



AI for Next Generation Cellular Networks

Bozidar Radunovic

Microsoft Research

Why Should I Listen to This Talk?

- How to efficiently collect and filter large amount of data at scale?
- How to use wireless network to connect real-time edge systems?

Also, a high-level and hopefully slightly entertaining tutorial on cellular networking.

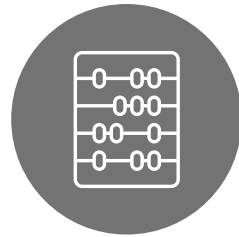
Talk Outline



INTRO



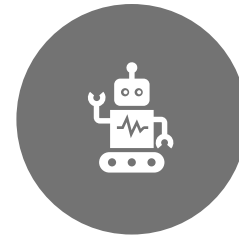
HOW TO BUILD
A NETWORK?



HOW TO PROGRAM
A NETWORK?



HOW TO IMPROVE
EXISTING NETWORKS?



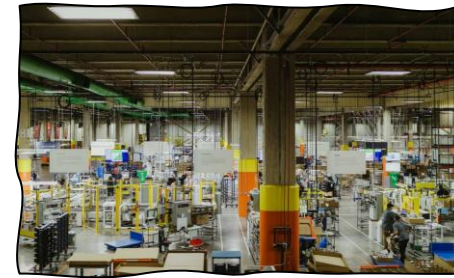
HOW TO CREATE
NEW NETWORKS?

What is a Cellular Network?

Macro network



Indoor/enterprise network



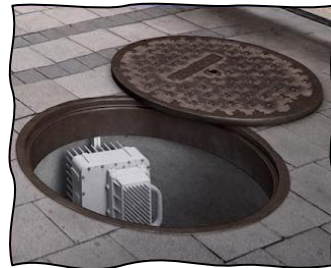
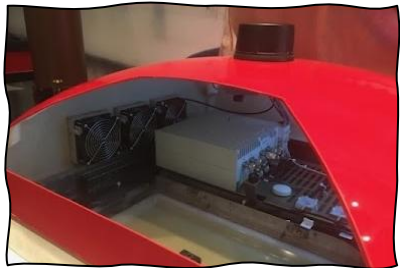
Mine

Hotel/shopping mall

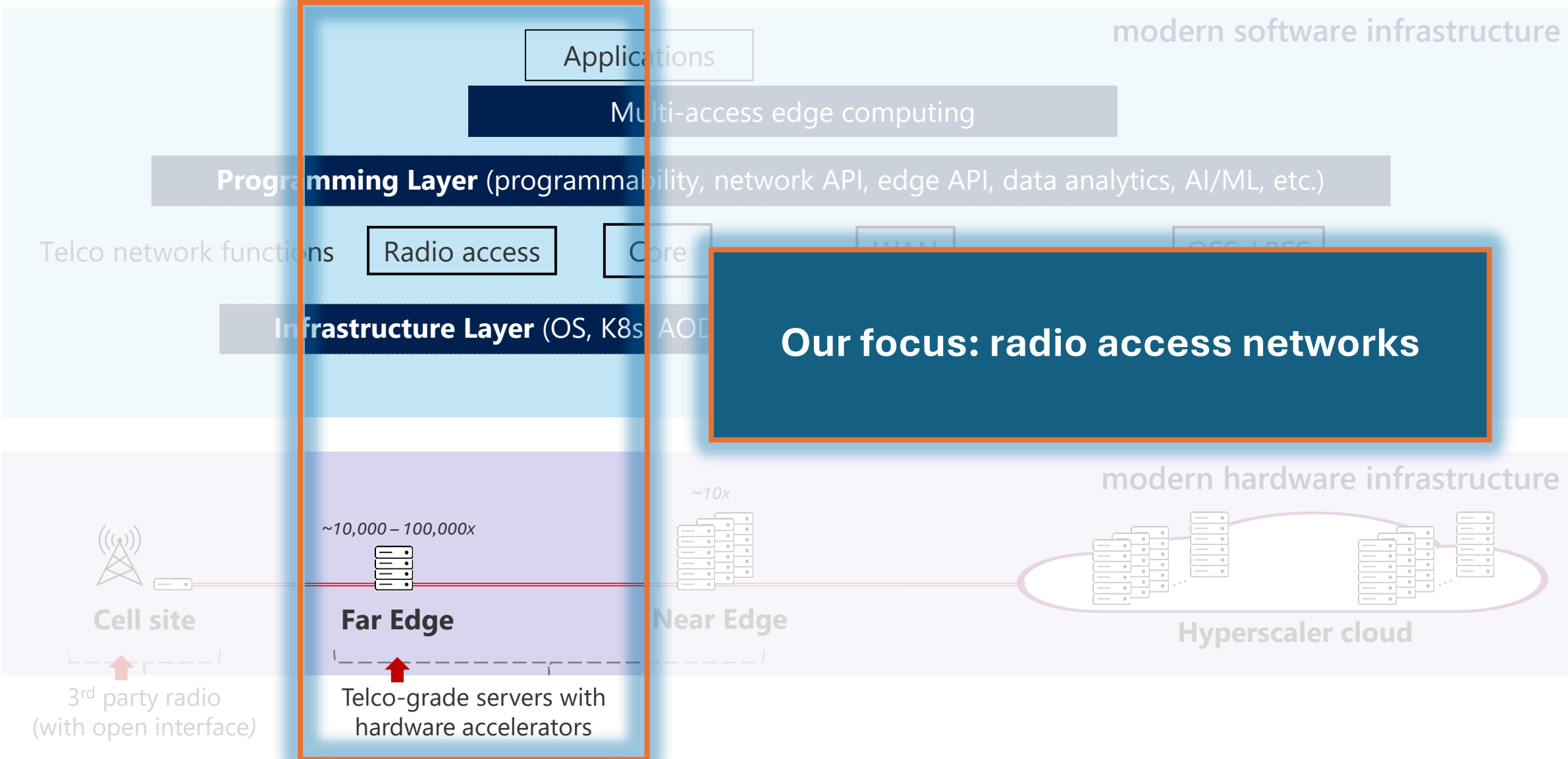
Factory

Port

Densification for performance



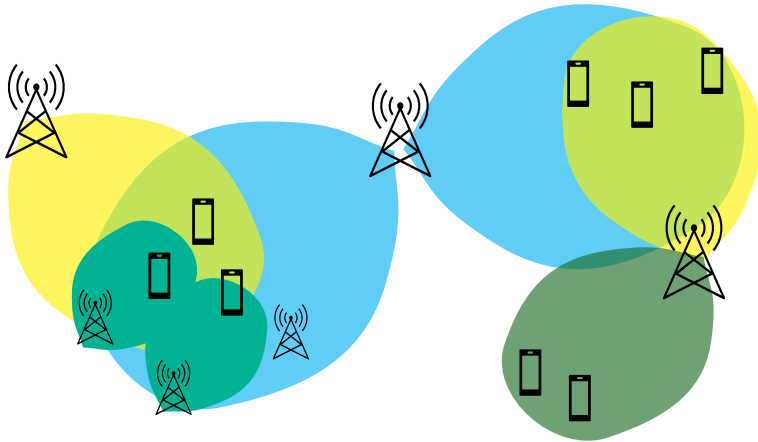
Cellular Network in the Era of Cloud



Why Do We Focus on Radio-access Networks?

