

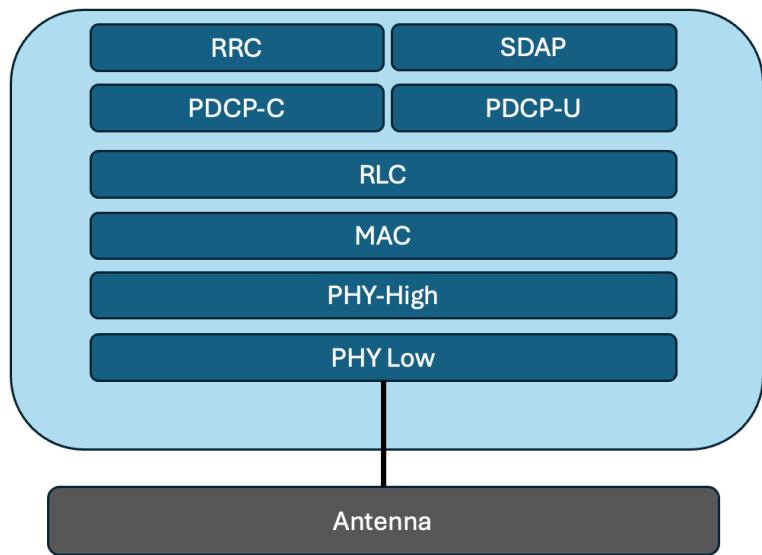
# Hands-On Introduction to Open RAN Setting Up a 5G Network with Open Source Components

---

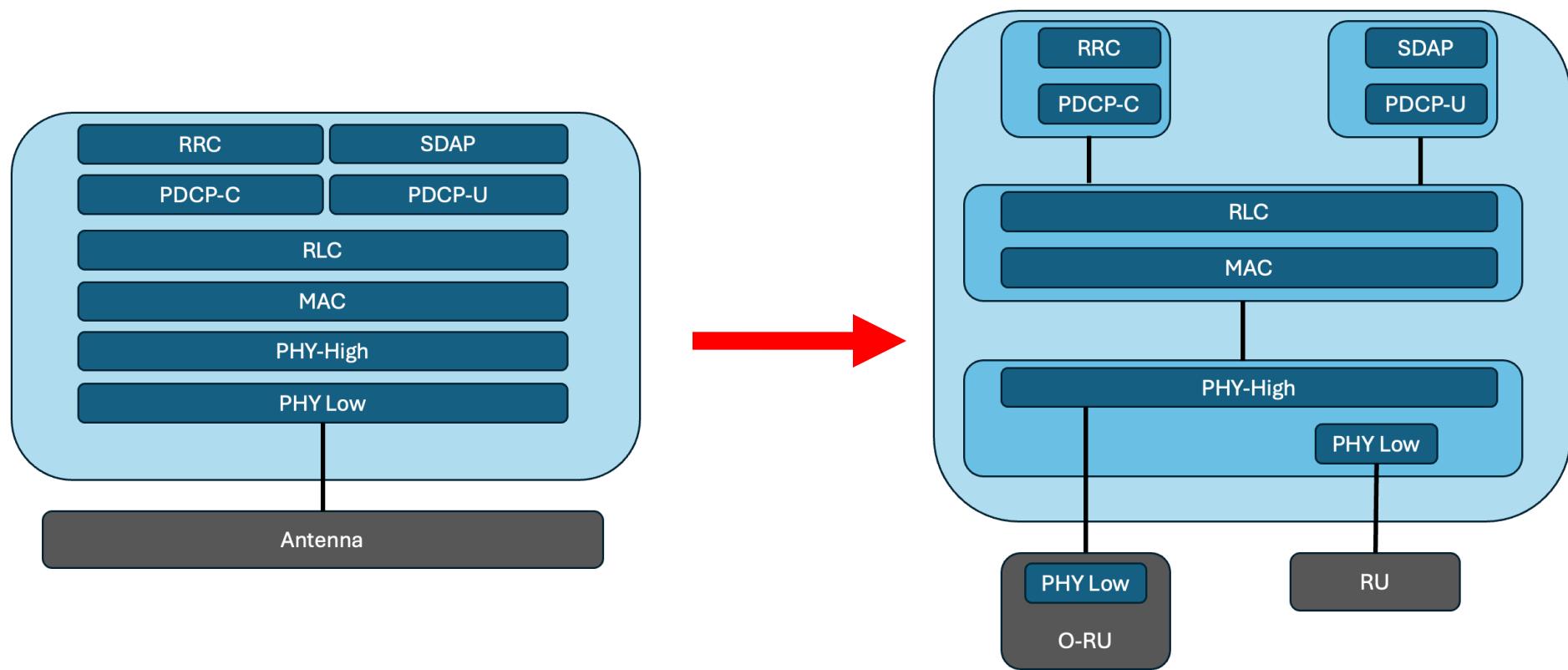
Merim Džaferagić

Date: June 25<sup>th</sup>, 2025

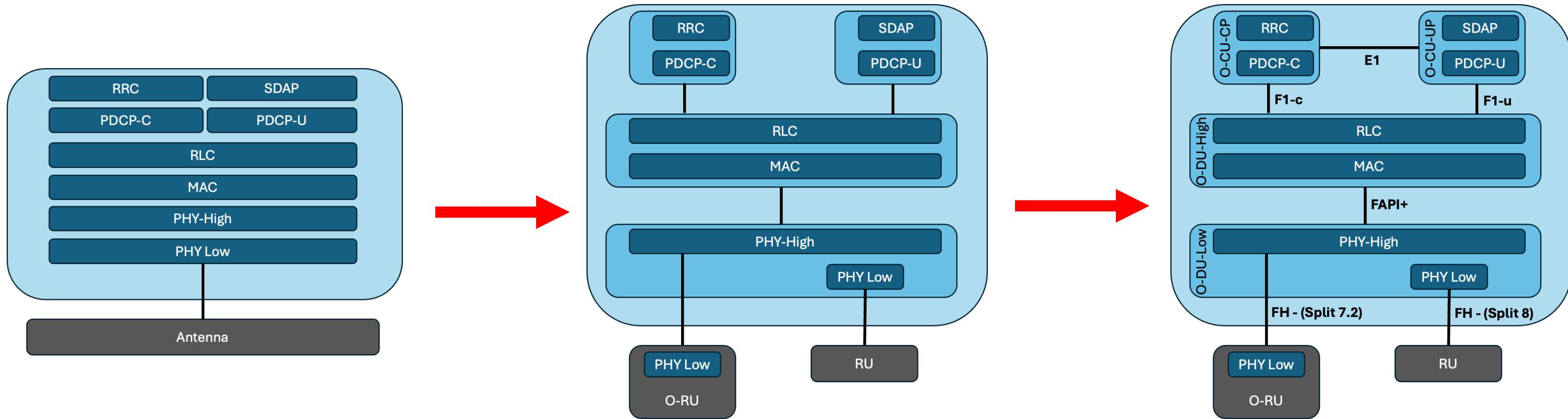
# Traditional BS



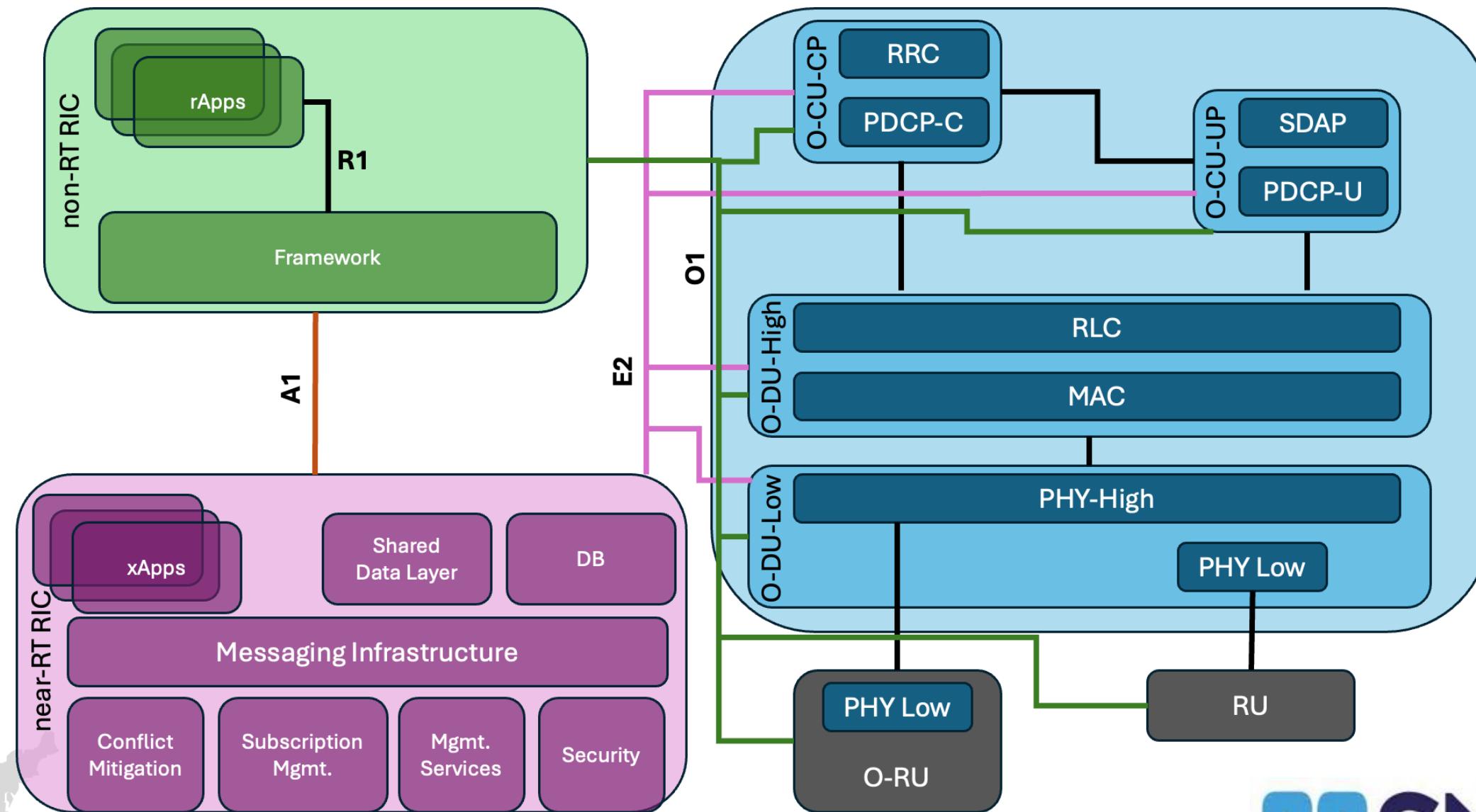
# Traditional BS → Softwarization and Virtualization



