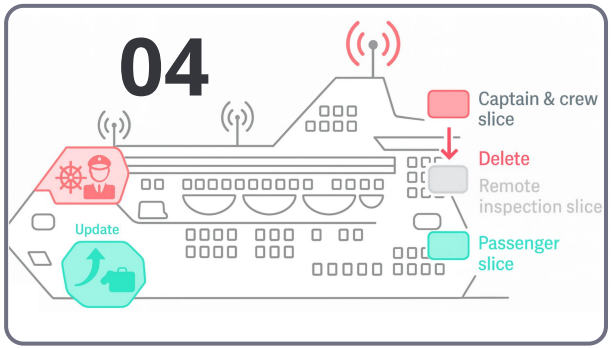
```yaml
apiVersion: athena.trirematics.io/v1
kind: ServiceProfile
metadata:
  name: passengers
  namespace: trirematics
spec:
  coverage:
    ...
  data-flows:
    - name: passengers
      description: |
        Data flows designated for passenger traffic.
      data-network: internet
      traffic-class: burstable
      content-rate:
        downlink:
          maximum: 25Mbps
          average: 25Mbps
        uplink:
          maximum: 10Mbps
          average: 10Mbps
      latency-class: unrestricted
      payload-types:
        - use-case: data-transfer
          application-protocol: http/3
          transport-protocol: tcp
          content-type:
            - application/octet-stream
```

Remote Inspection Service

```yaml
apiVersion: athena.trirematics.io/v1
kind: ServiceProfile
metadata:
  name: inspection
  namespace: trirematics
spec:
  coverage:
      ...
  data-flows:
    - name: inspection
      description: |
        Data flows designated for remote inspection.
      data-network: miot
      traffic-class: persistent
      content-rate:
        downlink:
          maximum: 30Mbps
          average: 30Mbps
        uplink:
          maximum: 50Mbps
          average: 50Mbps
      latency-class: mec
      payload-types:
        - use-case: video-stream
          application-protocol: srtp
          transport-protocol: udp
          content-type:
            - video/h264
            - audio/opus
```

bubble RAN

Maritime use case Service Profiles



Push-to-Talk Service

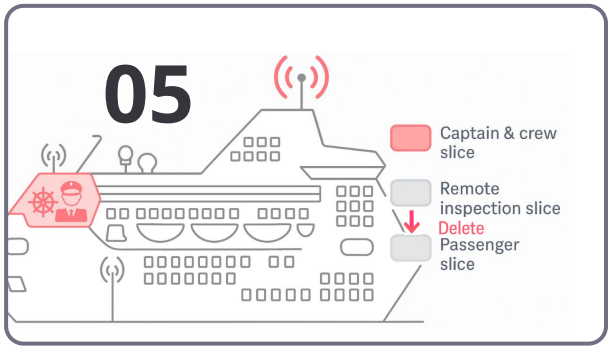
```yaml
apiVersion: athena.trirematics.io/v1
kind: ServiceProfile
metadata:
  name: crew
  namespace: trirematics
spec:
  coverage:
    ...
  data-flows:
    - name: crew
      description: |
        Video chat among the captain and crew.
      data-network: p2p
      traffic-class: burstable
      content-rate:
        downlink:
          maximum: 10Mbps
          average: 10Mbps
        uplink:
          maximum: 10Mbps
          average: 10Mbps
      latency-class: p2p
      payload-types:
        - use-case: video-chat
          application-protocol: rtp
          transport-protocol: udp
          content-type:
            - video/h264
            - audio/opus
```

Passenger Traffic Service

```yaml
apiVersion: athena.trirematics.io/v1
kind: ServiceProfile
metadata:
  name: passengers
  namespace: trirematics
spec:
  coverage:
    ...
  data-flows:
    - name: passengers
      description: |
        Data flows designated for passenger traffic.
      data-network: internet
      traffic-class: burstable
      content-rate:
        downlink:
          maximum: 50Mbps
          average: 50Mbps
        uplink:
          maximum: 20Mbps
          average: 20Mbps
      latency-class: unrestricted
      payload-types:
        - use-case: data-transfer
          application-protocol: http/3
          transport-protocol: tcp
          content-type:
            - application/octet-stream
```

bubble RAN

Maritime use case Service Profiles



Push-to-Talk Service

```yaml
apiVersion: athena.trirematics.io/v1
kind: ServiceProfile
metadata:
  name: crew
  namespace: trirematics
spec:
  coverage:
      ...
  data-flows:
    - name: crew
      description: |
        Video chat among the captain and crew.
      data-network: p2p
      traffic-class: burstable
      content-rate:
        downlink:
          maximum: 10Mbps
          average: 10Mbps
        uplink:
          maximum: 10Mbps
          average: 10Mbps
      latency-class: p2p
      payload-types:
        - use-case: video-chat
          application-protocol: rtp
          transport-protocol: udp
          content-type:
            - video/h264
            - audio/opus
```

Summary

- BubbleRAN's **3GPP and O-RAN-compliant** Slice Operator **automates** the full **slice lifecycle** across RAN and Core.
- **Intent** is defined as **Service Profiles** specifying high-level requirements (e.g., latency, bandwidth) which are then **automatically** translated into deployable **Slice Profiles**.
- Reconciliation loops ensure slices always meet their **SLA targets**, enabling **closed-loop**, **intent-driven automation**.

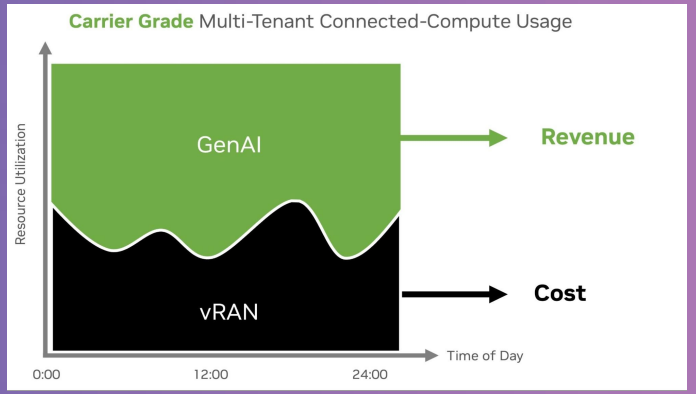
BubbleRAN

AUTOMATION INTELLIGENCE OPEN ECOSYSTEM

bubble RAN

🔀 The phase changes in 6G Ecosystem and Economy driven by AI

- Built-in HPC and Agentic AI factory
- Fusion of Physical and Digital Network Replicas
- Application-centric Ultra-flexible Decision-making
- Self-Synthesized and Evolving Networks
- Human-to-Agent and Agent-to-Network Symbiosis



Questions in the Chat

- Q1: So many features, how to facilitate app interactions?
  - SBA for Apps to facilitate App collaboration
- Q2: xApp conflict mitigation?
  - Traffic steering and energy efficiency
- Q3: What is the main challenge public operator are facing?
- Q4. K8s Operator and AI agents?
- How dynamic are the slices in BubbleRAN?
- Are the slices are 3GPP only or they are support O-RAN RAN Slicing?
-

Q&A

Thank you so much!

E-mail: contact@bubbleran.com
Website: www.bubbleran.com
Linkedin: https://www.linkedin.com/company/bubbleran

And remember our next webinar:

MX-AI: The Agentic Shift in Network Operations

Tuesday 4th of November at 4PM CET