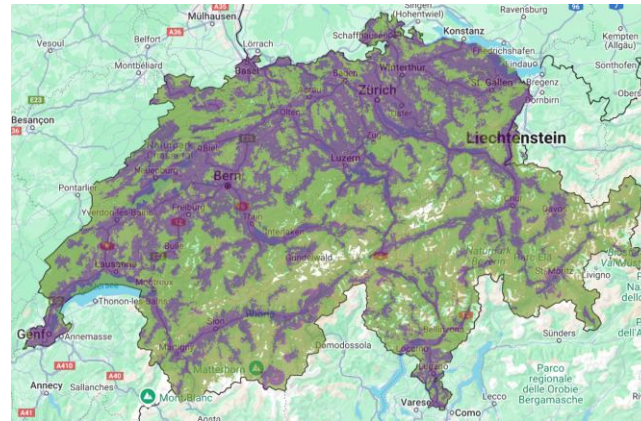
Spectrum is the bottleneck

Challenge: Efficiency



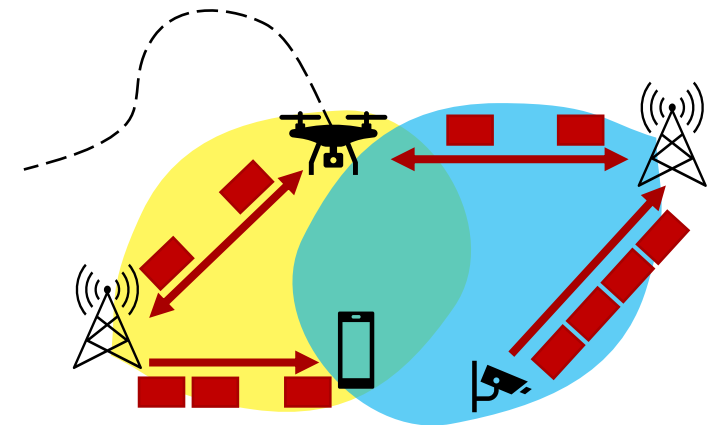
Networks are increasingly complex (more spectrum, denser deployments)

Challenge: Scale



100,000s of cells and infrastructure (\$\$\$, GWs)

Challenge: Control



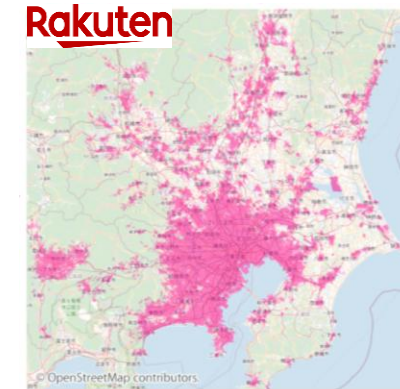
Guaranteed QoS, precise location, low-latency edge application

Why Now?



DALLAS, December 04, 2023

AT&T to Accelerate Open and Interoperable Radio Access Networks (RAN) in the United States through new collaboration with Ericsson



- RAN is being virtualized now, as a part of 5G transition
- This introduces open interfaces that allow to replace different components
- It is very easy to deploy realistic 5G systems

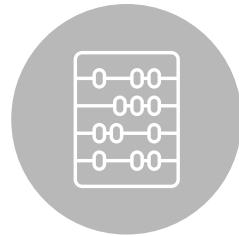
Talk Outline



INTRO



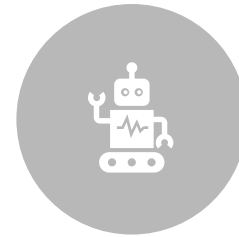
HOW TO BUILD
A NETWORK?



HOW TO PROGRAM
A NETWORK?

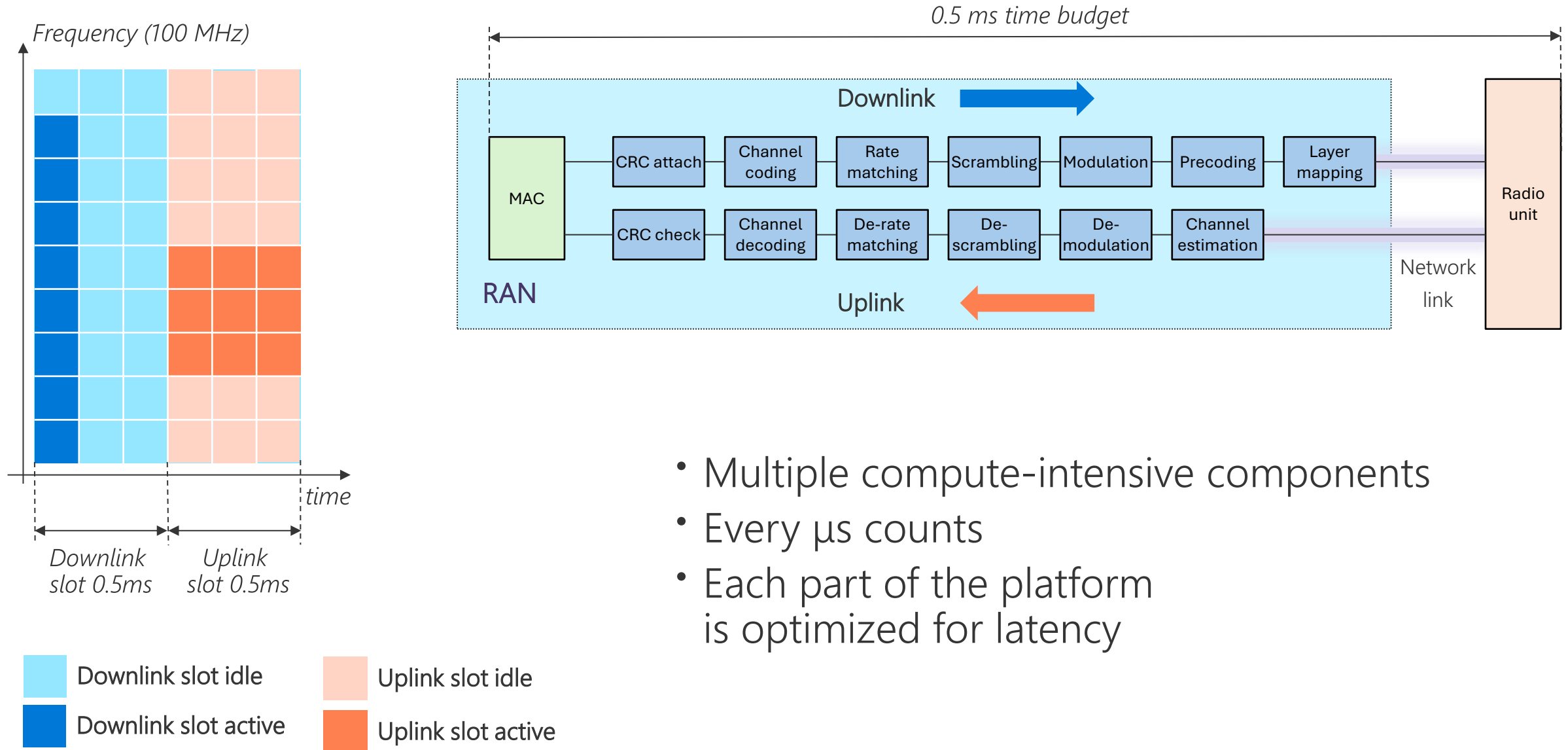


HOW TO IMPROVE
EXISTING NETWORKS?



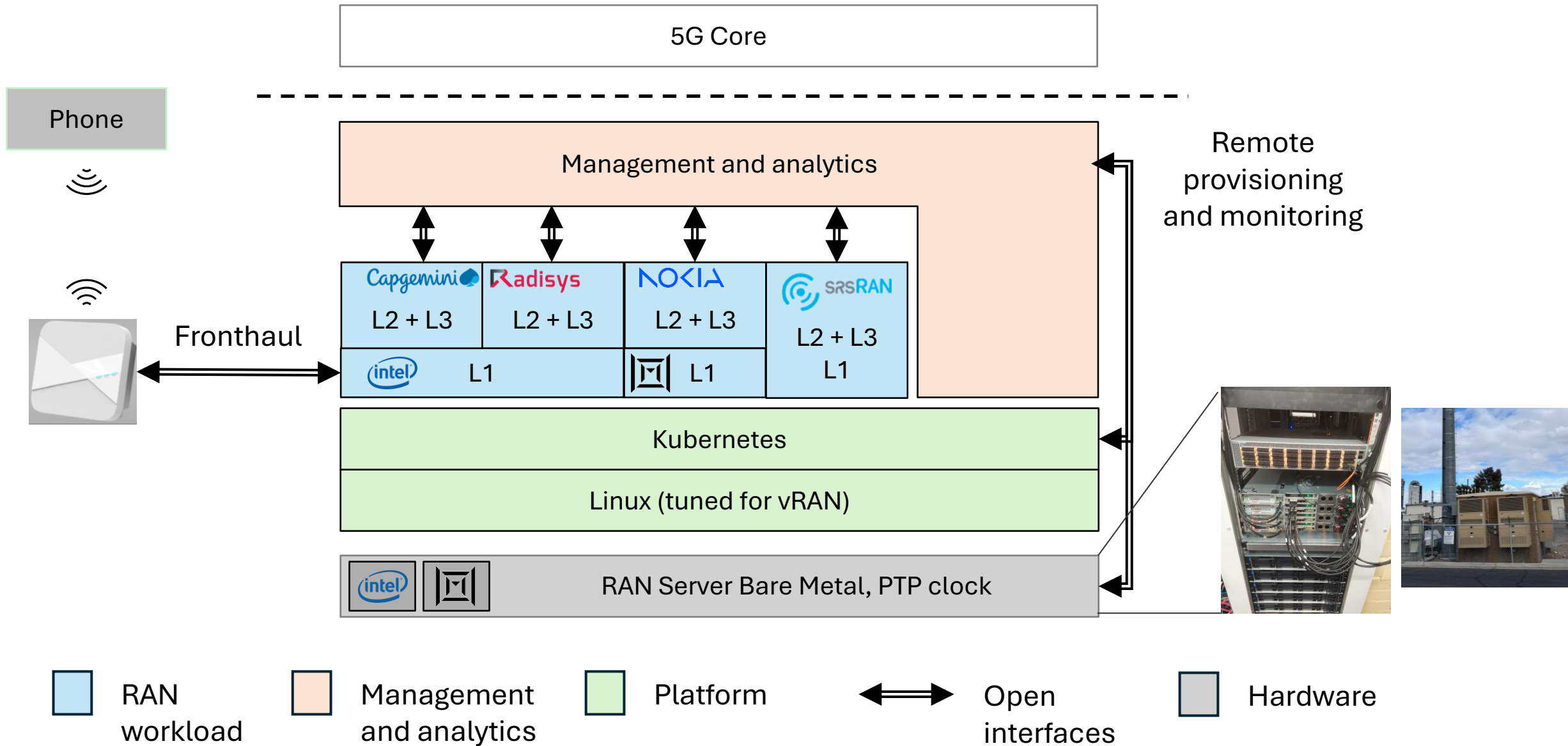
HOW TO CREATE
NEW NETWORKS?