# Traditional BS → Softwarization and Virtualization → O-RAN

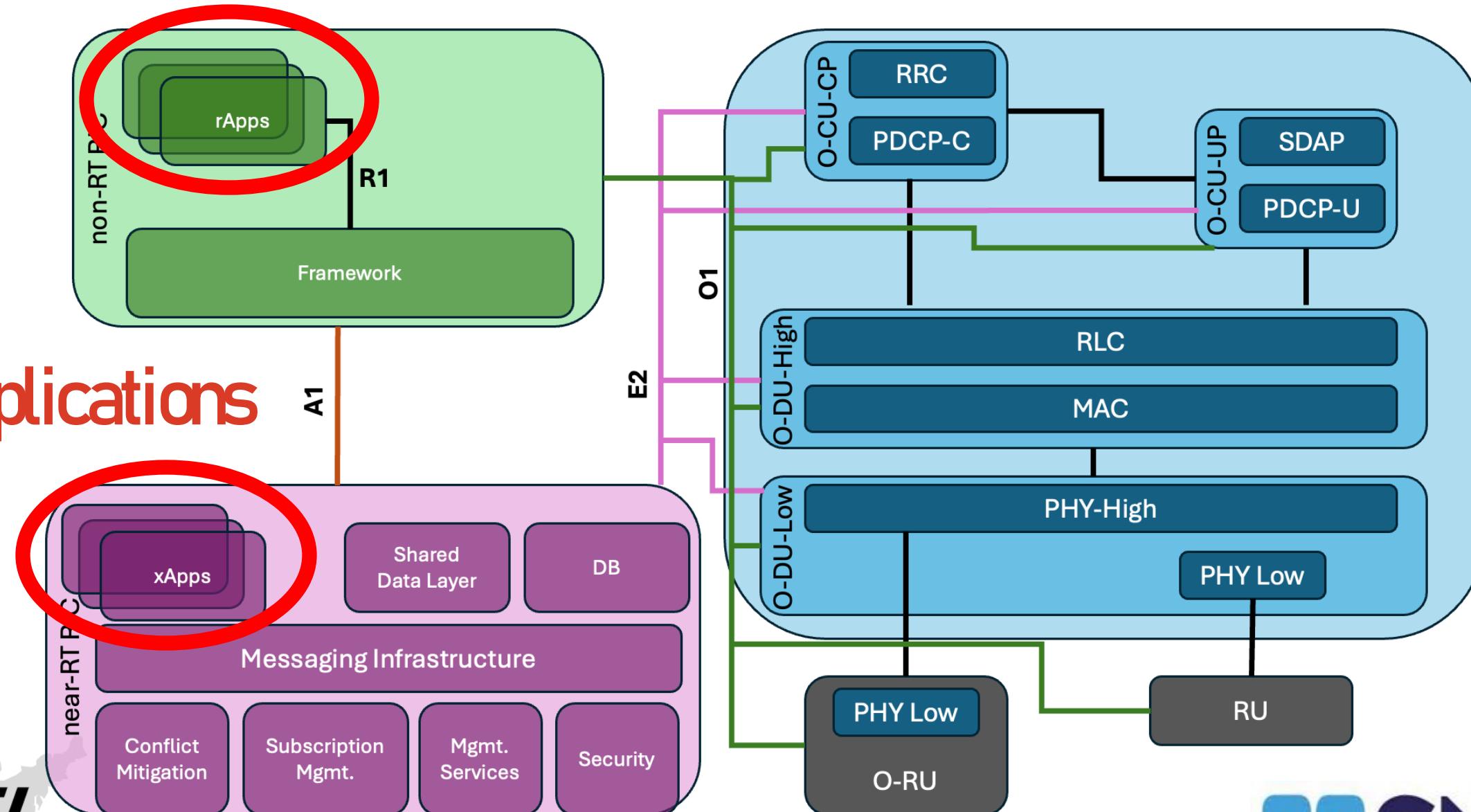


# O-RAN Controllers

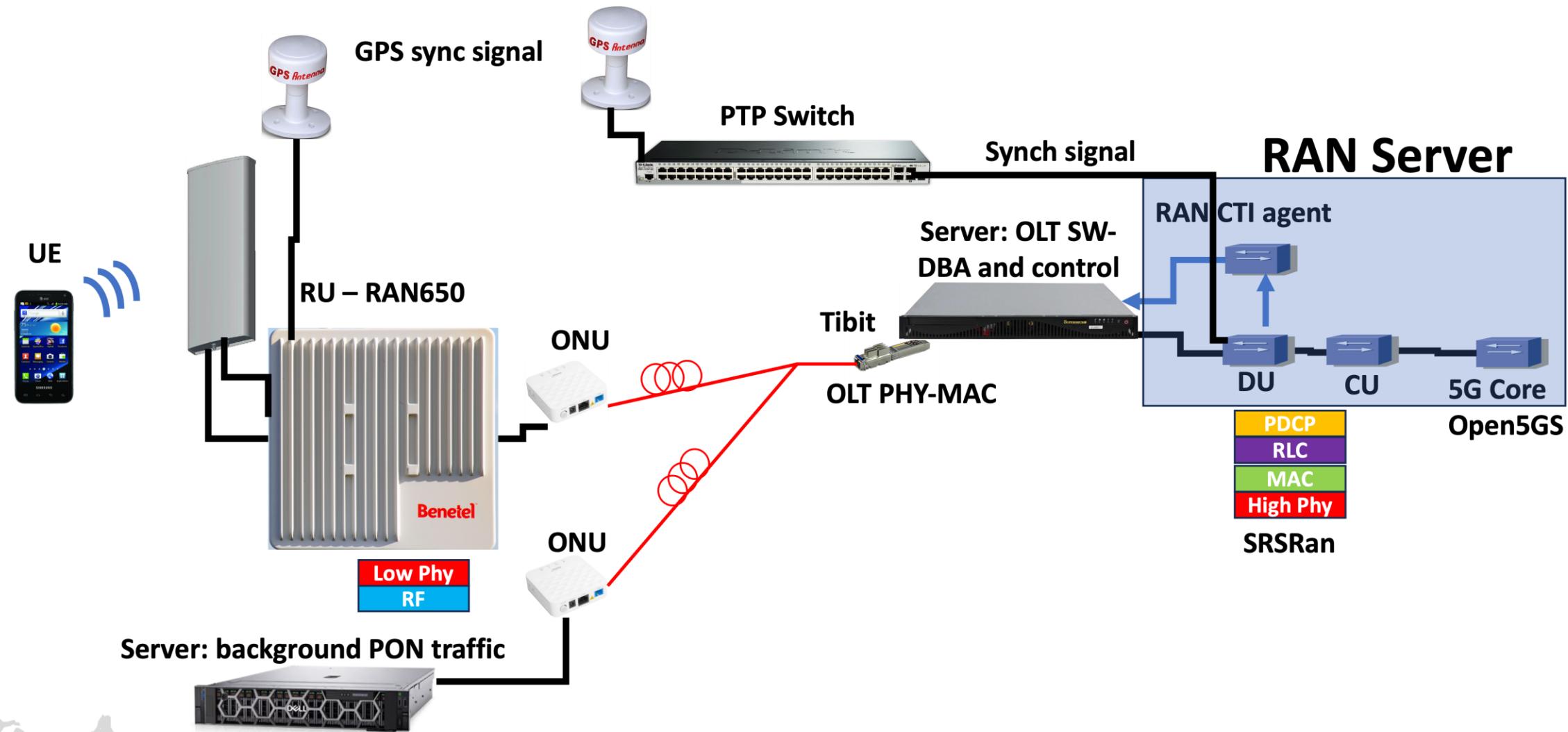


# O-RAN Controllers

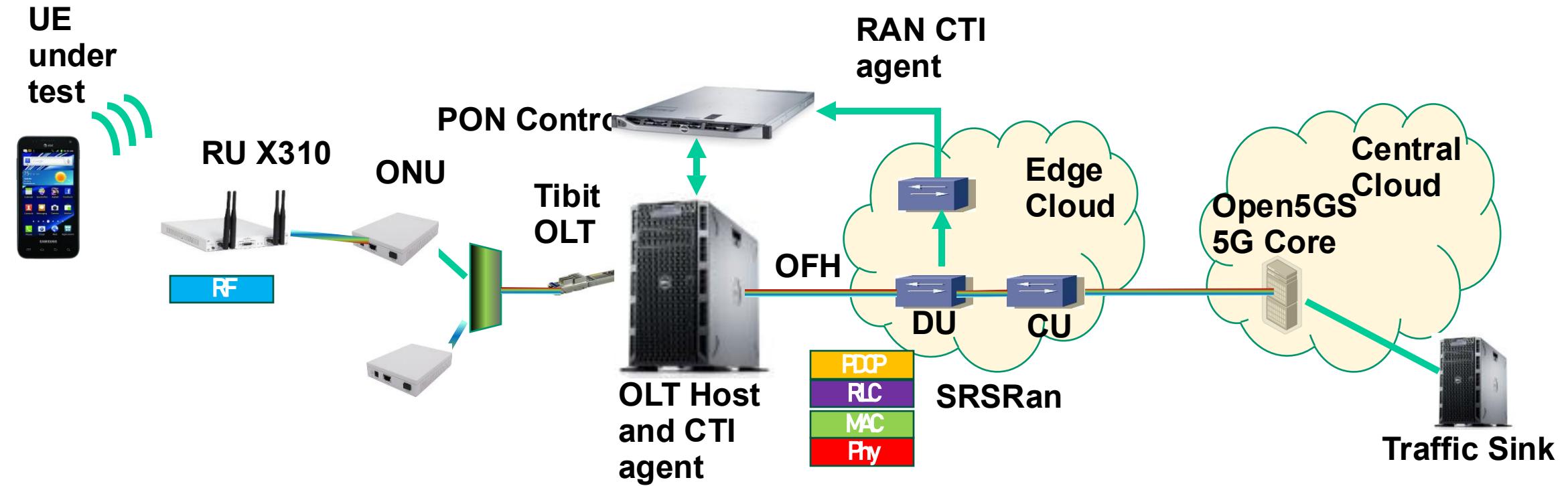
## Control Applications



# Deployment options



# Deployment options

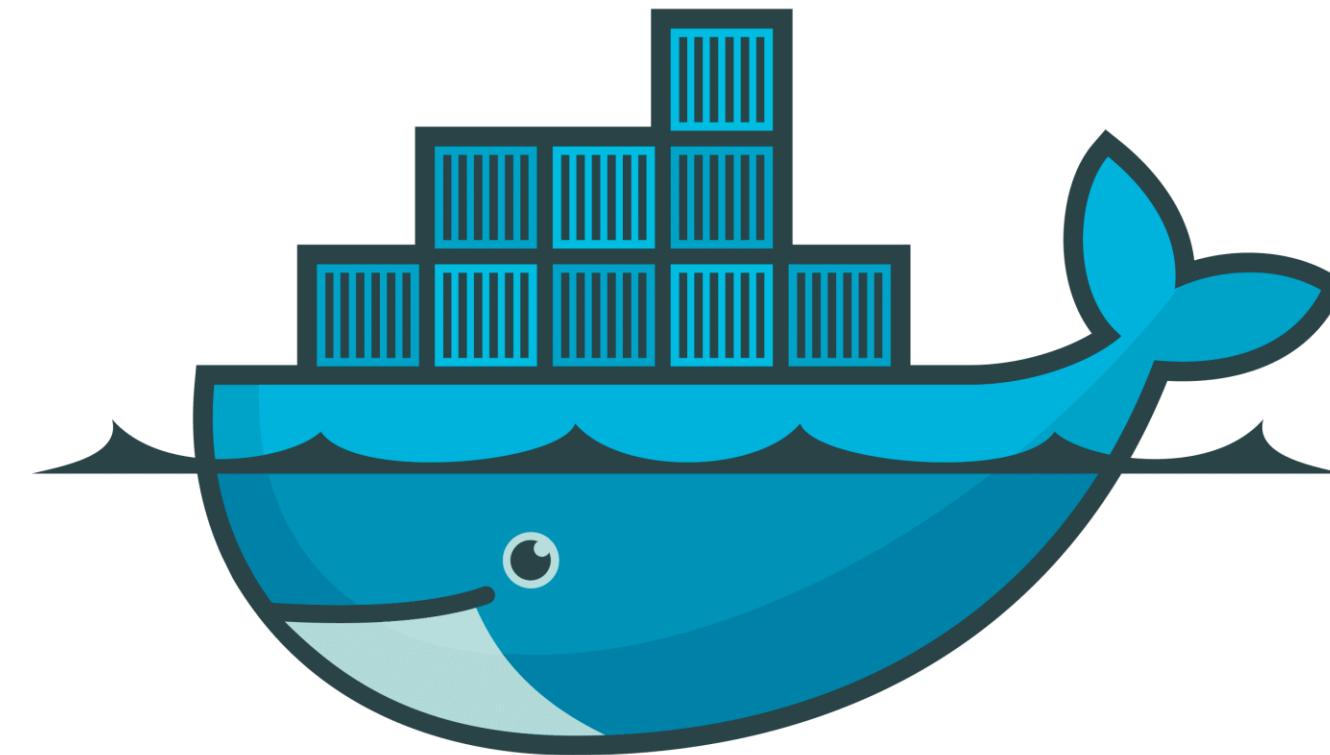


# Why open-source is important

- Innovation and Rapid Experimentation
- Reduced Vendor Lock-in
- Faster Standard Adoption and Ecosystem Growth
- Cost-Effective Testbeds and Education



Docker (SWContainerization) - 



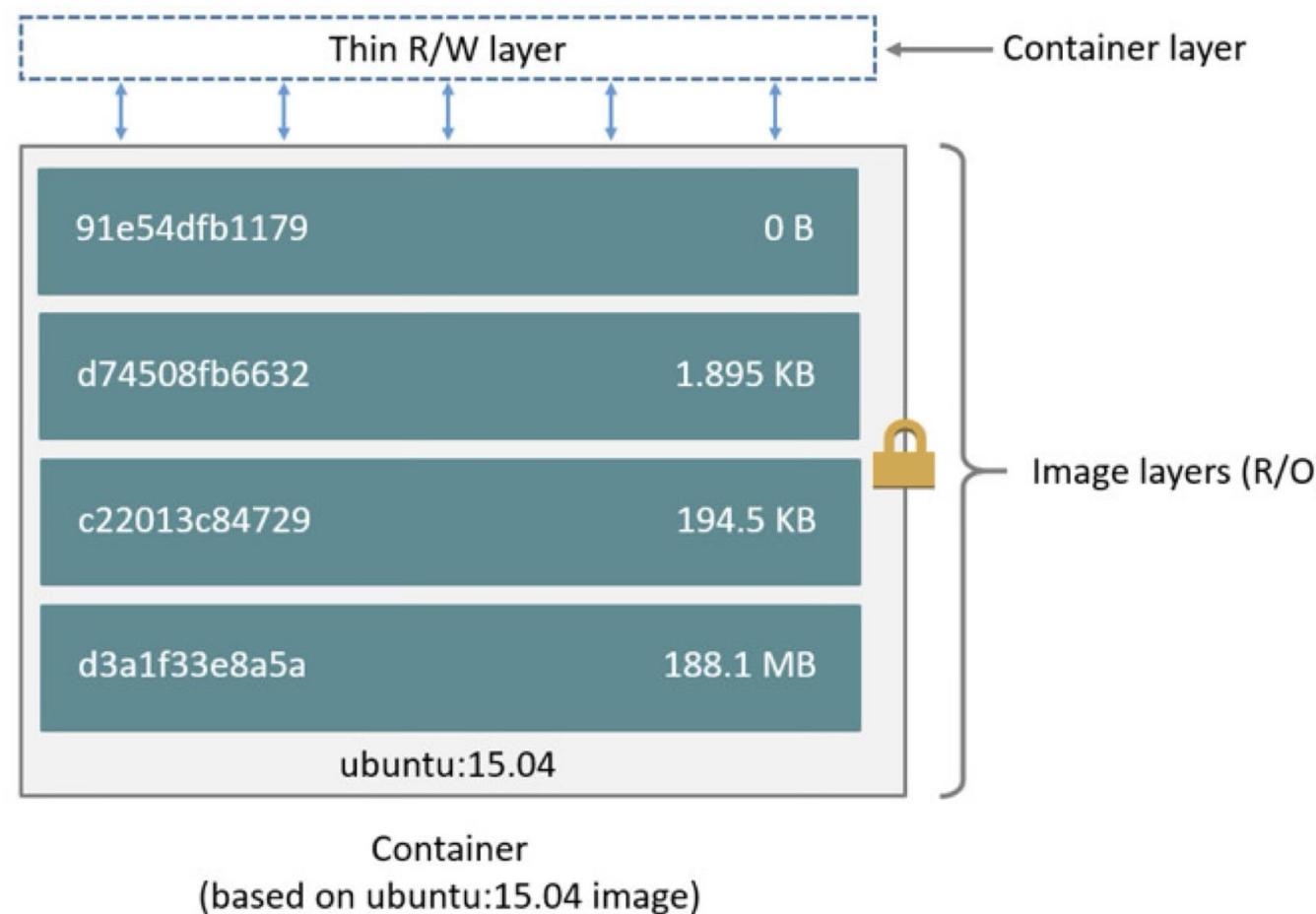
docker



# Docker (SWContainerization) -

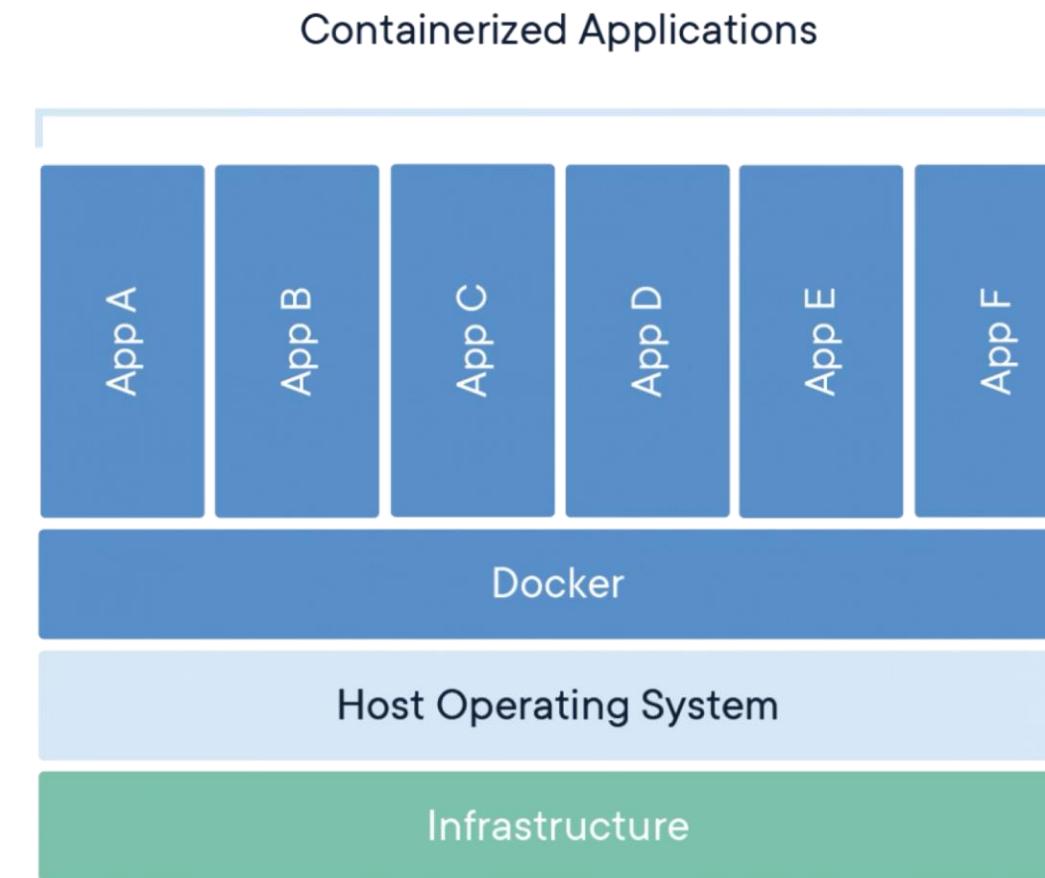
- **What it is:** Docker is a platform for packaging applications into containers—lightweight, isolated environments that include the application and all its dependencies.
- **Why it matters:** In 5G setups, Docker allows running multiple components (like CU DU RLC core) on a single machine or across nodes without interference or dependency conflicts.
- **Benefit:** Faster deployment, easier reproducibility of experiments, and consistent behavior across different environments (e.g., your laptop, testbed server, or cloud).

# Docker Image



- Each layer is only a set of differences from the layer before it.
- When you create a new container, you add a new writable layer on top of the underlying layers.
  - This layer is often called the “container layer”. All changes made to the running container, such as writing new files, modifying existing files, and deleting files, are written to this thin writable container layer.