What is Special About Radio (RAN) Software?

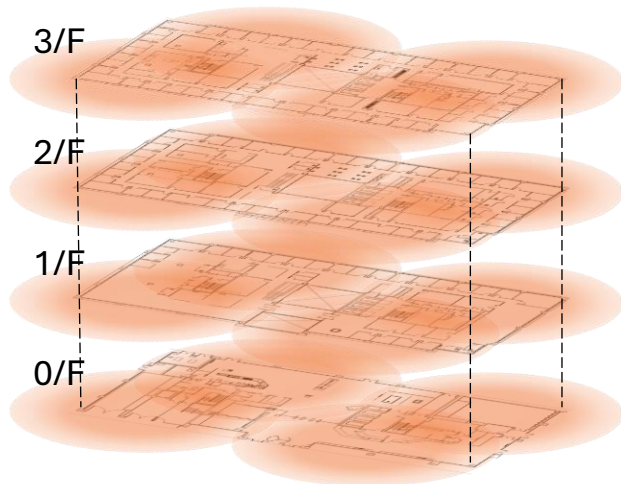


- Multiple compute-intensive components
- Every μs counts
- Each part of the platform is optimized for latency

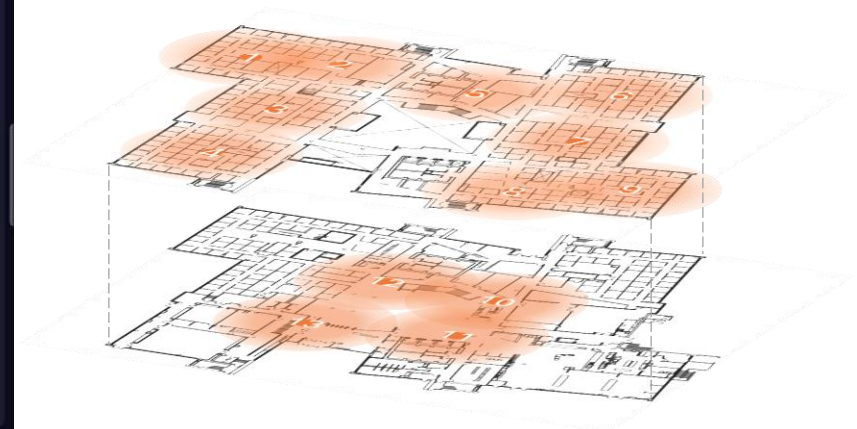
Architecture of a 5G Rack – O-RAN standard



Microsoft Research 5G Testbed



Cambridge: 5 floors, 18 RUs, ~40 devices



Redmond: 2 floors, 14 RUs



Mobicom demo, 2023

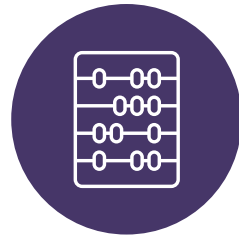
Talk Outline



INTRO



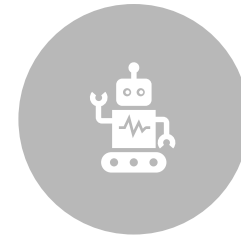
HOW TO BUILD
A NETWORK?



HOW TO PROGRAM
A NETWORK?



HOW TO IMPROVE
EXISTING NETWORKS?



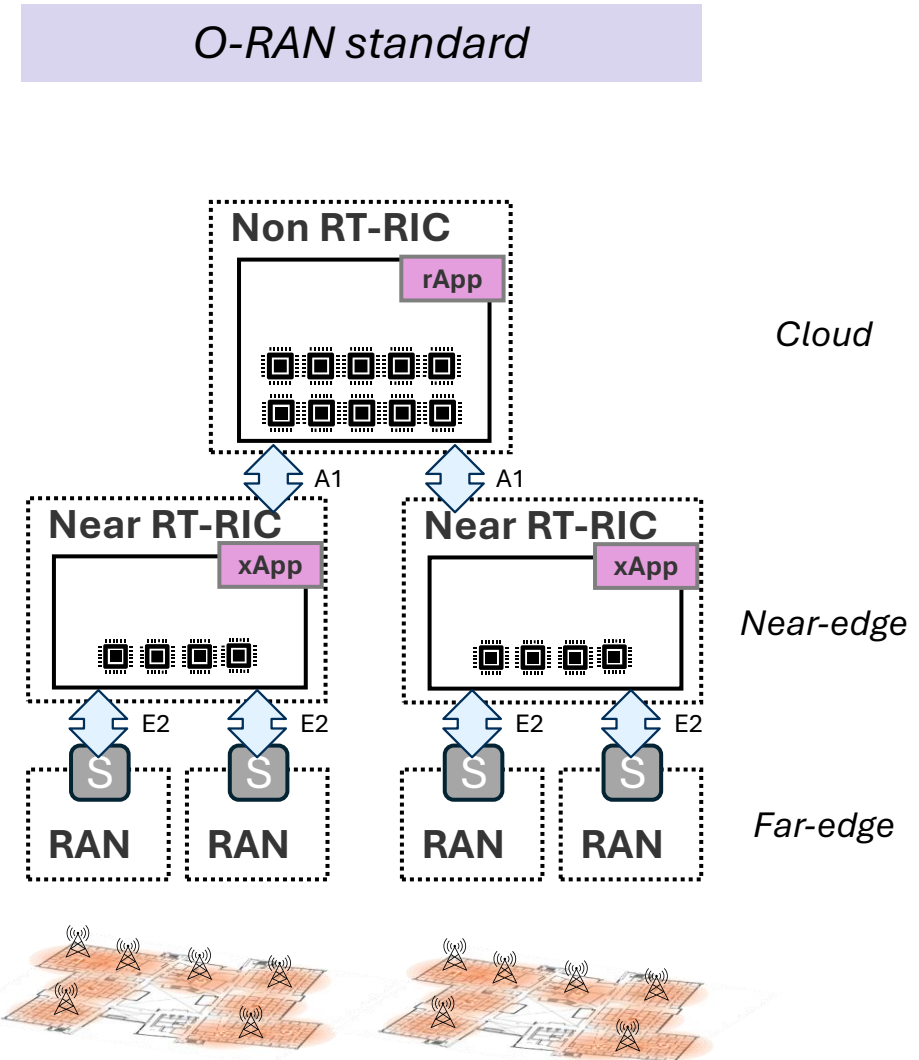
HOW TO CREATE
NEW NETWORKS?