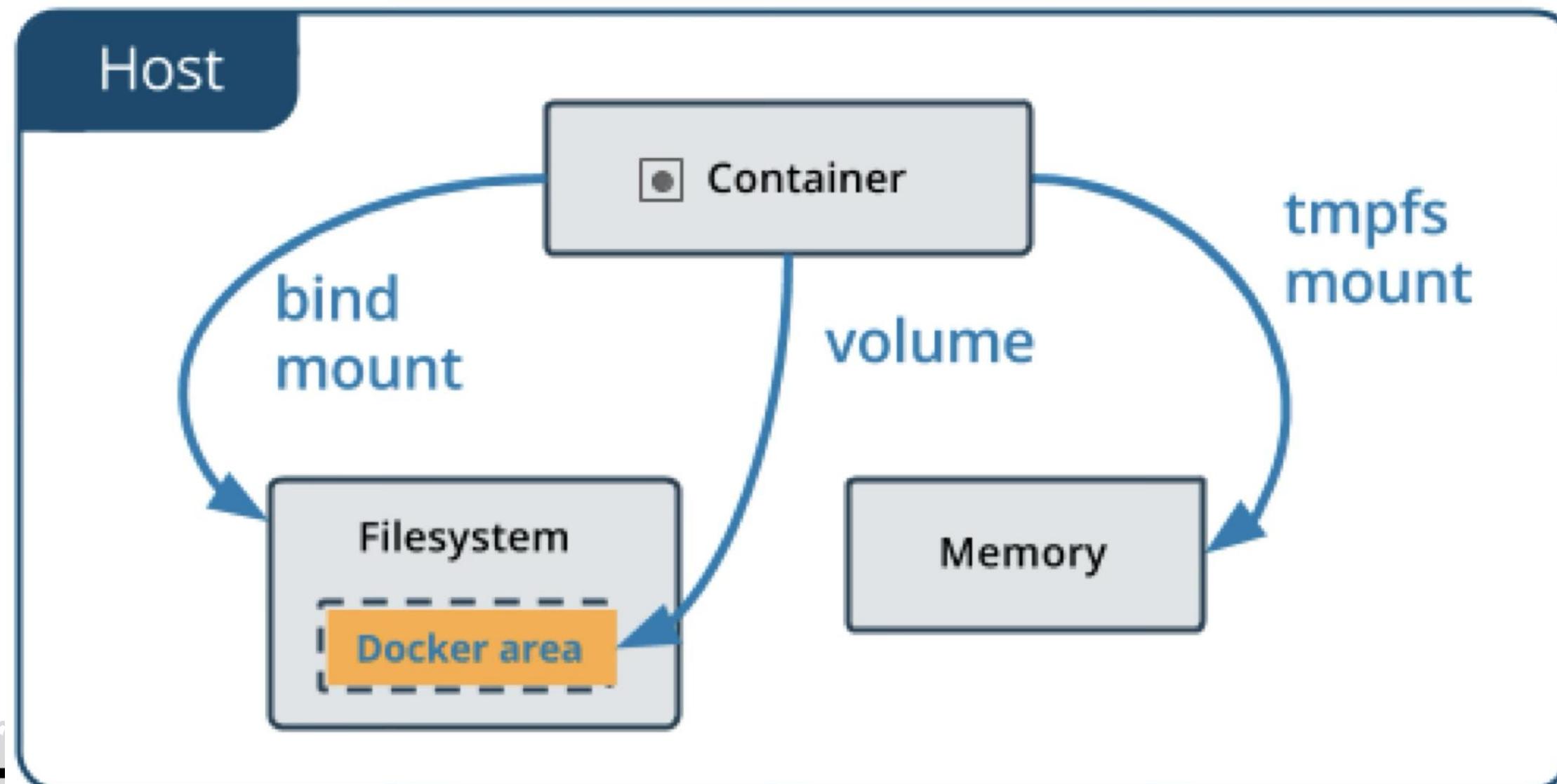
# Docker (cont'd)



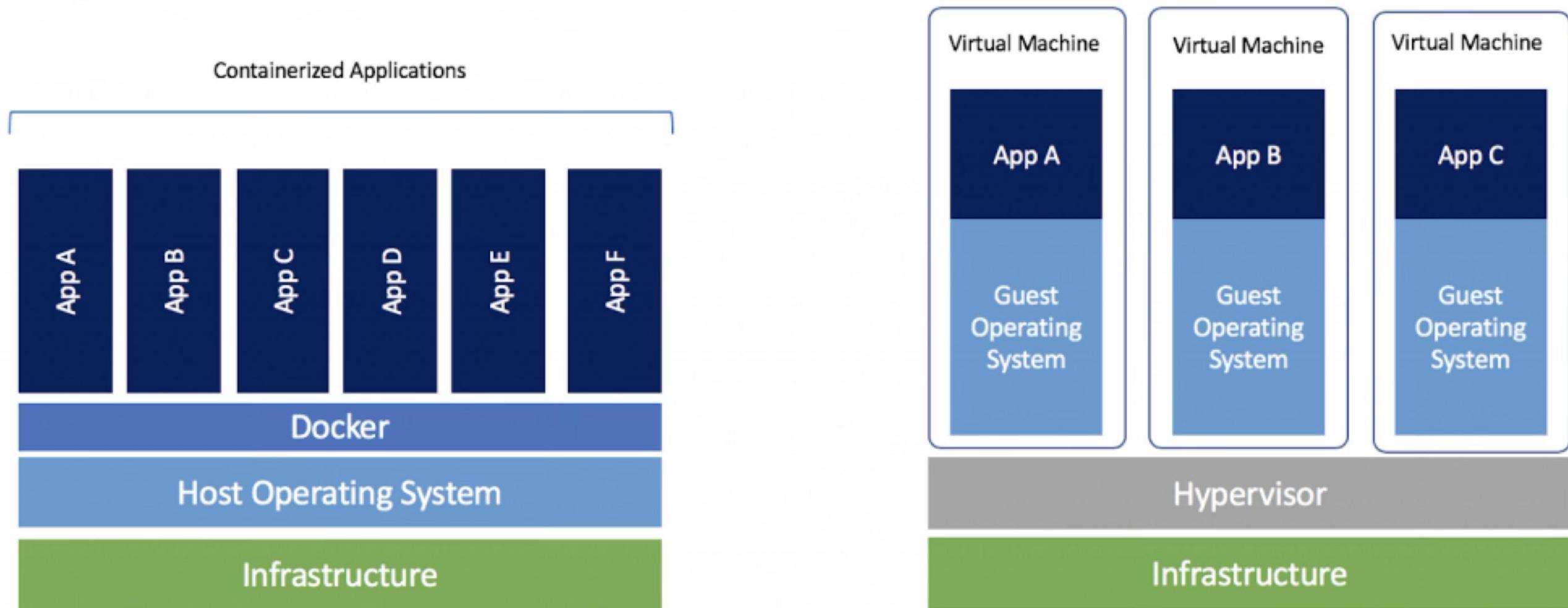
# Docker Volumes

- Docker Volumes are directories created and managed within the docker environment on the host machine.
- They allow us to share the memory between multiple containers, and a container and the host machine.
- Besides Docker Volumes, Docker enables two other ways to access the memory outside of a virtual environment:
  - bind mounts are files or directories from the host machine that are mounted into a container,
  - Tmpfs mounts (Linux only) allow the containers to create files outside of the containers writable layer.  
(Can not be shared between containers).

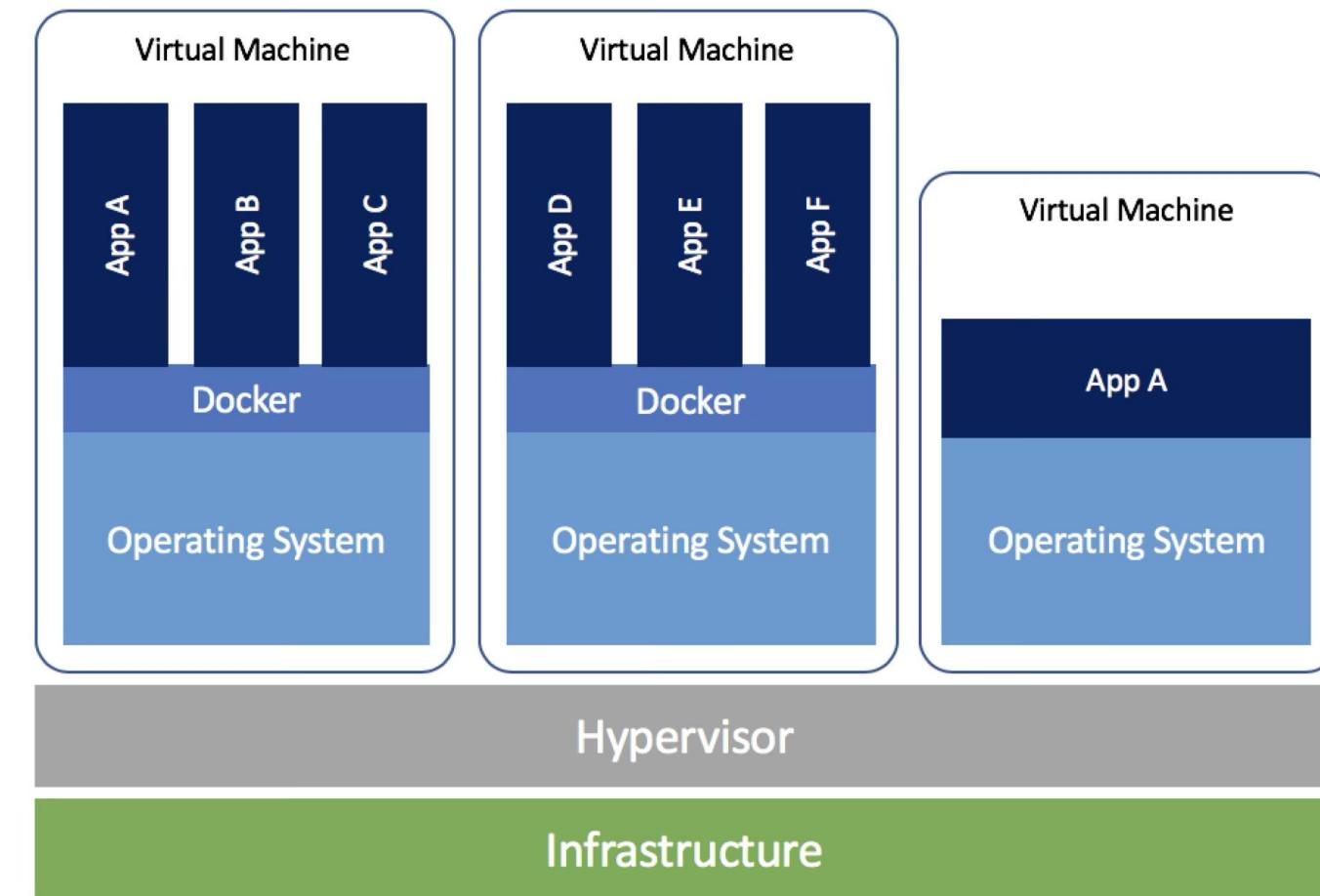
# Docker Volumes (cont'd)



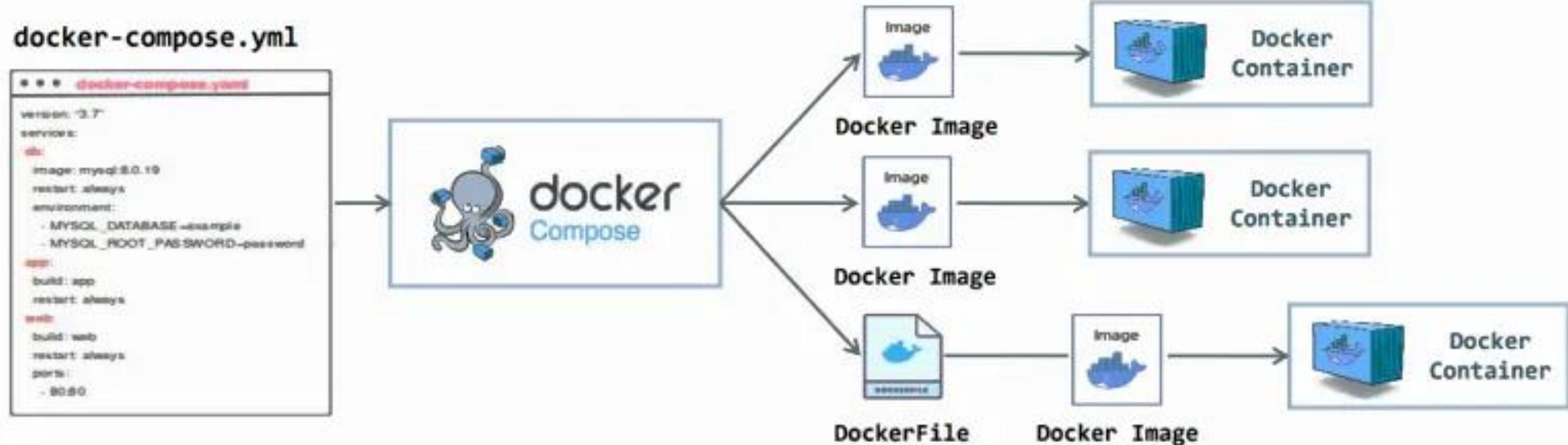
# Containers vs Virtual Machines



# Containers vs Virtual Machines (cont'd)



# Docker Compose



<https://blog.devops.dev/what-and-why-of-docker-compose-dc95314c74b8?gi=57d1d63b5644>

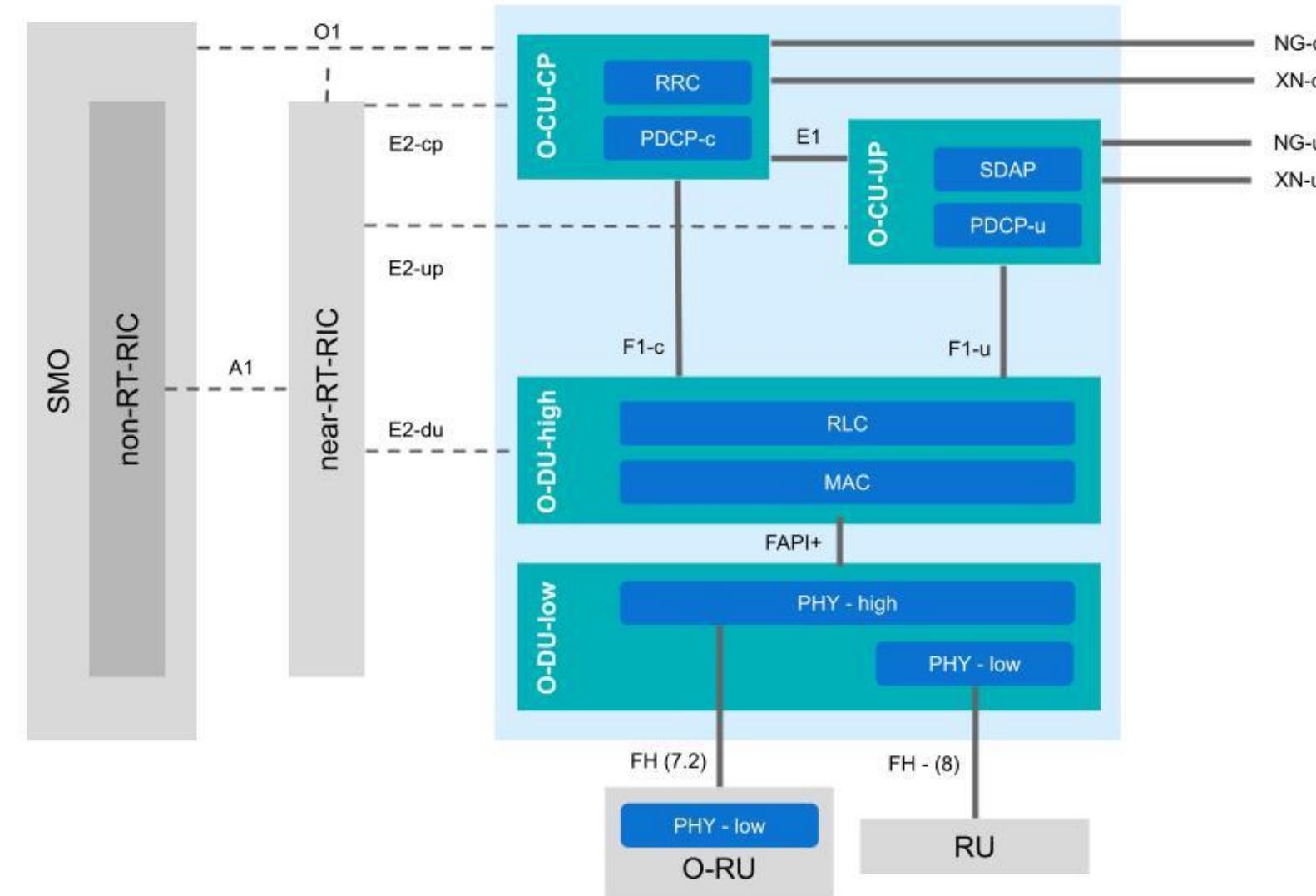


# Tutorial SW

- RAN
  - srsRAN (α, DU dummy-RU)
- Near-RT RIC
  - FlexRIC
  - ORAN-SC RIC
- Core network
  - open5GS



# srsRAN architecture



# srsRAN Configuration Reference

All configuration parameters are presented here below in the following format:

```
parameter: default_value      # Optional/Required TYPE (default value). Parameter description. Format: <format description> OR Supported: <
```

gNB    srsCU    srsDU

```
ran_node_name: cu_cp_01      # Optional TEXT (cu_cp_01). Sets the text ID associated with the gNB. Format: string without spaces.
gnb_id: 411                  # Optional UINT (411). Sets the numerical ID associated with the gNB.
gnb_id_bit_length: 22        # Optional UNIT (22). Sets the bit length of the gnb_id above. Format: integer between [22 - 32].
gnb_cu_up_id: 0              # Optional UINT (0). Sets the gNB CU-UP ID. Supported: [0 - 68719476735].
gnb_du_id: 0                  # Optional UINT (0). Sets the gNB DU ID. Supported: [0 - 68719476735].
```

cells:

```
- pci: 1                      # Optional TEXT. Sets the cell configuration on a per cell basis, overwriting the default configuration set via the configuration file. For more information see the relevant example configuration file: gnb_cell_cfg.
  dl_arfcn: 536020            # Optional TEXT. Override the PCI defined by the default cell in cell_cfg.
- pci: 2                      # Optional TEXT. Set the ARFCN for the above PCI. Note, this must match the value used in the cell_cfg.
  dl_arfcn: 532020            # Optional TEXT. Set the PCI of the second cell.
```

qos:

```
# Optional TEXT. Configures RLC and PDCP radio bearers on a per QoS basis. This can only be set via the configuration file. For more information see the relevant example configuration file: gnb_qos_cfg.
```

srbs:

```
# Optional TEXT. Configures signaling radio bearers. Each list entry should begin with "-".
```

cu\_cp:

```
max_nof_dus: 1                # Optional UINT. Maximum number of DU connections that the CU-CP may accept.
max_nof_cu_ups: 1              # Optional UINT. Maximum number of CU-UP connections that the CU-CP may accept.
max_nof_ues: 1                 # Optional UINT. Maximum number of UEs that the CU-CP may accept.
max_nof_drbs_per_ue: 1         # Optional UINT. Maximum number of DRBs per UE. Supported: [1 - 29].
inactivity_timer: 120          # Optional INT (120). Sets the UE/PDU Session/DRB inactivity timer in seconds. Supported: [1 - 720].
request_pdu_session_timeout: 3 # Optional INT (3). Timeout for the setup of a PDU session after an InitialUeMessage was sent to the AMF.
```

amf:

```
no_core: false                # Optional BOOLEAN. Setting to true allows the gNB to run without a core. Supported: [0, 1].
amf_reconnection_retry_time: 1000 # Optional INT (1000). Sets the time in ms to wait before retrying to connect to the AMF after a disconnection.
addr:
  port: 38412                  # Required TEXT. Sets the IP address or hostname of the AMF. Format: IPV4 or IPV6 IP address.
  bind_addr: 127.0.0.1          # Optional UINT. Sets the AMF port. Format: integer between [20000 - 40000].
  bind_interface: auto         # Optional TEXT. Sets local IP address to bind for N2 interface. Format: IPV4 or IPV6 IP address.
  sctp_rto_initial: 120         # Optional INT. Sets the initial retransmission timeout when creating the SCTP connection.
  sctp_rto_min: 120             # Optional INT. Sets the minimum retransmission timeout for the SCTP connection.
  sctp_rto_max: 500             # Optional INT. Sets the maximum retransmission timeout for the SCTP connection.
  sctp_init_max_attempts: 3     # Optional INT. Sets the maximum retransmission attempts for the initial SCTP connection.
  sctp_max_init_timeo: 500       # Optional INT. Sets the maximum retransmission timeout for the initial SCTP connection.
  sctp_nodelay: false           # Optional BOOLEAN. Setting to true allows the gNB to send SCTP messages as soon as possible without waiting for an acknowledgement.
  supported_tracking_areas: -   # Required TEXT. Sets the list of tracking areas supported by this AMF.
    # Supported TAC #1
```

latest

[https://docs.srsran.com/projects/project/en/latest/user\\_manuals/source/config\\_ref.html](https://docs.srsran.com/projects/project/en/latest/user_manuals/source/config_ref.html)



# Tutorial GitHub



<https://github.com/merimdzaferic/CONVERGE-summer-school>

# Tutorial stages

1. Install docker and docker compose
2. Build and run the docker containers
3. Inspect basic setup
4. Access dashboard with simple measurements
5. Simple configuration file modification
6. Build and run FlexRIC
7. Build and run ORAN-SCRIC



# 1. Install docker and docker compose

- Scripts are available in the GitHub repository
- On your laptop create a new file called `install_docker.sh`
- Make the file executable: `chmod +x install_docker.sh`
- Run the script as root:
  - `sudo bash ./install_docker.sh`
- Check if docker was installed correctly
  - `docker --version`
  - `docker run hello-world`
- `docker-compose --version`
- `docker-compose ls`

## 2. Build and run the docker containers

- Clone the GitHub repository
  - *git clone https://github.com/merimdzferagic/CONVERGE-summer-school.git*
- Change directory
  - *cd CONVERGE-summer-school/srsRAN/docker/*
- Build *docker-compose build*
- Run: *docker-compose up*
- Inspect the running containers: *docker ps*
- Inspect the srsRAN setup

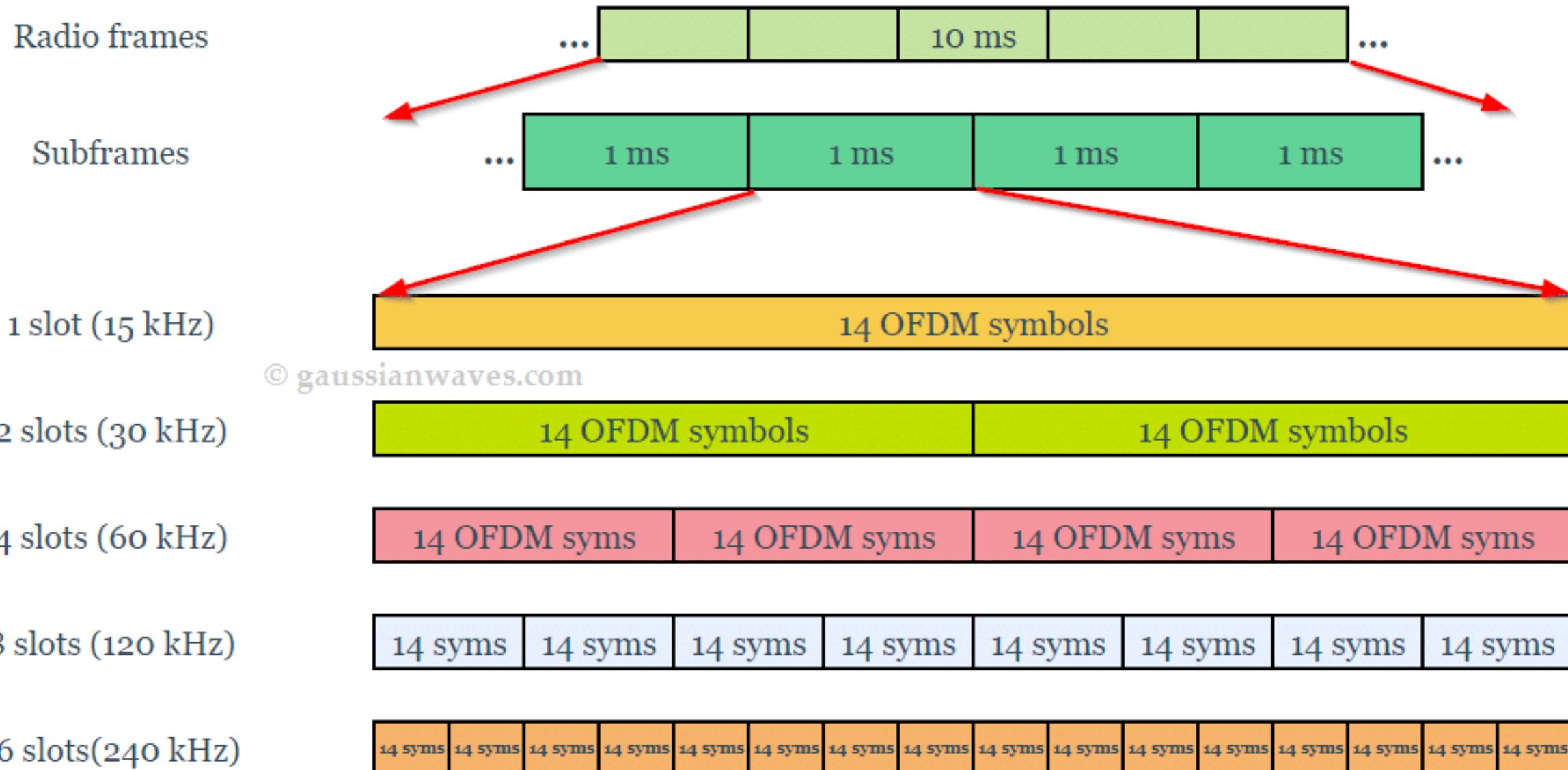


### 3. Inspect basic setup

- Attach to container running srsRANgnb
- Run the console to inspect RAN performance measurements
- Stop all containers, change configuration and inspect again



### 3. Inspect basic setup



### 3. Inspect basic setup

$\mu$	Subcarrier Spacing $\Delta f = 2^\mu \times 15 \text{ kHz}$	Resource block bandwidth $\Delta f \times 12 \text{ subcarriers}$	Frame duration	Slots per frame	Slot duration	OFDM Symbols per slot	Symbol duration	Cyclic prefix
0	15 kHz	180 kHz	10 ms	10	1 ms	14	71.43 $\mu\text{s}$	Normal
1	30 kHz	360 kHz	10 ms	20	500 $\mu\text{s}$	14	35.71 $\mu\text{s}$	Normal
2	60 kHz	720 kHz	10 ms	40	250 $\mu\text{s}$	14	17.86 $\mu\text{s}$	Normal, Extended
3	120 kHz	1.44 MHz	10 ms	80	125 $\mu\text{s}$	14	8.93 $\mu\text{s}$	Normal
4	240 kHz	2.88 MHz	10 ms	160	62.5 $\mu\text{s}$	14	4.46 $\mu\text{s}$	Normal

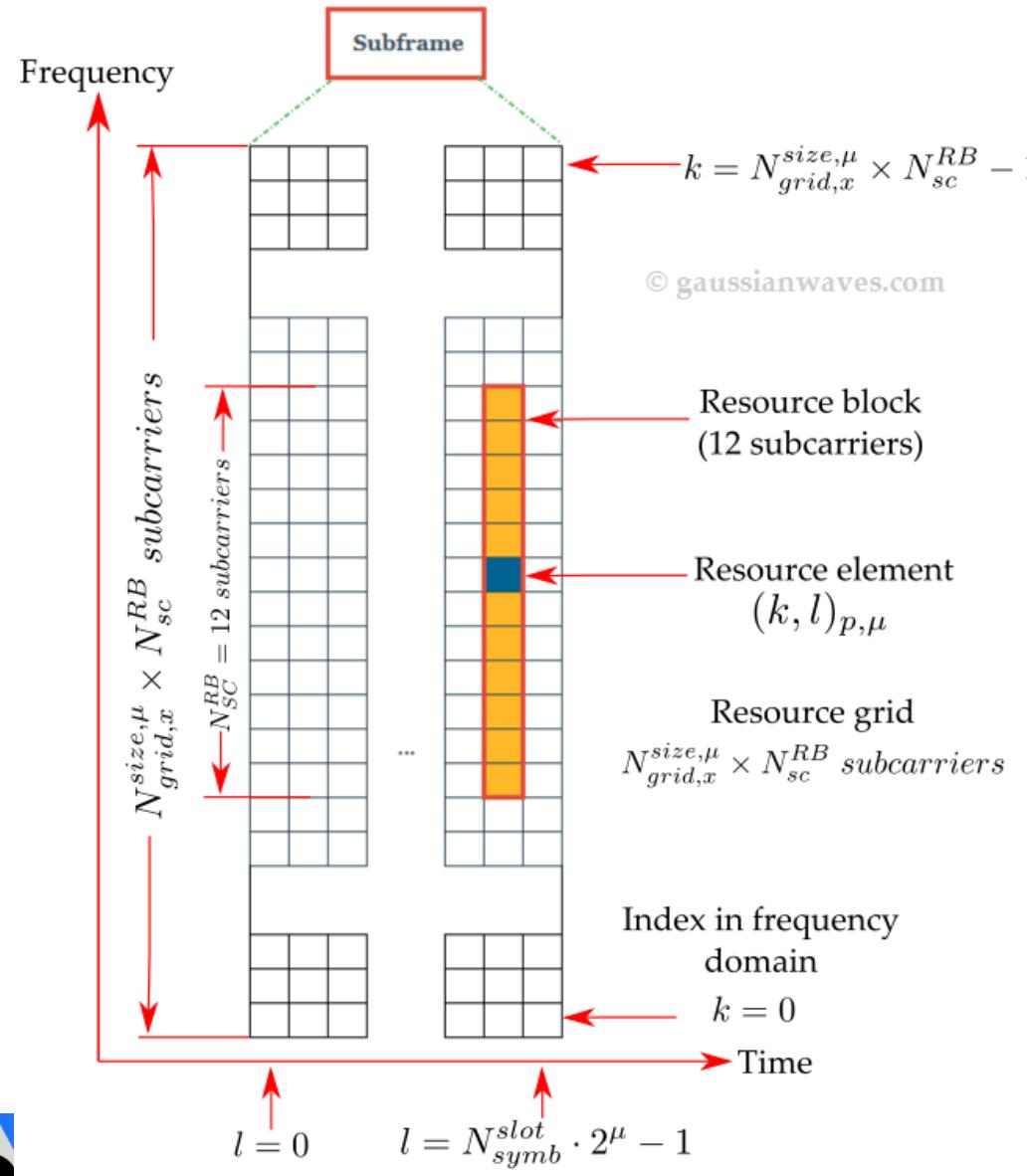
© gaussianwaves.com



<https://www.gaussianwaves.com/2022/02/5g-nr-resource-block/>



# 3. Inspect basic setup



$\mu$	$N_{symb}^{slot}$	$N_{slot}^{subframe, \mu}$	$N_{symb}^{subframe, \mu}$
0	14	1	14
1	14	2	28
2	14	4	56
3	14	8	112
4	14	16	224