Cellular Radio (RAN) Programmability Today

- Near-RT RIC (RAN intelligent controller):
 - Detailed control (e.g. handover, traffic steering)
 - Control loop $>10\text{ms}$
- Non-RT RIC:
 - High-level policy
 - Control loop $>1\text{s}$
- Standardized service model (S)

Challenges:

- Hard to get data for new applications
- Hard to implement real-time control loops ($<10\text{ms}$)



Challenges With Service Models

Static data model definition:

Table 4.2.1.2.2-1: Definition for Average over-the-air packet delay in the UL per DRB per UE

Definition	Average over-the-air packet delay in the UL per DRB per UE. This measurement is applicable for EN-DC and SA. This measurement refers to packet delay for DRBs. This measurement provides the average (arithmetic mean) time it takes to successfully receive a transport block from the time of UL transmission indicated in scheduling grant.
Detailed Definition:	$M(T, drbid) = \frac{\sum_{i \in T} t_{succ}(i, drbid) - t_{sched}(i, drbid)}{I(T)}$, where explanations can be found in the table 4.2.1.2.2-2 below.

- New device measurement reporting
- New test condition list
- Add OR to testing condition
- Flexible definition of sampling bins

Change Request				
Document	O-RAN.WG3.E2SM-KPM	ver	02.00	CR RSY-0002 rev 5
Reason for Change:	New Report Service Styles 4 and 5 introduced for flexible reporting of UE level measurements			
Summary of change:	- Addition of Report Service Style 4 and 5 for reporting of measurements for Condition based UEs and multiple UEs - Addition of Action Definition Format 4 and 5 for Report Service style 4 and 5 - Addition of Indication Message Format 3 for Report Service Style 4 and 5			
Consequences if not approved:	Reporting of UE level measurements for multiple UEs and Condition based filtering of UEs will not be available.			
Change Request				
Document	O-RAN.WG3.E2SM-KPM	ver	02.01	CR CMCC-0002 rev 2
Reason for Change:	Existing test condition types need to be extended for identifying UEs.			
Summary of change:	Extending new test condition types for identifying UEs.			
Consequences if not approved:	The test condition list is incomplete.			
Change Request				
Document	O-RAN.WG3.E2SM-KPM	ver	02.02	CR INT-0016 rev 1
Reason for Change:	Currently, the logical connection of multiple UE filtering conditions by the Matching Condition IE in the Action Definition Formats 3 and 4 can only be with "AND" connection. "OR" connection cannot be supported, thus limiting their applicability. In ASN.1, "MatchingCondition" was directly referred to the choice structure and not extensible.			
Change Request				
Document	O-RAN.WG3.E2SM-KPM	ver	02.02	CR INT-0017 rev 4
Reason for Change:	E2SM-KPM supports subscription of distribution type measurements from RAN nodes where a measured value is categorized based on range of values and the corresponding bin is incremented accordingly. For these distribution type measurements, what is reported to Near-RT RIC is the number of samples for each and every bins. However, currently, definition of each bin itself (i.e. ranges of values that the measured values are collected onto) are left up to RAN vendors implementation. Near-RT RIC has no other option but to follow the RAN vendor-specific bin ranges without knowing what range each bin represents for.			

Revision history

Date	Revision	Description
2018.11.25	00.00.0	Applied skeleton 00.01.04 to build KPM Monitor E2SM
2018.12.01	00.00.1	Applied change from E2SM-NI-v000.01.05
2018.12.09	00.00.2	Removed Policy section, specifies detail list of container IE
2018.12.11	00.00.3	Align with E2SM-NI as per comments from A. Urie
2018.12.12	00.00.4	Add E2 Node ID, O-CU-CP/O-CU-UP container as per comments from WG3
2018.12.18	00.00.5	Updated Style Type and Format Type definition aligned with Nokia E2SM-NI v00.01.08
2019.01.14	00.00.6	Change name from KPIMON to KPMMON, additional alignment with Nokia E2SM-NI v00.01.08 and ASN.1 message addition
2019.01.16	00.00.7	ASN.1 update
2019.01.17	00.00.8	Add RIC Style Type in RIC Indication message IE with a corresponding change to the ASN.1 encoding
2019.01.19	00.00.9	Change the name to E2SM-KPM, add Action Definition with RIC style list, made section 7.8 update for additional alignment with E2SM-NI, add Annex A.
2019.01.20	00.00.10	Update Scope, rename Slice ID to S-NSSAI, add Action Definition to each style definition, remove EPC and 5GC style, add CU-CP EPC style, fix the use of 5QI and QCI for E2 indication header, and correctly reference 28.552 for 5GC IEs
2019.01.20	00.00.11	Remove Report Period IE Test Condition and Report Period IE Value from trigger definition and ASN.1. Reference 28.552 for Active UE and PDCP DL/UL data volume
2019.01.22	00.00.11a	Removed en-gNB definition
2019.01.22	00.00.12	E2SM-KPM-IEs { iso(1) identified-organization(3) dod(6) internet(1) private(4) enterprise(1) 53148 e2(1) version1 (1) e2sm(2) e2sm-KPM-IEs (2)}
2019.01.22	00.00.13	Section 6.1 update, Change E2 Node ID to KPM Node ID
2019.01.22	00.00.13a	Change E2SM-NI-IndicationMessage to E2SM-KPM-IndicationMessage
2020.01.29	v01.00	Adopt Jio's comments, change the number of NR DL/UL PRB from 100 to 273.
2020.12.16	02.00.00	Adopt INTEL.AO's CR-0001 and CR-0002 for E2SM-KPM with cleaning up old texts and ASN.1 in v01.00
2021.02.24	02.00.01	Adopt AT&T.AO's CR-0001 for UE-level measurements subscription and retrieval
2021.03.03	02.00.02	Adopt CSP.AO's CR-0001 for Incomplete Flag
2021.03.30	02.00.03	Adopt INTEL.AO's CR-0005 for clean-up
2021.06.09	02.00.04	Adopt (1) INT's CR-0006; (2) INT's CR-0008; (3) RSY's CR-0004
2021.07.09	02.00.05	Adopt (1) INT's CR-0009; (2) NEU.AO's CR-0001
2021.08.10	02.00	TSC Approved
2021.10.13	02.01.00	Adopt (1) INTEL.AO's CR-0011; (2) RSY.AO's CR-0002
2021.10.27	02.01.01	Adopt KDDI's CR-0001.
2021.11.22	02.01.02	Editorial Updates based on review comments during WG3 approval process
2022.02.07	02.01	Version ready for Nov21 publication
2022.03.23	02.02.00	Adopt (1) TIM.AO's CR-0003; (2) CMCC.AO's CR-0001; (3) CMCC.AO's CR-0002
2022.04.14	02.02	Version ready for Mar22 publication
2022.05.11	02.02.01	Adopt (1) INT's CR-0015; (2) INT's CR-0016; (3) INT's CR-0017
2022.07.20	02.02.02	Adopt INT's CR-0022
2022.07.24	02.02.03	Aligned to new template
2022.08.09	02.03	TSC Approved
2022.11.09	02.03.01	Adopt (1) INT's CR-0024; (2) INT's CR-0027
2022.11.20	02.03.02	Editorial changes reflecting comments received during WG3 approval process
2022.11.25	03.00	TSC Approved
2023.06.30	03.00.01	Adopt (1) NOK.AO's CR-0002; (2) MAV.AO's CR-0013
2023.07.26	03.00.02	Editorial changes reflecting comments received during WG3 approval process
2023.07.26	03.00.03	Further Editorial changes reflecting comments received during WG3 approval process

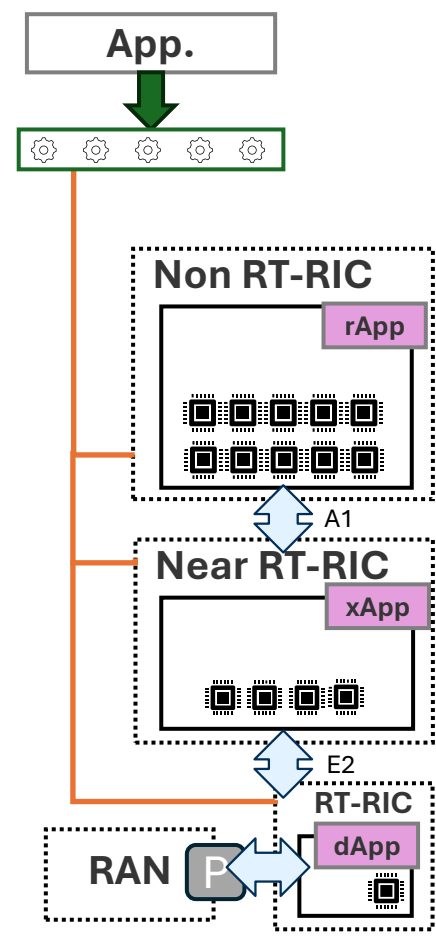


Project Janus – Programmable radio and platform for RAN

- **Dynamic probes**
Fine-tune service model for different applications
- **Real-time RIC**
Ability to write low-latency applications
- **Application deployment**
Simplify application deployment across different components