# 3. Inspect basic setup

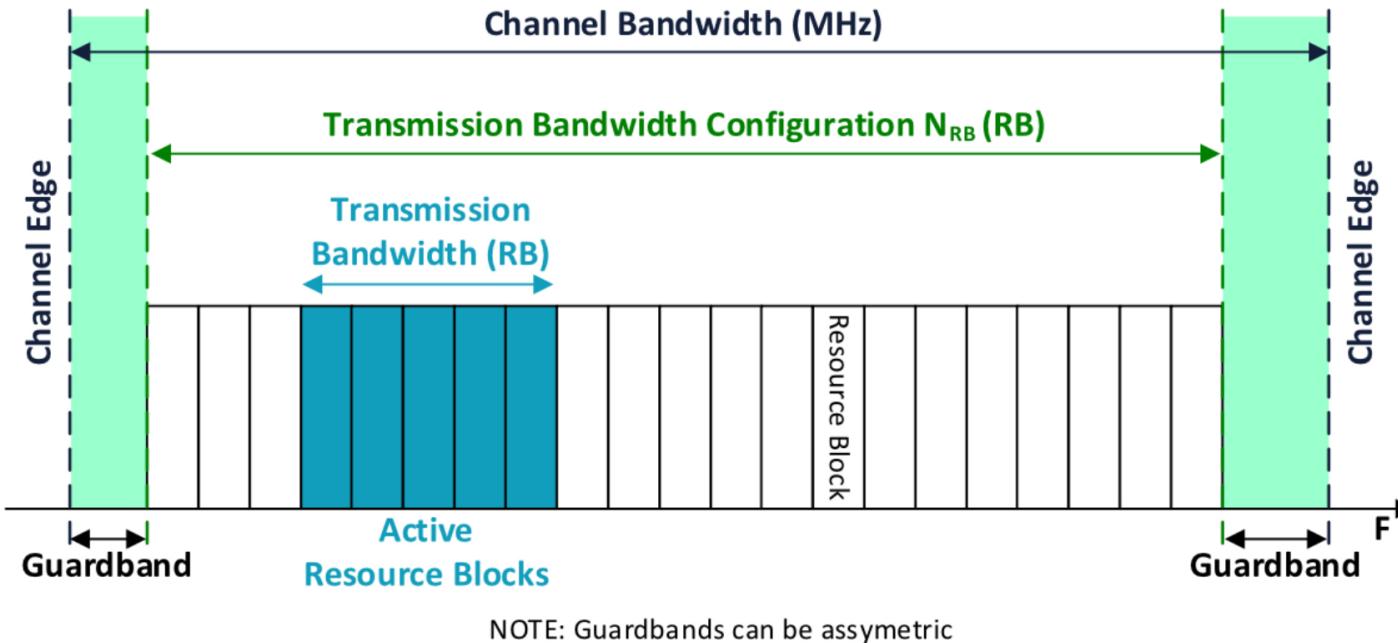


Figure 2-4: Channel bandwidth and the maximum transmission bandwidth configuration [9].

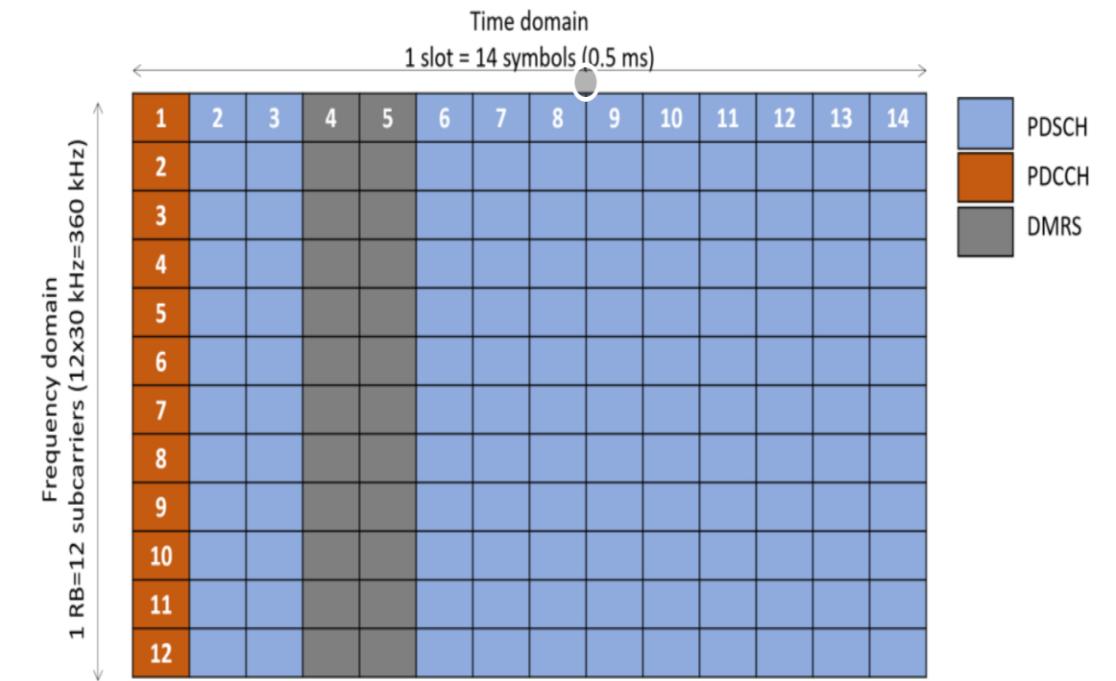


Table 2-2: Maximum number of resource blocks configured for transmission [9].

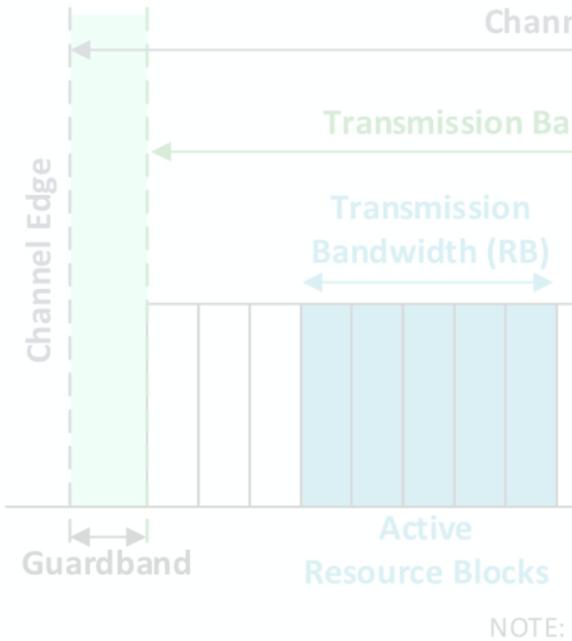
Bandwidth [MHz]	5	10	15	20	25	30	40	50	60	70	80	90	100
$N_{RB}$	11	24	38	51	65	78	106	133	162	189	217	245	273



<https://www.gaussianwaves.com/2022/02/5g-nr-resource-block/>  
And OAI Box Lab sheet



# 3. Inspect basic setup



Considering a 256QAM (Modulation order 8) and code rate 0.925, results in:  
 Spectral efficiency =  $8 \times 0.925 = 7.4063$   
 , which corresponds to MCS Index 27 in 3GPP 38.214 Table 5.1.3.1-2 (see [Table 1-8](#)).  
 Data carried in one Resource Block over one slot =  $132 \times 7.4063 = 978$  bits in 0.5ms

Considering the example where the 5G Bandwidth is 40 MHz which is the maximum bandwidth supported by the OAIBOX 40:  
 Number of Resource Blocks in 40MHz =  $40,000 \text{ KHz} / 360 \text{ KHz} = 111$   
 (106 are used, 4 RBs are used as guard ([Figure 2-4](#))).  
 Total Number of Slots available in one second =  $1,000 \text{ ms} / 0.5 \text{ ms} = 2,000$  slots

Considering a TDD frame with DL/UL Ratio of 4:1  
 Downlink Slots = 1,600 and Uplink Slots = 400

Considering the number of MIMO layers = 1  
 Max DL bit rate =  $(978 \text{ bits} \times 106 \text{ RBs} \times 1600 \text{ slots} \times 1 \text{ layers}) / 10^6 = 166 \text{ Mbps}$   
 Max UL bit rate =  $(978 \text{ bits} \times 106 \text{ RBs} \times 400 \text{ slots} \times 1 \text{ layers}) / 10^6 = 41 \text{ Mbps}$

Table 2-2: Maximum number of RBs per bandwidth

Bandwidth [MHz]	5	10	15	30	45	50	60	75	90	100			
$N_{RB}$	11	24	38	51	65	78	106	133	162	189	217	245	273



PDSCH  
PDCCH  
DMRS

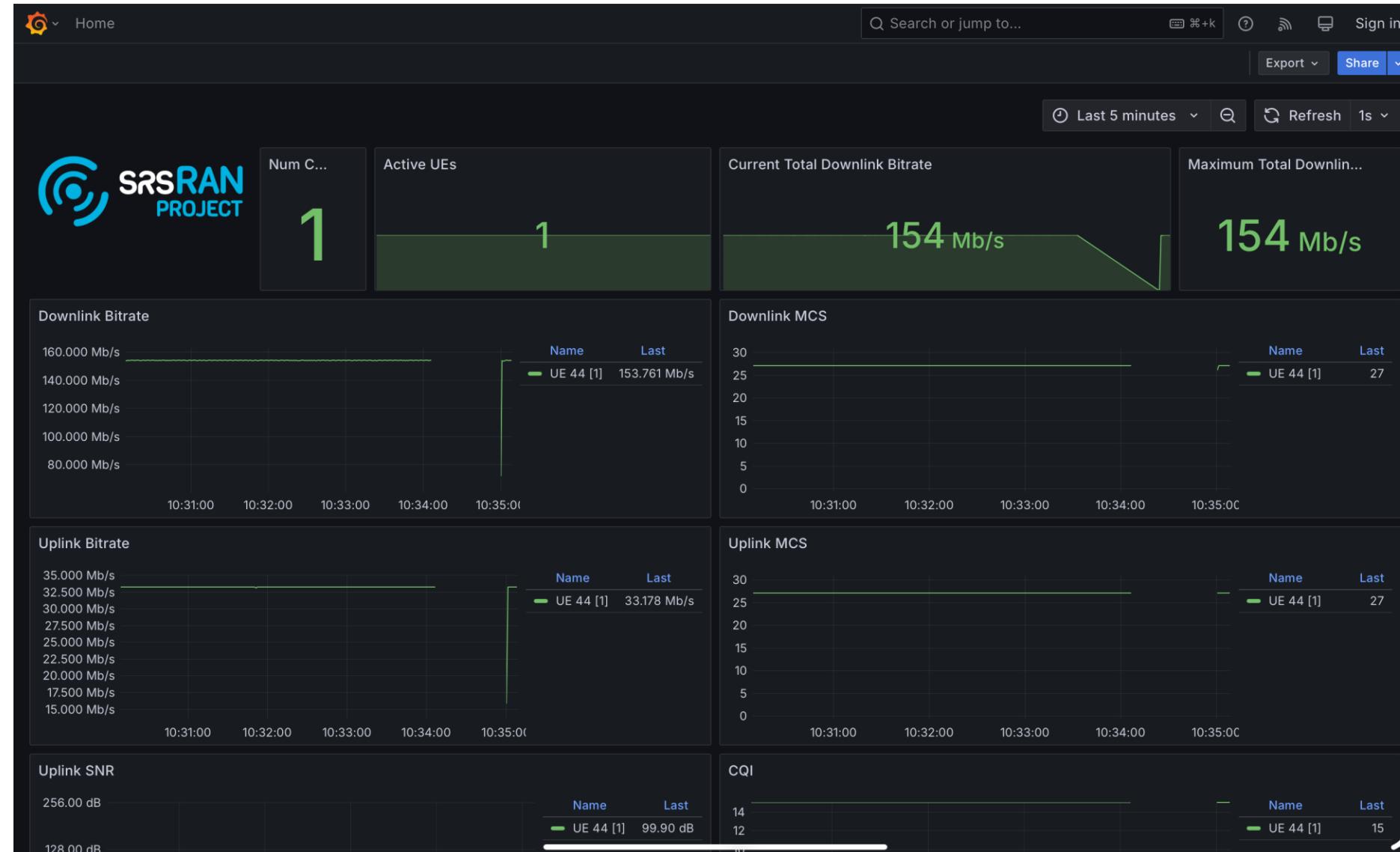
Figure 2-4: Channel bandwidth and the number of RBs



OAI Box Lab sheet



# 4. Access dashboard with simple measurements



# 5. Simple configuration file modification

```
• docker git:(main) ✘ ls
  docker-compose_flex_ric.yml  docker-compose_oranSC.yml  Dockerfile  grafana      open5gs    README.md  srsRAN_conf
  docker-compose_no_ric.yml   docker-compose.yml        flexRIC     metrics_server  oran-sc-ric scripts
• docker git:(main) ✘ ls srsRAN_conf
  gnb_config_flex_ric.yml  gnb_config_no_ric.yml  >nb_config_oranSC.yml testmode.yml
• docker git:(main) ✘
```

- Change:
  - Number of users connected
  - Bandwidth used
  - Subcarrier spacing
  - Max MCS

# 5. Simple configuration file modification

```
◆ docker git:(main) ✘ ls
docker-compose_flex_ric.yml  docker-compo...
docker-compose_no_ric.yml  docker-compo...
◆ docker git:(main) ✘ ls srsRAN_conf
gnb_config_flex_ric.yml  gnb_config_no_...
◆ docker git:(main) ✘
```

```
~ 1 ran_node_name: gnbd_001_001_00019b_0          # Optional TEXT (cu_cp_01). Sets the text ID associated with the gNB. Format: string without spaces
~ 2   .
~ 3   .
~ 4   .
~ 5 gnb_id: 411                                # Optional UINT (411). Sets the numerical ID associated with the gNB.
~ 6 gnb_id_bit_length: 22                         # Optional UNIT (22). Sets the bit length of the gnb_id above. Format: integer between [22 - 32]
~ 7 gnb_cu_up_id: 0                             # Optional UINT (0). Sets the gNB CU-UP ID. Supported: [0 - 68719476735].
~ 8 gnb_du_id: 0
~ 9 cu_cp:
~ 10   amf:
~ 11     addr: 10.53.1.2
~ 12     bind_addr: 10.53.1.3
~ 13     supported_tracking_areas:
~ 14       -
~ 15         tac: 7
~ 16         plmn_list:
~ 17           -
~ 18             plmn: "00101" # PLMN '99940' and TAC '7'
~ 19             tai_slice_support_list:
~ 20               -
~ 21                 sst: 1
~ 22
~ 23
~ 24 ru_dummy:
~ 25   dl_processing_delay: 1
~ 26   time_scaling: 1
~ 27
~ 28
~ 29
~ 30
~ 31
~ 32
~ 33 cell_cfg:
~ 34   dl_arfcn: 663334
~ 35   band: 77
~ 36   channel_bandwidth_MHz: 40
~ 37   common_scs: 30
~ 38   plmn: "00101"
~ 39   tac: 7
~ 40   pci: 1
NORMAL main | ~6 gnb_config_no_ric.yml
All parsers are up-to-date!
0-0 0:nvim*
```

utf-8 | unix | yaml Top 1:1

"nvim srsRAN\_conf/gnb\_" 11:12 22-Jun-25



# 5. Simple configuration file modification

```
• docker git:(main) ✘ ls
docker-compose_flex_ric.yml  docker-compo
docker-compose_no_ric.yml  docker-compo
• docker git:(main) ✘ ls srsRAN_conf
gnb_config_flex_ric.yml  gnb_config_no_!
• docker git:(main) ✘
```

```
1 test_mode:
  1 test_ue:
    2 cqi: 15
    3 nof_ues: 1
    4 pdsch_active: true
    5 pusch_active: true
    6 ri: 1
    7 rnti: 68
```

```
~ 1 ran_node_name: gnbd_001_001_00019b_0      # Optional TEXT (cu_cp_01). Sets the text ID associated with the gNB. Format: string without spaces
~ 1 gnb_id: 411                                # Optional UINT (411). Sets the numerical ID associated with the gNB.
~ 2 gnb_id_bit_length: 22                      # Optional UNIT (22). Sets the bit length of the gnb_id above. Format: integer between [22 - 32]
~ 3 gnb_cu_up_id: 0                            # Optional UINT (0). Sets the gNB CU-UP ID. Supported: [0 - 68719476735].
~ 4 gnb_du_id: 0
5 cu_cp:
6   amf:
7     addr: 10.53.1.2
8     bind_addr: 10.53.1.3
9     supported_tracking_areas:
10    -
11      tac: 7
12      plmn_list:
13        -
14          plmn: "00101" # PLMN '99940' and TAC '7'
15          tai_slice_support_list:
16            -
17              sst: 1
18
19 _dummy:
20   dl_processing_delay: 1
21   time_scaling: 1
22
23 _sdr:
24   device_driver: uhd
25   device_args: type=b200
26   clock: external
27   rate: 30.72
28   tx_gain: 15
29   rx_gain: 15
30
31 _l_cfg:
32   dl_arfcn: 663334
33   band: 77
34   channel_bandwidth_MHz: 40
35   common_scs: 30
36   plmn: "00101"
37   tac: 7
38   pci: 1
39
40 NORMAL main | ~6 gnb_config_no_ric.yml
All parsers are up-to-date!
0-0 0:nvim*
```



utf-8 | unix | yaml Top 1:1

"nvim srsRAN\_conf/gnb\_" 11:12 22-Jun-25

## 6. FlexRIC build and run

- Prepare the correct docker-compose file
  - *cp docker-compose\_flex\_ric.yml docker-compose.yml*
- Build the missing containers (flexric and xappmon)
  - *sudo docker-compose build flexric xappmon*
- Run all services
  - *sudo docker-compose up*



# 6. FlexRC build and run

```
dicty_duration.86400] (./src/nrf/nnrf-handler.c:445)
open5gs_5gc | 06/22 11:50:50.968: [sbi] INFO: [617674b0-4f4e-41f0-9ce6-8d685cc42c0c] Subscription created until 2025-06-23T11:50:50.968205+02:00 [duration:86400,validity:86399.999660,patch:43199.999830] (./lib/sbi/nnrf-handler.c:708)
open5gs_5gc | Open5GS daemon v2.7.0
open5gs_5gc
open5gs_5gc | 06/22 11:50:51.023: [app] INFO: Configuration: 'open5gs-5gc.yml' (./lib/app/ogs-init.c:130)
open5gs_5gc | 06/22 11:50:51.025: [dbi] INFO: MongoDB URI: 'mongodb://127.0.0.1/open5gs' (./lib/dbi/ogs-mongoc.c:130)
open5gs_5gc | 06/22 11:50:51.025: [sbi] INFO: NF Service [nudr-dr] (./lib/sbi/context.c:1812)
open5gs_5gc | 06/22 11:50:51.026: [sbi] INFO: nghttp2_server() [http://127.0.0.20]:7777 (./lib/sbi/nghttp2-server.c:414)
open5gs_5gc | 06/22 11:50:51.026: [app] INFO: UDR initialize...done (./src/udr/app.c:31)
open5gs_5gc | 06/22 11:50:51.027: [nrf] INFO: [617f1d4a-4f4e-41f0-9b24-77a840828337] NF registered [Heartbeat:10s] (./src/nrf/nf-sm.c:190)
open5gs_5gc | 06/22 11:50:51.027: [sbi] INFO: (NRF-notify) NF registered [617f1d4a-4f4e-41f0-9b24-77a840828337:1] (./lib/sbi/nnrf-handler.c:924)
open5gs_5gc | 06/22 11:50:51.027: [sbi] INFO: [UDR] (NRF-notify) NF Profile updated [617f1d4a-4f4e-41f0-9b24-77a840828337:1] (./lib/sbi/nnrf-handler.c:938)
open5gs_5gc | 06/22 11:50:51.028: [sbi] INFO: (NRF-notify) NF registered [617f1d4a-4f4e-41f0-9b24-77a840828337:1] (./lib/sbi/nnrf-handler.c:924)
open5gs_5gc | 06/22 11:50:51.028: [sbi] INFO: [UDR] (NRF-notify) NF Profile updated [617f1d4a-4f4e-41f0-9b24-77a840828337:1] (./lib/sbi/nnrf-handler.c:938)
open5gs_5gc | 06/22 11:50:51.028: [sbi] INFO: (NRF-notify) NF registered [617f1d4a-4f4e-41f0-9b24-77a840828337:1] (./lib/sbi/nnrf-handler.c:924)
open5gs_5gc | 06/22 11:50:51.028: [sbi] INFO: [UDR] (NRF-notify) NF Profile updated [617f1d4a-4f4e-41f0-9b24-77a840828337:1] (./lib/sbi/nnrf-handler.c:938)
open5gs_5gc | 06/22 11:50:51.028: [sbi] INFO: [617f1d4a-4f4e-41f0-9b24-77a840828337] NF registered [Heartbeat:10s] (./lib/sbi/nf-sm.c:221)
open5gs_5gc | 06/22 11:50:51.029: [nrf] INFO: [617fd924-4f4e-41f0-9ce6-8d685cc42c0c] Subscription created until 2025-06-23T11:50:51.029749+02:00 [validity_duration:86400] (./src/nrf/nnrf-handler.c:445)
open5gs_5gc | 06/22 11:50:51.030: [sbi] INFO: [617fd924-4f4e-41f0-9ce6-8d685cc42c0c] Subscription created until 2025-06-23T11:50:51.029749+02:00 [duration:86400,validity:86399.999113,patch:43199.999556] (./lib/sbi/nnrf-handler.c:708)
srsran_gnb | === srsRAN gNB (commit ) ===
```



# 6. FlexRIC build and run

```
◆ docker git:(main) ✘ ls
docker-compose_flex_ric.yml  docker-compose_oranSC.yml  Dockerfile  grafana      open5gs    README.md  srsRAN_conf
docker-compose_no_ric.yml   docker-compose.yml        flexRIC  metrics_server  oran-sc-ric  scripts
◆ docker git:(main) ✘ ls flexRIC
Dockerfile  flexric ric.conf  xapp_mon_e2sm_kpm.conf
◆ docker git:(main) ✘ █
```

# 6. FlexRIC build and run

```
◆ docker git:(main) ✘ ls
docker-compose_flex_ric.yml  docker-compose_oranSC.yml  Dockerfile  grafana      open5gs      README.md  srsRAN_conf
docker-compose_no_ric.yml    docker-compose.yml        flexRIC     metrics_server  oran-sc-ric  scripts
◆ docker git:(main) ✘ ls flexRIC
Dockerfile  flexric ric.conf  xapp_mon_e2sm_kpm.conf
◆ docker git:(main) ✘
```

```
1  SM_DIR = "/usr/local/lib/flexric/"
1
2 # supported name = xApp
3 Name = "xApp"
4 NearRT_RIC_IP = "10.53.1.11"
5 E42_Port = 36422
6
7 # supported name : KPM
8 # supported time (ms): 1000
9 # supported format: 1
10 # supported ran_type: ngran_gNB
11 Sub_ORAN_SM_List = (
12     { name = "KPM", time = 1000,
13       format = 1,
14       ran_type = "ngran_gNB_DU",
15       actions = (
16           { name = "CQI" },
17           { name = "RSRP" }
18       )
19   }
20 )
21
22 xApp_DB = {
23     enable = "OFF"
24     ip = "127.0.0.1"
25     dir = "/tmp/"
26     filename = "testdb"
27     username = "your_username" # if using mysql
28     password = "your_passwd" # if using mysql
29 }
```



# 6. FlexRIC build and run

```
◆ docker git:(main) ✘ ls
docker-compose_flex_ric.yml  docker-compose_oranSC.yml  Dockerfile  grafana    open5gs    README.md  srsRAN_conf
docker-compose_no_ric.yml    docker-compose.yml        flexRIC   metrics_server  oran-sc-ric  scripts
◆ docker git:(main) ✘ ls flexRIC
Dockerfile  flexric ric.conf  xapp_mon_e2sm_kpm.conf
◆ docker git:(main) ✘
```

```
1 $M_DIR = "/usr/local/lib/flexric/"
1
2 # supported name = xApp
3 Name = "xApp"
4 NearRT_RIC_IP = "10.53.1.11"
5 E42_Port = 36422
```

```
1 $M_DIR = "/usr/local/lib/flexric/"
1
2 # supported name = NearRT_RIC, E2_Agent, E2_Proxy_Agent, xApp
3 Name = "NearRT_RIC"
4 NearRT_RIC_IP = "10.53.1.11"
5 E2_Port = 36421
6 E42_Port = 36422
```

```
14      rdn_type = "granule_DU",
15      actions = (
16          { name = "CQI" },
17          { name = "RSRP" }
18      )
19  }
20 )
21
22 xApp_DB = {
23     enable = "OFF"
24     ip = "127.0.0.1"
25     dir = "/tmp/"
26     filename = "testdb"
27     username = "your_username" # if using mysql
28     password = "your_passwd" # if using mysql
29 }
```



# 6. FlexRIC build and run

```
xapp_mon [NEAR-RIC]: Loading SM ID = 144 with def = PDCP_STATS_V0
xapp_mon [xApp]: E42 SETUP-REQUEST tx
flex_ric [iApp]: E42 SETUP-REQUEST rx
flex_ric [iApp]: E42 SETUP-RESPONSE tx
xapp_mon [xApp]: E42 SETUP-RESPONSE rx
xapp_mon [xApp]: xApp ID = 7
xapp_mon [xApp]: Registered E2 Nodes = 1
xapp_mon Connected E2 nodes = 1
xapp_mon Registered node 0 ran func id = 2
xapp_mon [xApp]: reporting period = 1000 [ms]
xapp_mon xApp subscribes RAN Func ID 2 in E2 node idx 0, nb_id 411
xapp_mon E42_RIC_SUBSCRIPTION_REQUEST 31
xapp_mon [xApp]: E42 RIC SUBSCRIPTION REQUEST tx RAN_FUNC_ID 2 RIC_REQ_ID 1
flex_ric [iApp]: SUBSCRIPTION-REQUEST RAN_FUNC_ID 2 RIC_REQ_ID 1 tx
xapp_mon [xApp]: SUBSCRIPTION RESPONSE rx
xapp_mon [xApp]: Successfully subscribed to RAN_FUNC_ID 2
xapp_mon KPM-v3 ind_msg latency = 1002837708331085769 µs from E2-node type 7 ID 411
xapp_mon meas record INTEGER_MEAS_VALUE value 15
xapp_mon meas record INTEGER_MEAS_VALUE value 100
xapp_mon KPM-v3 ind_msg latency = 1003040576817345473 µs from E2-node type 7 ID 411
xapp_mon meas record INTEGER_MEAS_VALUE value 15
xapp_mon meas record INTEGER_MEAS_VALUE value 100
xapp_mon KPM-v3 ind_msg latency = 1000477056868251167 µs from E2-node type 7 ID 411
xapp_mon meas record INTEGER_MEAS_VALUE value 15
xapp_mon meas record INTEGER_MEAS_VALUE value 100
xapp_mon KPM-v3 ind_msg latency = 1007688109390738011 µs from E2-node type 7 ID 411
xapp_mon meas record INTEGER_MEAS_VALUE value 15
xapp_mon meas record INTEGER_MEAS_VALUE value 100
```



## 7. ORAN-SC RC build and run

- Prepare the correct docker-compose file
  - *cp docker-compose\_oranSC.yml docker-compose.yml*
- Build the missing containers (*dbaas rtmgr\_sim submgr e2term appmgr e2mgr python\_xapp\_runner*)
  - *sub docker-compose build dbaas rtmgr\_sim submgr e2term appmgr e2mgr python\_xapp\_runner*
- Run all services
  - *sub docker-compose up*

# 7. ORAN-SC RIC build and run

```
◆ docker git:(main) ✘ ls
docker-compose_flex_ric.yml  docker-compose_oranSC.yml  Dockerfile  grafana  open5gs  README.md  srsRAN_conf
docker-compose_no_ric.yml    docker-compose.yml        flexRIC   metrics_server  oran-sc-ric scripts
◆ docker git:(main) ✘ ls oran-sc-ric
COPYRIGHT  create_ric_config_files.sh  docker-compose.yml  e2-agents  LICENSE  README.md  ric  xApps
◆ docker git:(main) ✘ ls oran-sc-ric/xApps/
python
◆ docker git:(main) ✘ ls oran-sc-ric/xApps/python
kpm_mon_xapp.py  lib  simple_mon_xapp.py  simple_rc_xapp.py  simple_xapp.py
◆ docker git:(main) ✘
```



# 7. ORAN-SC RIC build and run

```
◆ docker git:(main) ✘ ls
docker-compose_flex_ric.yml  docker-compose_oran
docker-compose_no_ric.yml    docker-compose.yml
◆ docker git:(main) ✘ ls oran-sc-ric
COPYRIGHT  create_ric_config_files.sh  docker-co
◆ docker git:(main) ✘ ls oran-sc-ric/xApps/
python
◆ docker git:(main) ✘ ls oran-sc-ric/xApps/python
kpm_mon_xapp.py  lib  simple_mon_xapp.py  simple
◆ docker git:(main) ✘
```

```
1  #!/usr/bin/env python3
2
3  import argparse
4  import signal
5  from lib.xAppBase import xAppBase
6
7  class MyXapp(xAppBase):
8      def __init__(self, config, http_server_port, rmr_port):
9          super(MyXapp, self).__init__(config, http_server_port, rmr_port)
10
11     def my_subscription_callback(self, e2_agent_id, subscription_id, indication_hdr, indication_msg, kpm_report_style, ue_id):
12         if kpm_report_style == 2:
13             print("\nRIC Indication Received from {} for Subscription ID: {}, KPM Report Style: {}, UE ID: {}".format(e2_agent_id, subscription_id, kpm_report_style, ue_id))
14         else:
15             print("\nRIC Indication Received from {} for Subscription ID: {}, KPM Report Style: {}".format(e2_agent_id, subscription_id, kpm_report_style))
16
17         indication_hdr = self.e2sm_kpm.extract_hdr_info(indication_hdr)
18         meas_data = self.e2sm_kpm.extract_meas_data(indication_msg)
19
20         print("E2SM_KPM RIC Indication Content:")
21         print("-ColletStartTime: ", indication_hdr['colletStartTime'])
22         print("-Measurements Data:")
23
24         granulPeriod = meas_data.get("granulPeriod", None)
25         if granulPeriod is not None:
26             print("-granulPeriod: {}".format(granulPeriod))
27
28         if kpm_report_style in [1,2]:
29             for metric_name, value in meas_data["measData"].items():
30                 print("--Metric: {}, Value: {}".format(metric_name, value))
31
32         else:
33             for ue_id, ue_meas_data in meas_data["ueMeasData"].items():
34                 print("--UE_id: {}".format(ue_id))
35                 granulPeriod = ue_meas_data.get("granulPeriod", None)
36                 if granulPeriod is not None:
37                     print("---granulPeriod: {}".format(granulPeriod))
38
39                 for metric_name, value in ue_meas_data["measData"].items():
40
41 NORMAL main kpm_mon_xapp.py
All parsers are up-to-date!
0-0 0:nvim*
```