Open source: <https://github.com/microsoft/jbpf>



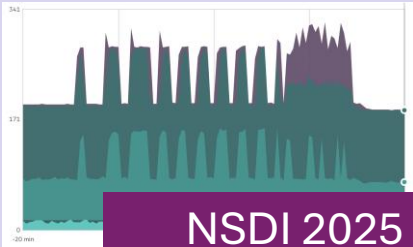


Project Janus – New Use Cases



Efficiency

Power saving



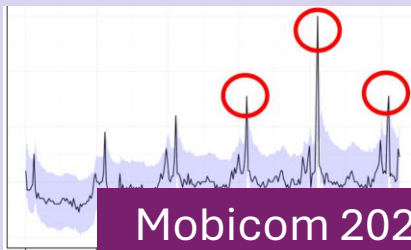
NSDI 2025

Compute efficiency



Sigcomm 2021

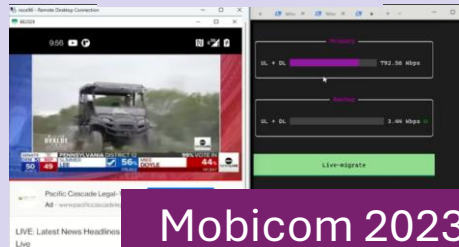
Anomaly detection



Mobicom 2024

Cloud native radio

Live migration



Mobicom 2023

Failover management



Mobicom 2023

Live updates



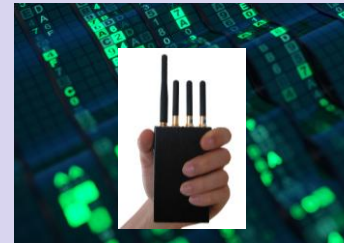
Security

DDoS detection



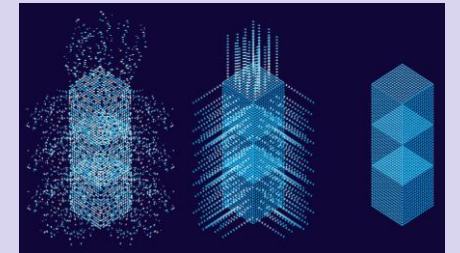
Security 2024

Interference localization

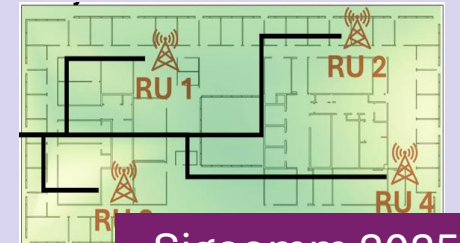


New value

AI/ML + Signal processing



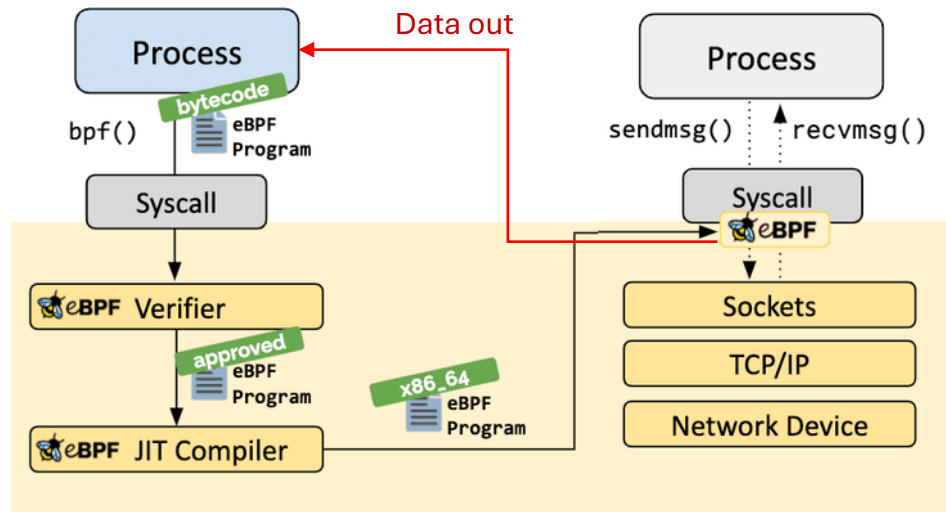
Distributed coordination



Sigcomm 2025

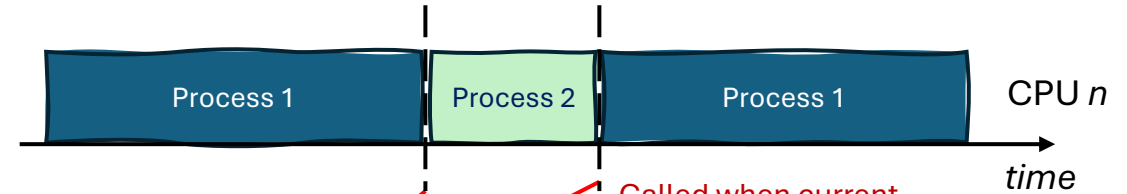
Allow developer ecosystem to build novel applications and services

Dynamic Probes: eBPF for Linux Kernel Observability



Safe dynamic programs for kernel observability and tracing

Program example



```
int raw_tracepoint_sched_switch(struct bpf_raw_tracepoint_args *ctx)
{
    struct task_struct *prev = (struct task_struct *)ctx->args[1];
    struct task_struct *next = (struct task_struct *)ctx->args[2];

    u32 prev_pid = prev->pid;
    u32 next_pid = next->pid;

    if (0 || prev_pid == 1 || next_pid == 1) {
        u64 ts = bpf_ktime_get_ns();

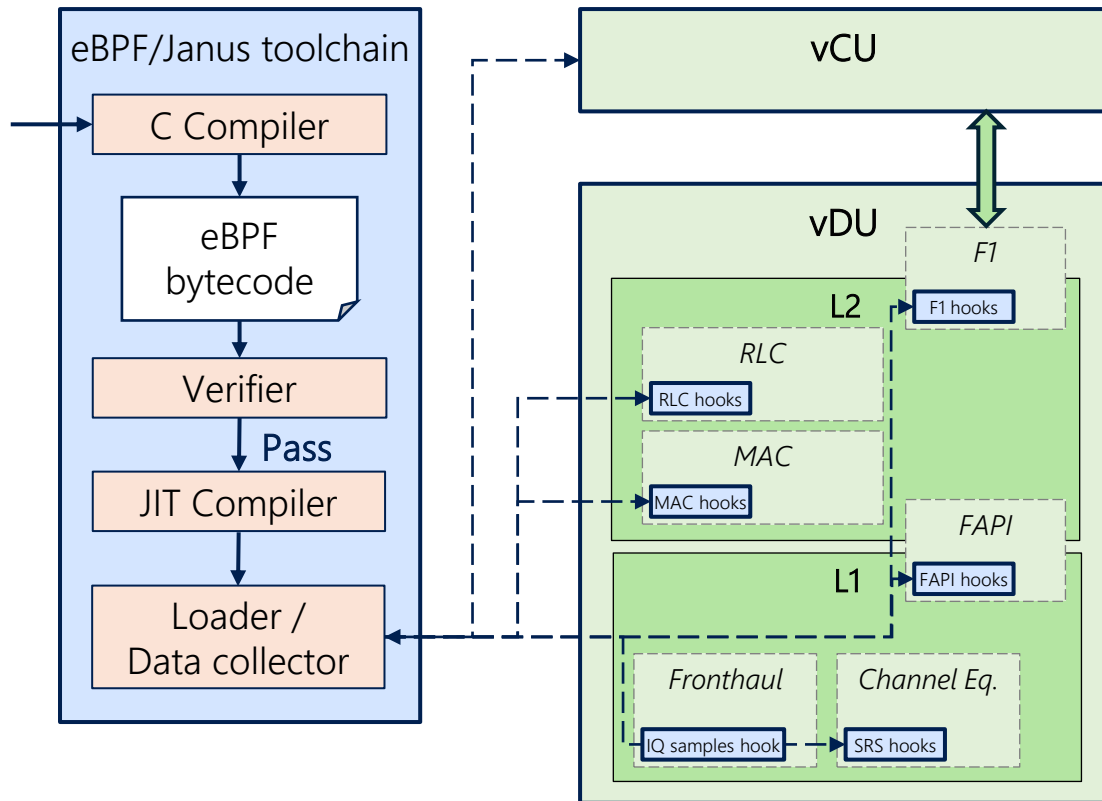
        u32 *pthid = thids_to_arrid.lookup(&prev_pid);
        if (pthid)
            store_start(pthid, ts);

        u32 *thid = thids_to_arrid.lookup(&next_pid);
        if (thid)
            update_hist(next_pid, ts, thid);
    }
    return 0;
}
```