utf-8 | unix | python Top 1:1

"nvim oran-sc-ric/xApp" 11:22 22-Jun-25



# 7. ORAN-SC RIC build and run

```
open5gs_5gc | 06/22 12:15:26.729: [sbi] INFO: (NRF-notify) NF registered [d1155fc2-4f51-41f0-973e-abdf91128289:1] (./lib/sbi/nnrf-handler.c:924)
open5gs_5gc | 06/22 12:15:26.729: [sbi] INFO: [BSF] (NRF-notify) NF Profile updated [d1155fc2-4f51-41f0-973e-abdf91128289:1] (./lib/sbi/nnrf-handler.c:938)
open5gs_5gc | 06/22 12:15:26.730: [sbi] INFO: (NRF-notify) NF registered [d1155fc2-4f51-41f0-973e-abdf91128289:1] (./lib/sbi/nnrf-handler.c:924)
open5gs_5gc | 06/22 12:15:26.730: [sbi] INFO: [BSF] (NRF-notify) NF Profile updated [d1155fc2-4f51-41f0-973e-abdf91128289:1] (./lib/sbi/nnrf-handler.c:938)
open5gs_5gc | 06/22 12:15:26.730: [sbi] INFO: [d1155fc2-4f51-41f0-973e-abdf91128289] NF registered [Heartbeat:10s] (./lib/sbi/nf-sm.c:221)
open5gs_5gc | 06/22 12:15:26.730: [nrf] INFO: [d115f8c4-4f51-41f0-bad1-f5940dfb41ba] Subscription created until 2025-06-23T12:15:26.730895+02:00 [validity_duration:86400] (./src/nrf/nnrf-handler.c:445)
open5gs_5gc | 06/22 12:15:26.731: [sbi] INFO: [d115f8c4-4f51-41f0-bad1-f5940dfb41ba] Subscription created until 2025-06-23T12:15:26.730895+02:00 [duration:86400,validity:86399.999472,patch:43199.999736] (./lib/sbi/nnrf-handler.c:708)
open5gs_5gc | Open5GS daemon v2.7.0
open5gs_5gc |
open5gs_5gc | 06/22 12:15:26.783: [app] INFO: Configuration: 'open5gs-5gc.yml' (./lib/app/ogs-init.c:130)
open5gs_5gc | 06/22 12:15:26.785: [dbi] INFO: MongoDB URI: 'mongodb://127.0.0.1/open5gs' (./lib/dbi/ogs-mongoc.c:130)
open5gs_5gc | 06/22 12:15:26.785: [sbi] INFO: NF Service [nudr-dr] (./lib/sbi/context.c:1812)
open5gs_5gc | 06/22 12:15:26.786: [sbi] INFO: nghttp2_server() [http://127.0.0.20]:7777 (./lib/sbi/nghttp2-server.c:414)
open5gs_5gc | 06/22 12:15:26.786: [app] INFO: UDR initialize...done (./src/udr/app.c:31)
open5gs_5gc | 06/22 12:15:26.787: [nrf] INFO: [d11e36ba-4f51-41f0-acd3-e33e1b1fe644] NF registered [Heartbeat:10s] (./src/nrf/nf-sm.c:190)
open5gs_5gc | 06/22 12:15:26.787: [sbi] INFO: (NRF-notify) NF registered [d11e36ba-4f51-41f0-acd3-e33e1b1fe644:1] (./lib/sbi/nnrf-handler.c:924)
open5gs_5gc | 06/22 12:15:26.787: [sbi] INFO: [UDR] (NRF-notify) NF Profile updated [d11e36ba-4f51-41f0-acd3-e33e1b1fe644:1] (./lib/sbi/nnrf-handler.c:938)
open5gs_5gc | 06/22 12:15:26.788: [sbi] INFO: (NRF-notify) NF registered [d11e36ba-4f51-41f0-acd3-e33e1b1fe644:1] (./lib/sbi/nnrf-handler.c:924)
open5gs_5gc | 06/22 12:15:26.788: [sbi] INFO: [UDR] (NRF-notify) NF Profile updated [d11e36ba-4f51-41f0-acd3-e33e1b1fe644:1] (./lib/sbi/nnrf-handler.c:938)
open5gs_5gc | 06/22 12:15:26.788: [sbi] INFO: (NRF-notify) NF registered [d11e36ba-4f51-41f0-acd3-e33e1b1fe644:1] (./lib/sbi/nnrf-handler.c:924)
open5gs_5gc | 06/22 12:15:26.788: [sbi] INFO: [UDR] (NRF-notify) NF Profile updated [d11e36ba-4f51-41f0-acd3-e33e1b1fe644:1] (./lib/sbi/nnrf-handler.c:938)
open5gs_5gc | 06/22 12:15:26.788: [sbi] INFO: [d11e36ba-4f51-41f0-acd3-e33e1b1fe644] NF registered [Heartbeat:10s] (./lib/sbi/nf-sm.c:221)
open5gs_5gc | 06/22 12:15:26.789: [nrf] INFO: [d11ee7b8-4f51-41f0-bad1-f5940dfb41ba] Subscription created until 2025-06-23T12:15:26.789437+02:00 [validity_duration:86400] (./src/nrf/nnrf-handler.c:445)
open5gs_5gc | 06/22 12:15:26.789: [sbi] INFO: [d11ee7b8-4f51-41f0-bad1-f5940dfb41ba] Subscription created until 2025-06-23T12:15:26.789437+02:00 [duration:86400,validity:86399.999473,patch:43199.999736] (./lib/sbi/nnrf-handler.c:708)
srsran_gnb |
srsran_gnb | ---- srsRAN gNB (commit ) ----
srsran_gnb |
srsran_gnb | Cell pci=1, bw=40 MHz, 1T1R, dl_arfcn=663334 (n77), dl_freq=3950.01 MHz, dl_ss_arfcn=662304, ul_freq=3950.01 MHz
srsran_gnb |
srsran_gnb | E2AP: Connection to Near-RT-RIC on 10.0.2.10:36421 completed
srsran_gnb | ---- gNB started ----
srsran_gnb | Type <h> to view help
ric_rtmgr_sim | 2025/06/22 10:15:30 POST /ric/v1/handles/associate-ran-to-e2t body: [{"E2TAddress":"10.0.2.10:38000","ranNamelist":["gnbd_001_001_00019b_0"]}] elapsed: 97.169µs
0-0 0:sudo*                                     "sudo docker-compose u" 11:15 22-Jun-25
```



# 7. ORAN-SC RIC build and run

```
• docker git:(main) ✘ sudo docker compose exec python_xapp_runner ./kpm_mon_xapp.py --metrics=CQI,RSRP --kpm_report_style=1
[sudo] password for merim:
1750587356408 7/RMR [INFO] ric message routing library on SI95 p=4562 mv=3 flg=00 id=a (f447e29 4.9.4 built: Dec 13 2023)
Subscribe to E2 node ID: gnbd_001_001_00019b_0, RAN func: e2sm_kpm, Report Style: 1, metrics: ['CQI', 'RSRP']
Successfully subscribed with Subscription ID: 2yrKfjRKVW7fwNoYdVq1HaU3AQI
Received Subscription ID to E2EventInstanceId mapping: 2yrKfjRKVW7fwNoYdVq1HaU3AQI -> 1
{'response': 'OK', 'status': 200, 'payload': '{}', 'ctype': 'application/json', 'attachment': None, 'mode': 'plain'}
10.0.2.13 - - [22/Jun/2025 10:15:56] "POST /ric/v1/subscriptions/response HTTP/1.1" 200 -

RIC Indication Received from gnbd_001_001_00019b_0 for Subscription ID: 1, KPM Report Style: 1
E2SM_KPM RIC Indication Content:
-CollectStartTime: 2025-06-22 10:15:57
-Measurements Data:
--granulPeriod: 1000
--Metric: CQI, Value: [15]
--Metric: RSRP, Value: [100]

RIC Indication Received from gnbd_001_001_00019b_0 for Subscription ID: 1, KPM Report Style: 1
E2SM_KPM RIC Indication Content:
-CollectStartTime: 2025-06-22 10:15:58
-Measurements Data:
--granulPeriod: 1000
--Metric: CQI, Value: [15]
--Metric: RSRP, Value: [100]

RIC Indication Received from gnbd_001_001_00019b_0 for Subscription ID: 1, KPM Report Style: 1
E2SM_KPM RIC Indication Content:
-CollectStartTime: 2025-06-22 10:15:59
-Measurements Data:
--granulPeriod: 1000
--Metric: CQI, Value: [15]
--Metric: RSRP, Value: [100]
```



# 7. ORAN-SC RC build and run

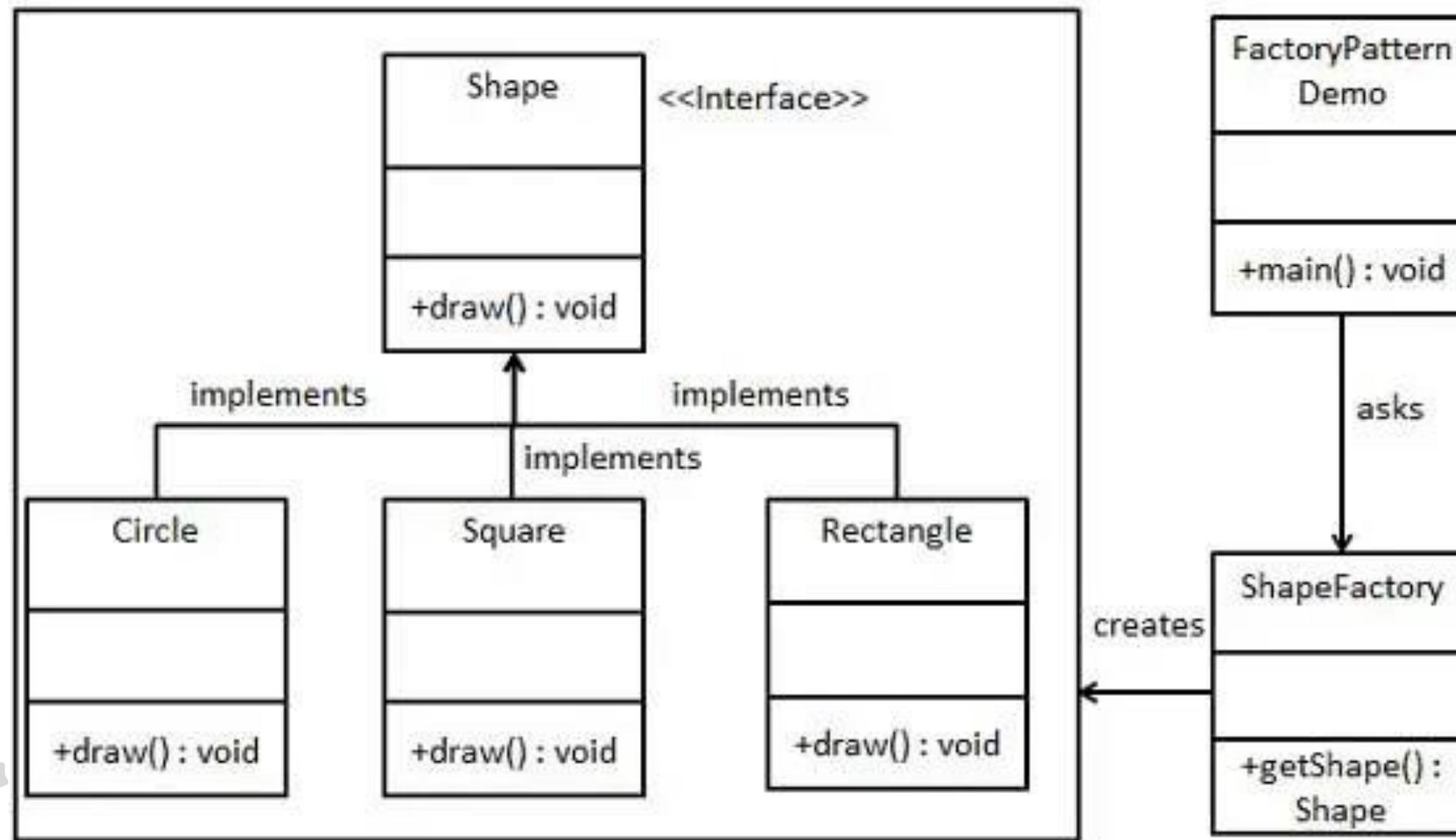
```
ler.c:938)
open5gs_5gc | 06/22 12:15:26.730: [sbi] INFO: (NRF-notify) NF registered [d1155fc2-4f51-41f0-973e-abdf91128289:1] (./lib/sbi/nnrf-handler.c:924)
open5gs_5gc | 06/22 12:15:26.730: [sbi] INFO: [BSF] (NRF-notify) NF Profile updated [d1155fc2-4f51-41f0-973e-abdf91128289:1] (./lib/sbi/nnrf-hand
ler.c:938)
open5gs_5gc | 06/22 12:15:26.730: [sbi] INFO: [d1155fc2-4f51-41f0-973e-abdf91128289] NF registered [Heartbeat:10s] (./lib/sbi/nf-sm.c:221)
open5gs_5gc | 06/22 12:15:26.730: [nrf] INFO: [d115f8c4-4f51-41f0-bad1-f5940dfb41ba] Subscription created until 2025-06-23T12:15:26.730895+02:00 [
validity_duration:86400] (./src/nrf/nnrf-handler.c:445)
open5gs_5gc | 06/22 12:15:26.731: [sbi] INFO: [d115f8c4-4f51-41f0-bad1-f5940dfb41ba] Subscription created until 2025-06-23T12:15:26.730895+02:00 [
duration:86400,validity:86399.999472,patch:43199.999736] (./lib/sbi/nnrf-handler.c:708)
open5gs_5gc | Open5GS daemon v2.7.0
open5gs_5gc
open5gs_5gc | 06/22 12:15:26.783: [app] INFO: Configuration: 'open5gs-5gc.yml' (./lib/app/ogs-init.c:130)
open5gs_5gc | 06/22 12:15:26.785: [dbi] INFO: MongoDB URI: 'mongodb://127.0.0.1/open5gs' (./lib/dbi/ogs-mongoc.c:130)
open5gs_5gc | 06/22 12:15:26.785: [sbi] INFO: NF Service [nudr-dr] (./lib/sbi/context.c:1812)
open5gs_5gc | 06/22 12:15:26.786: [sbi] INFO: nghttp2_server() [http://127.0.0.20]:7777 (./lib/sbi/nghttp2-server.c:414)
open5gs_5gc | 06/22 12:15:26.786: [app] INFO: UDR initialize...done (./src/udr/app.c:31)
open5gs_5gc | 06/22 12:15:26.787: [nrf] INFO: [d11e36ba-4f51-41f0-acd3-e33e1b1fe644] NF registered [Heartbeat:10s] (./src/nrf/nf-sm.c:190)
open5gs_5gc | 06/22 12:15:26.787: [sbi] INFO: (NRF-notify) NF registered [d11e36ba-4f51-41f0-acd3-e33e1b1fe644:1] (./lib/sbi/nnrf-handler.c:924)
open5gs_5gc | 06/22 12:15:26.787: [sbi] INFO: [UDR] (NRF-notify) NF Profile updated [d11e36ba-4f51-41f0-acd3-e33e1b1fe644:1] (./lib/sbi/nnrf-hand
ler.c:938)
open5gs_5gc | 06/22 12:15:26.788: [sbi] INFO: (NRF-notify) NF registered [d11e36ba-4f51-41f0-acd3-e33e1b1fe644:1] (./lib/sbi/nnrf-handler.c:924)
open5gs_5gc | 06/22 12:15:26.788: [sbi] INFO: [UDR] (NRF-notify) NF Profile updated [d11e36ba-4f51-41f0-acd3-e33e1b1fe644:1] (./lib/sbi/nnrf-hand
ler.c:938)
open5gs_5gc | 06/22 12:15:26.788: [sbi] INFO: (NRF-notify) NF registered [d11e36ba-4f51-41f0-acd3-e33e1b1fe644:1] (./lib/sbi/nnrf-handler.c:924)
open5gs_5gc | 06/22 12:15:26.788: [sbi] INFO: [UDR] (NRF-notify) NF Profile updated [d11e36ba-4f51-41f0-acd3-e33e1b1fe644:1] (./lib/sbi/nnrf-hand
ler.c:938)
open5gs_5gc | 06/22 12:15:26.788: [sbi] INFO: [d11e36ba-4f51-41f0-acd3-e33e1b1fe644] NF registered [Heartbeat:10s] (./lib/sbi/nf-sm.c:221)
open5gs_5gc | 06/22 12:15:26.789: [nrf] INFO: [d11ee7b8-4f51-41f0-bad1-f5940dfb41ba] Subscription created until 2025-06-23T12:15:26.789437+02:00 [
validity_duration:86400] (./src/nrf/nnrf-handler.c:445)
open5gs_5gc | 06/22 12:15:26.789: [sbi] INFO: [d11ee7b8-4f51-41f0-bad1-f5940dfb41ba] Subscription created until 2025-06-23T12:15:26.789437+02:00 [
duration:86400,validity:86399.999473,patch:43199.999736] (./lib/sbi/nnrf-handler.c:708)
srsran_gnb
srsran_gnb | === srsRAN gNB (commit ) ===
srsran_gnb
srsran_gnb | Cell pci=1, bw=40 MHz, 1T1R, dl_arfcn=663334 (n77), dl_freq=3950.01 MHz, dl_ssrb_arfcn=662304, ul_freq=3950.01 MHz
srsran_gnb
srsran_gnb | E2AP: Connection to Near-RT-RIC on 10.0.2.10:36421 completed
srsran_gnb | === gNB started ===
srsran_gnb | Type <h> to view help
ric_rtmgr_sim | 2025/06/22 10:15:30 POST /ric/v1/handles/associate-ran-to-e2t body: [{"E2TAddress": "10.0.2.10:38000", "ranNamelist": ["gnbd_001_001_0019b_0"]}] elapsed: 97.169µs
ric_rtmgr_sim | 2025/06/22 10:15:56 POST /ric/v1/handles/xapp-subscription-handle body: {"address": "10.0.2.20", "port": 4562, "subscription_id": 1} ela
psed: 40.692µs
0-0 0:sudo*Z
"sudo docker-compose u" 11:16 22-Jun-25
```