Update histogram of execution times

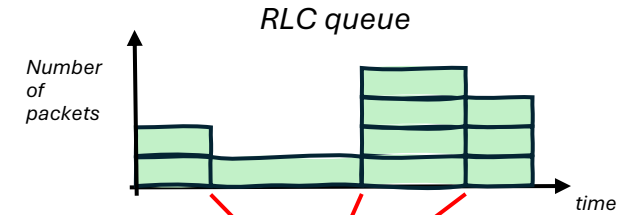
Dynamic Probes: eBPF for Radio (RAN) Observability

User-mode eBPF analog to the kernel one:



Hooks provide access to standard data structures (3GPP, O-RAN, FAPI)

Program example:



Called when RLC packet arrived or departed

```
// Total ingress throughput (pkts)
DEFINE_PROTOHASH_64(thr_P, MAX_NUM_UE);

SEC("janus_ran_layer2")
uint64_t bpf_prog(void *state)
{
    struct janus_ran_layer2_ctx *ctx;
    uint32_t index = 0;
    struct janus_size_ul_rlc_f1u *report, *report_end;
    uint64_t timestamp;

    ctx = (struct janus_ran_layer2_ctx *)state;

    timestamp = JANUS_TIME_GET_NS();

    uint32_t ind = JANUS_PROTOHASH_LOOKUP_ELEM_64(out, thr_B, thr_B, report->ue_index, report->rnti, cnt, 0);
    out->thr_B[ind].cnt+=report->size;
    ind = JANUS_PROTOHASH_LOOKUP_ELEM_64(out, thr_P, thr_P, report->ue_index, report->rnti, cnt, 0);
    out->thr_P[ind].cnt++;
}
```

Update packet count

Update bytes count

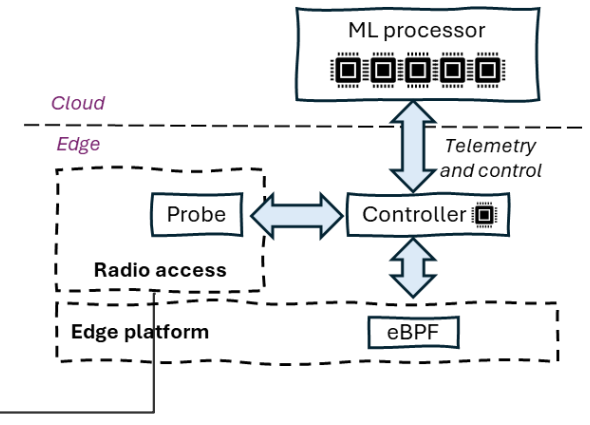
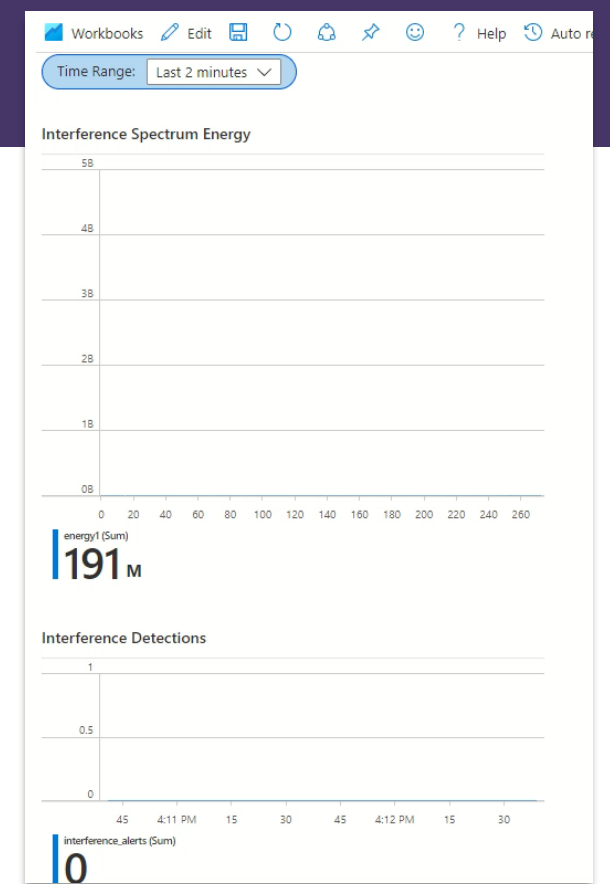
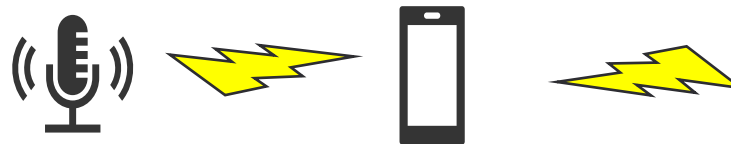
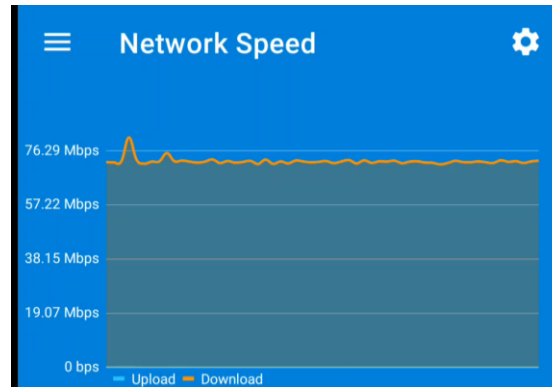
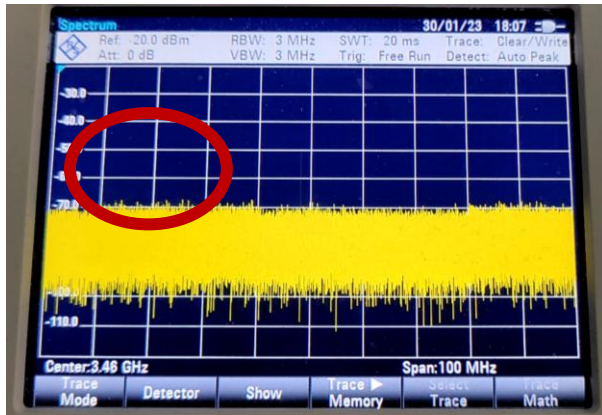
Table 4.2.1.2.2-1: Definition for Average over-the-air packet delay in the UL per DRB per UE

Definition	Average over-the-air packet delay in the UL per DRB per UE. This measurement is applicable for EN-DC and SA. This measurement refers to packet delay for DRBs. This measurement provides the average (arithmetic mean) time it takes to successfully receive a transport block from the time of UL transmission indicated in scheduling grant.
Detailed Definition:	$M(T, drbid) = \frac{\sum_i t_{succ}(i, drbid) - t_{sched}(i, drbid)}{I(T)}$, where explanations can be found in the table 4.2.1.2.2-2 below.

Example: Interference Detection

Turn RAN into spectrum analyzer:

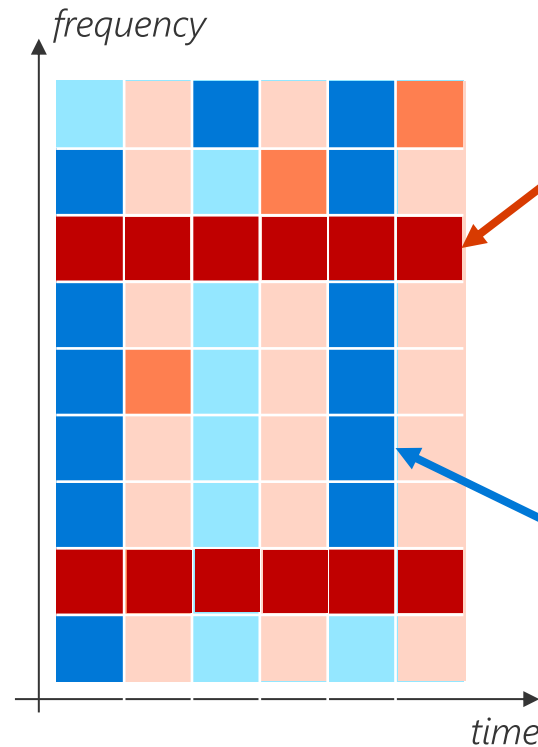
- Programmable RAN platform can detect external interference in live 5G network



Interference Detection – Behind The Scenes

Dynamic service model approach:

- Remove samples that have 5G traffic scheduled
- Send averages to reduce overhead



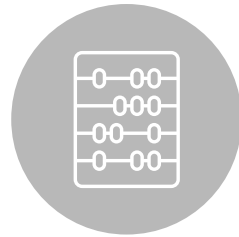
Talk Outline



INTRO



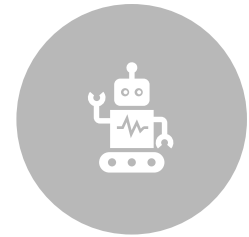
HOW TO BUILD
A NETWORK?



HOW TO PROGRAM
A NETWORK?



HOW TO IMPROVE
EXISTING NETWORKS?



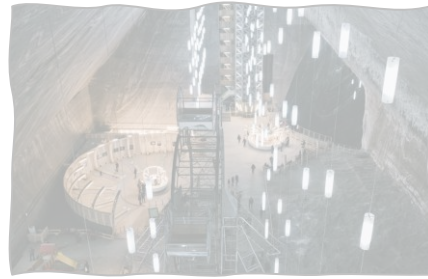
HOW TO CREATE
NEW NETWORKS?

Cellular Macro Networks

Macro network



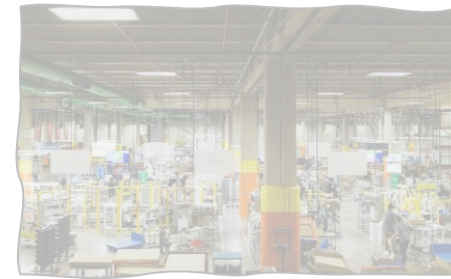
Indoor/enterprise network



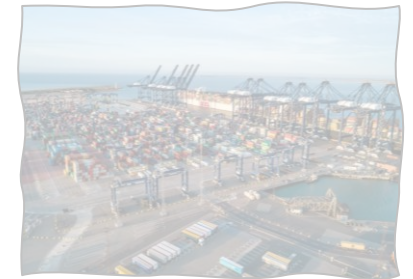
Mine



Hotel/shopping mall



Factory



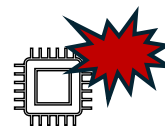
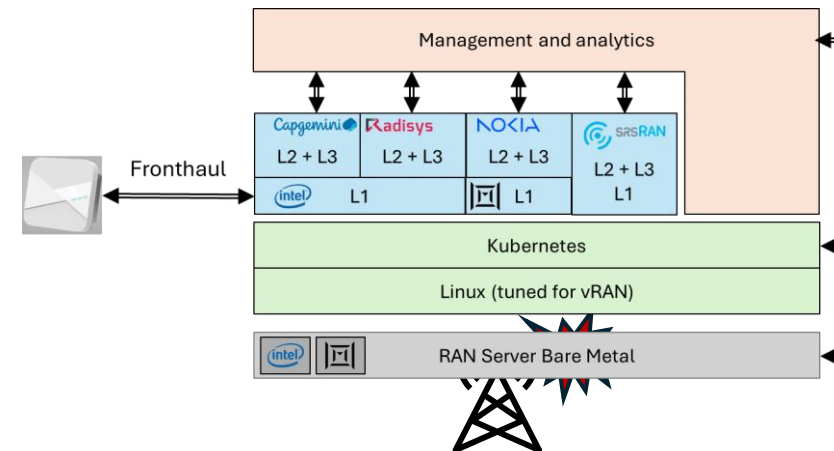
Port

- Critical network infrastructure: 0.99999 reliability (five 9s)
- Huge scale: 100k+ servers
- Hard to customize for individual use cases
- Motivating problem:
 - Troubleshoot issues at scale



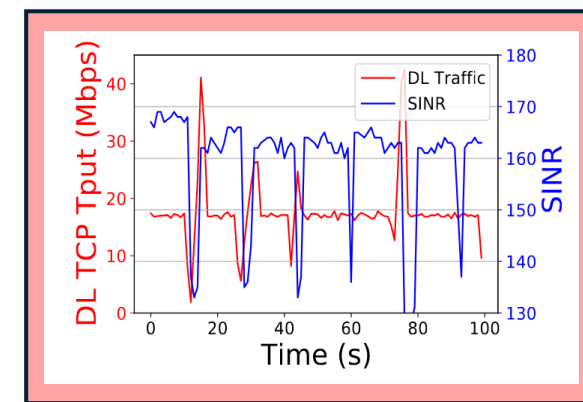
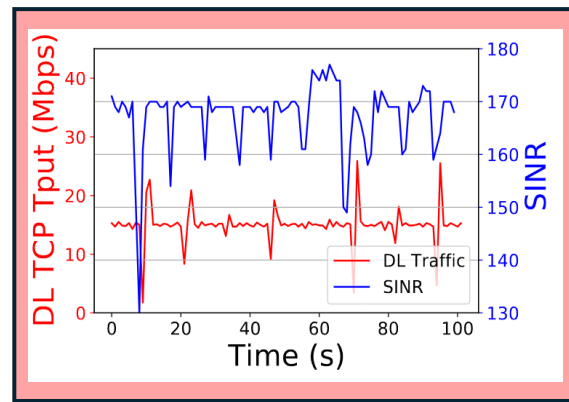
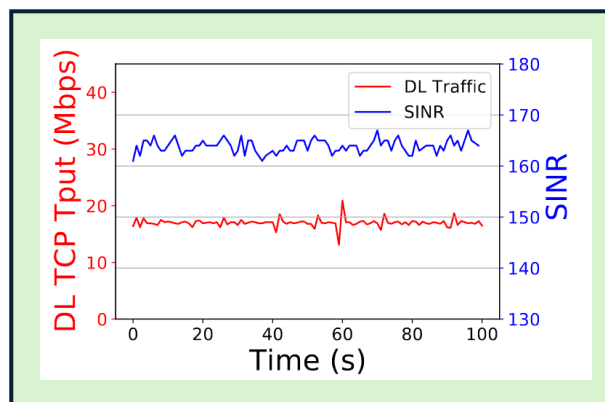
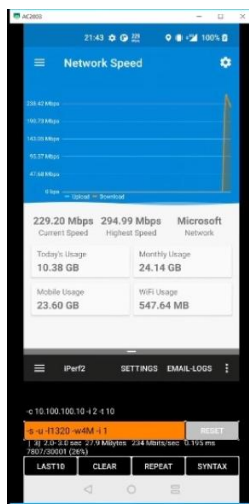
Motivating Example: RAN Troubleshooting

- Software and platform provided by different vendors
- Tight coupling between components
- Many possible sources of anomalies
- Requires dynamic data and distributed AI