Thank you!

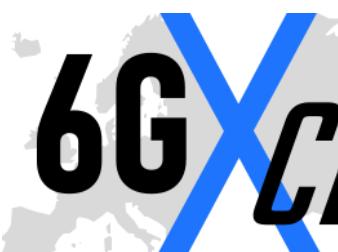


# srsRAN code organization



# srsRAN code organization - (*apps/gnb/CMakeLists.txt*)

```
| gnb.cpp * | flexible_o_du_application_u...h * | dynamic_o_du_application_....cpp * | CMakeLists.txt * |
20 target_link_libraries(gnb_base PRIVATE
19     srsran_app_services
18     srsran_f1u_connector
17     srsran_ngap
16     ngap_asn1
15     srsran_e2
14     e2ap_asn1
13     srsran_gateway
12     srsran_pcap
11     srsran_support
10     srsran_versioning
9     srsran_f1c_gateway
8     srsran_e1_gateway
7     srsran_o_cu_cp_app_unit
6     srsran_o_cu_up_app_unit
5 )
4
3 if (DPDK_FOUND)
2   add_definitions(-DDPDK_FOUND)
1   target_link_libraries(gnb_base PRIVATE hal_dpdk)
49 endif (DPDK_FOUND)
1
2 target_include_directories(gnb_base PRIVATE ${CMAKE_SOURCE_DIR} ${CMAKE_SOURCE_DIR}/external)
3
4 # Dynamic gnb targets with dynamic split.
5 add_executable(gnb gnb.cpp)
6 add_backward(gnb)
7 target_include_directories(gnb PRIVATE ${CMAKE_SOURCE_DIR})
8 install(TARGETS gnb RUNTIME)
9 target_link_libraries(gnb PRIVATE gnb_base srsran_flexible_o_du_split_dynamic)
10 notify_binary_target(gnb)
11
12 # Add the targets for the rest of the splits.
13 set(EXTENSIONS gnb_split_6 gnb_split_7_2 gnb_split_8)
14 set(LIBRARIES srsran_flexible_o_du_split_6 srsran_flexible_o_du_split_7_2 srsran_flexible_o_du_split_8)
15
16 foreach (NAME LIB IN ZIP_LISTS EXTENSIONS LIBRARIES)
17   add_executable(${NAME} gnb.cpp)
18   add_backward(${NAME})
19   target_include_directories(${NAME} PRIVATE ${CMAKE_SOURCE_DIR})
20   target_link_libraries(${NAME} PRIVATE gnb_base ${LIB})
NORMAL main CMakeLists.txt
:noh
0-0 0:nvim*
utf-8 | unix | cmake 65% 49:1
"nvim" 13:53 25-Jun-25
```



# srsRAN code organization - (*apps/gnb/gnb.cpp*)

```
21 {
20 // Set the application error handler.
19 set_error_handler(app_error_report_handler);
18
17 static constexpr std::string_view app_name = "gNB";
16 app_services::application_message_banners::announce_app_and_version(app_name);
15
14 // Set interrupt and cleanup signal handlers.
13 register_interrupt_signal_handler(interrupt_signal_handler);
12 register_cleanup_signal_handler(cleanup_signal_handler);
11
10 // Enable backtrace.
9 enable_backtrace();
8
7 // Setup and configure config parsing.
6 CLI::App app("srsGNB application");
5 app.config_formatter(create_yaml_config_parser());
4 app.allow_config_extras(CLI::config_extras_mode::error);
3 // Fill the generic application arguments to parse.
2 populate_cli11_generic_args(app);
1
227 gnb_appconfig gnb_cfg
1 // Configure CLI11 with the gNB application configuration schema.
2 configure_cli11_with_gnb_appconfig_schema(app, gnb_cfg);
3
4 auto o_cu_cp_app_unit = create_o_cu_cp_application_unit("gnb");
5 o_cu_cp_app_unit->on_parsing_configuration_registration(app);
6
7 auto o_cu_up_app_unit = create_o_cu_up_application_unit("gnb");
8 o_cu_up_app_unit->on_parsing_configuration_registration(app);
9
10 auto o_du_app_unit = create_flexible_o_du_application_unit("gnb");
11 o_du_app_unit->on_parsing_configuration_registration(app);
12
13 // Set the callback for the app calling all the autoderivation functions.
14 app.callback([&app, &gnb_cfg, &o_du_app_unit, &o_cu_cp_app_unit, &o_cu_up_app_unit](){
15     autoderive_gnb_parameters_after_parsing(app, gnb_cfg);
16
17     cu_cp_unit_config& cu_cp_cfg = o_cu_cp_app_unit->get_o_cu_cp_unit_config().cucp_cfg;
18
19     autoderive_slicing_args(o_du_app_unit->get_o_du_high_unit_config().du_high_cfg.config, cu_cp_cfg);
20     o_du_app_unit->on_configuration_parameters_autoderivation(app);
}

```



NORMAL main | E:42 W:2 gnb.cpp  
:noh  
0-0 0:nvim\*

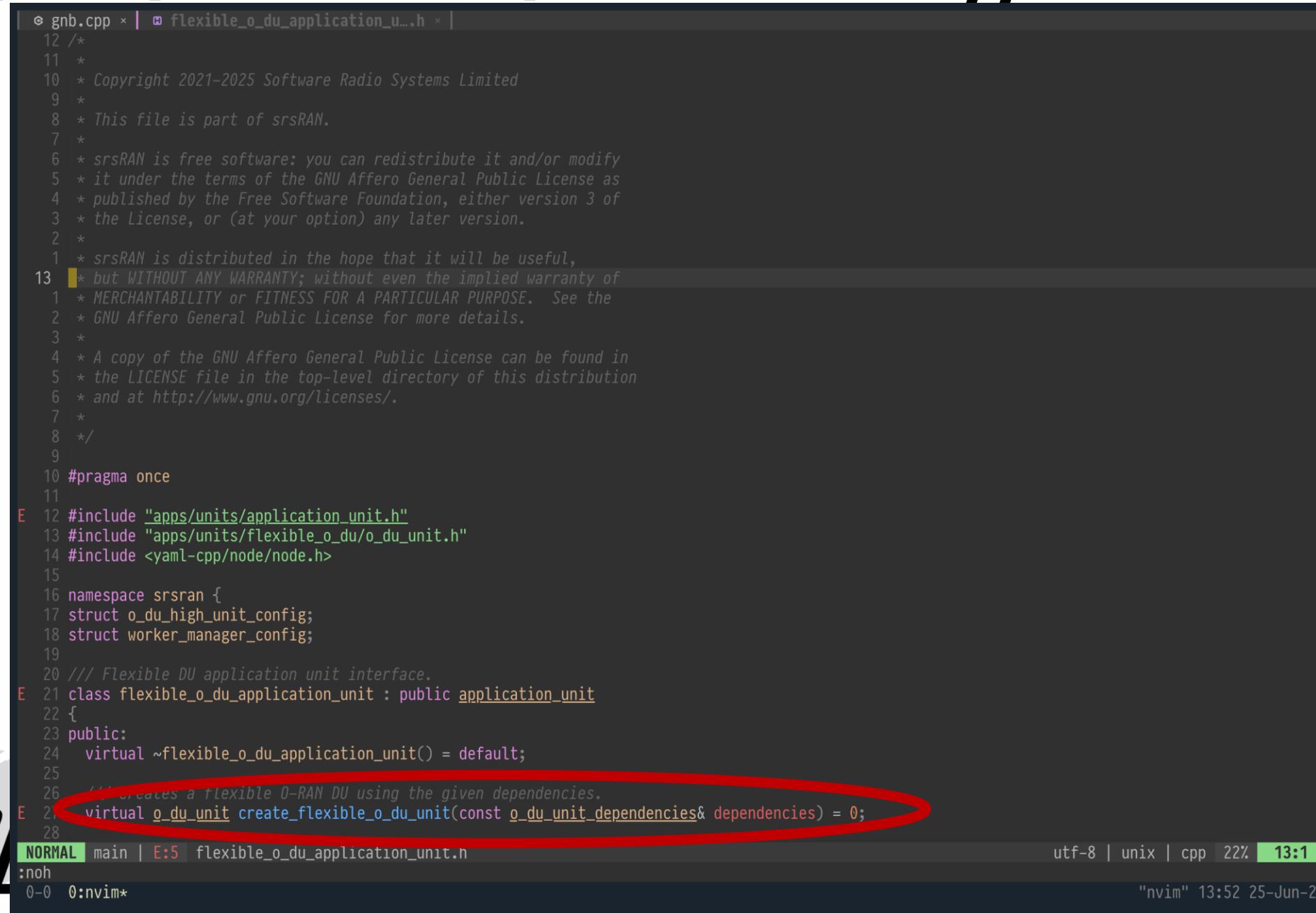
utf-8 | unix | cpp 39% 227:24

"nvim" 13:51 25-Jun-25

6GSNS



# srsRAN code organization – *(apps/units/flexible\_o\_du/flexible\_o\_du\_application\_unit.h)*



```
| ⑥ gnb.cpp * | ⑦ flexible_o_du_application_u...h * |
12 /*
11 *
10 * Copyright 2021-2025 Software Radio Systems Limited
9 *
8 * This file is part of srsRAN.
7 *
6 * srsRAN is free software: you can redistribute it and/or modify
5 * it under the terms of the GNU Affero General Public License as
4 * published by the Free Software Foundation, either version 3 of
3 * the License, or (at your option) any later version.
2 *
1 * srsRAN is distributed in the hope that it will be useful,
13 * but WITHOUT ANY WARRANTY; without even the implied warranty of
1 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
2 * GNU Affero General Public License for more details.
3 *
4 * A copy of the GNU Affero General Public License can be found in
5 * the LICENSE file in the top-level directory of this distribution
6 * and at http://www.gnu.org/licenses/.
7 *
8 */
9
10 #pragma once
11
E 12 #include "apps/units/application_unit.h"
13 #include "apps/units/flexible_o_du/o_du_unit.h"
14 #include <yaml-cpp/node/node.h>
15
16 namespace srsran {
17 struct o_du_high_unit_config;
18 struct worker_manager_config;
19
20 /// Flexible DU application unit interface.
E 21 class flexible_o_du_application_unit : public application_unit
22 {
23 public:
24     virtual ~flexible_o_du_application_unit() = default;
25
26     /**
27      * Creates a flexible O-RAN DU using the given dependencies.
28      */
E 28     virtual o_du_unit create_flexible_o_du_unit(const o_du_unit_dependencies& dependencies) = 0;
29
NORMAL main | E:5 flexible_o_du_application_unit.h
:noh
0-0 0:nvim*
```

utf-8 | unix | cpp 22% 13:1

"nvim" 13:52 25-Jun-25



# srsRAN code organization -

*(apps/units/flexible\_o\_du/split\_dynamic/dynamic\_o\_du/application\_unit\_impl.cpp)*

```
| e gnb.cpp * | e flexible_o_du_application_u...h * | e dynamic_o_du_application_u...cpp * |
17
16 bool dynamic_o_du_application_unit_impl::on_configuration_validation(
E 15   const os_sched_affinity_bitmask& available_cpus) const
14 {
13   return validate_dynamic_o_du_unit_config(unit_cfg, available_cpus);
12 }
11
E 10 dynamic_o_du_application_unit_impl::dynamic_o_du_application_unit_impl(std::string_view app_name)
9 {
E 8   unit_cfg.odu_high_cfg.du_high_cfg.config.pcaps.f1ap.filename = fmt::format("/tmp/{}_f1ap.pcap", app_name);
E 7   unit_cfg.odu_high_cfg.du_high_cfg.config.pcaps.f1u.filename = fmt::format("/tmp/{}_f1u.pcap", app_name);
E 6   unit_cfg.odu_high_cfg.du_high_cfg.config.pcaps.rlc.filename = fmt::format("/tmp/{}_rlc.pcap", app_name);
E 5   unit_cfg.odu_high_cfg.du_high_cfg.config.pcaps.mac.filename = fmt::format("/tmp/{}_mac.pcap", app_name);
4 // Note: do not update the default e2ap pcap filename.
3 }
2
E 1 void dynamic_o_du_application_unit_impl::on_parsing_configuration_registration(CLI::App& app)
81
1 configure_cli11_with_dynamic_o_du_unit_config_schema(app, unit_cfg);
2
E 4 o_du_unit dynamic_o_du_application_unit_impl::create_flexible_o_du_unit(const o_du_unit_dependencies& dependencies)
5 {
6   return dynamic_o_du_factory(unit_cfg).create_flexible_o_du(dependencies);
7 }
8
W 9 std::unique_ptr<flexible_o_du_application_unit> dynamic_o_du_application_unit_impl::create_flexible_o_du_unit(std::string_view app_name)
10
11 return std::make_unique<dynamic_o_du_application_unit_impl>(app_name);
12
13
E 14 void dynamic_o_du_application_unit_impl::dump_config(YAML::Node& node) const
15 {
16   fill_dynamic_o_du_unit_config_in_yaml_schema(node, unit_cfg);
17 }
18
E 19 void dynamic_o_du_application_unit_impl::fill_worker_manager_config(worker_manager_config& config)
20 {
21   fill_dynamic_o_du_worker_manager_config(config, unit_cfg);
22 }

NORMAL main | E:17 W:7 dynamic_o_du_application_unit_impl.cpp          utf-8 | unix | cpp 78% 81:1
:noh
0-0 0:nvim*
```

"nvim" 13:52 25-Jun-25