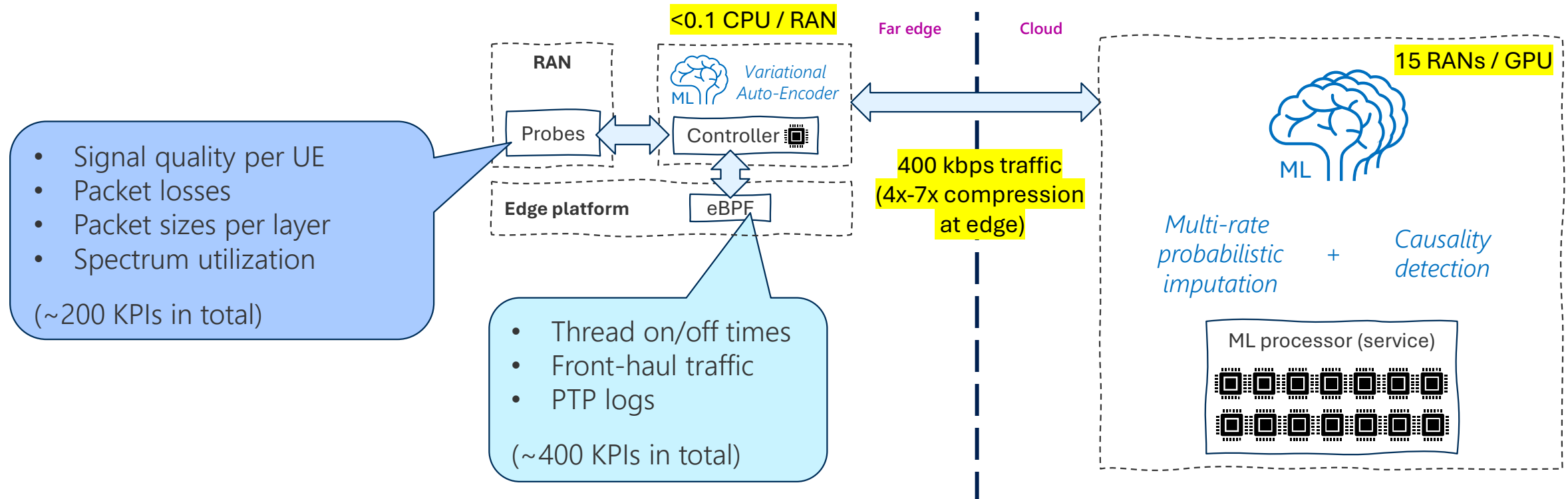
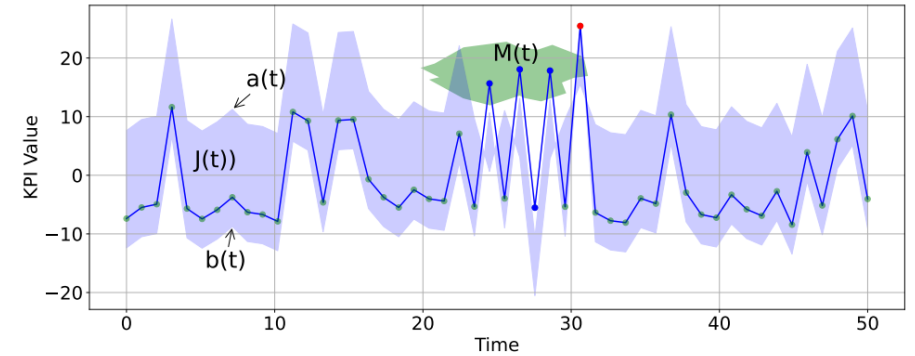
CPU contention:
NIC IRQ and radio share CPU core

Radio interference:
Overlapping radio cells



Distributed AI For Anomaly Detection

- Unsupervised anomaly detection:
 - Train ML model on well configured system
 - Continuously run the model in production



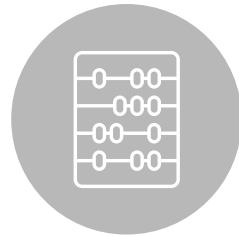
Talk Outline



INTRO



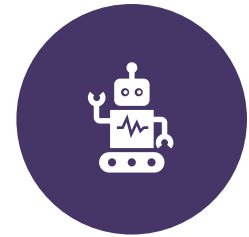
HOW TO BUILD
A NETWORK?



HOW TO PROGRAM
A NETWORK?



HOW TO IMPROVE
EXISTING NETWORKS?



HOW TO CREATE
NEW NETWORKS?

Industrial/Enterprise Cellular Networks

Macro network



Indoor/enterprise network



Mine



Hotel/shopping mall

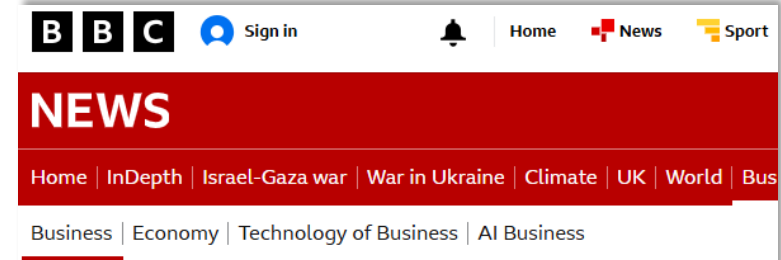
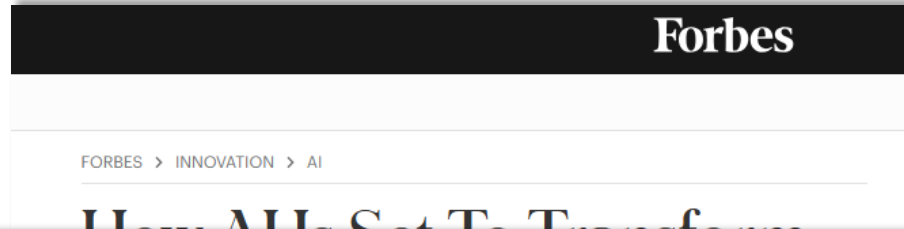
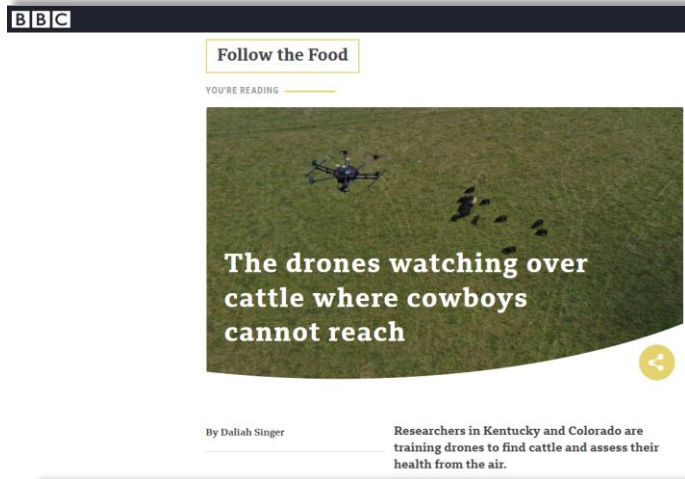


Factory



Port

Physical AI Will Use Industrial/Enterprise Cellular Networks



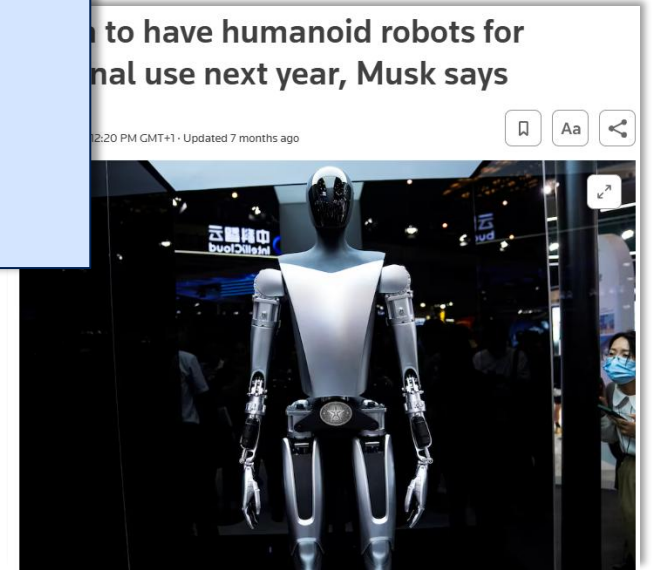
Most of these networks use private cellular spectrum and 5G network technology:
Predictable and programmable

AUTOMATION
The robots are here: How they are changing auto production

By Anushka Dixit | 29 January 2025



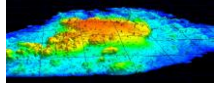

Explore the emerging role of AI-powered multipurpose at BMW, Tesla, and Mercedes-Benz, and what the future holds for robotics in manufacturing.


Despite these challenges, the trajectory for humanoid robots in automotive manufacturing is clear. As AI and robotics technologies continue to evolve, their capabilities will expand, making them even more integral to production processes...




Importance of Connectivity

Connectivity requirements


High bandwidth:    

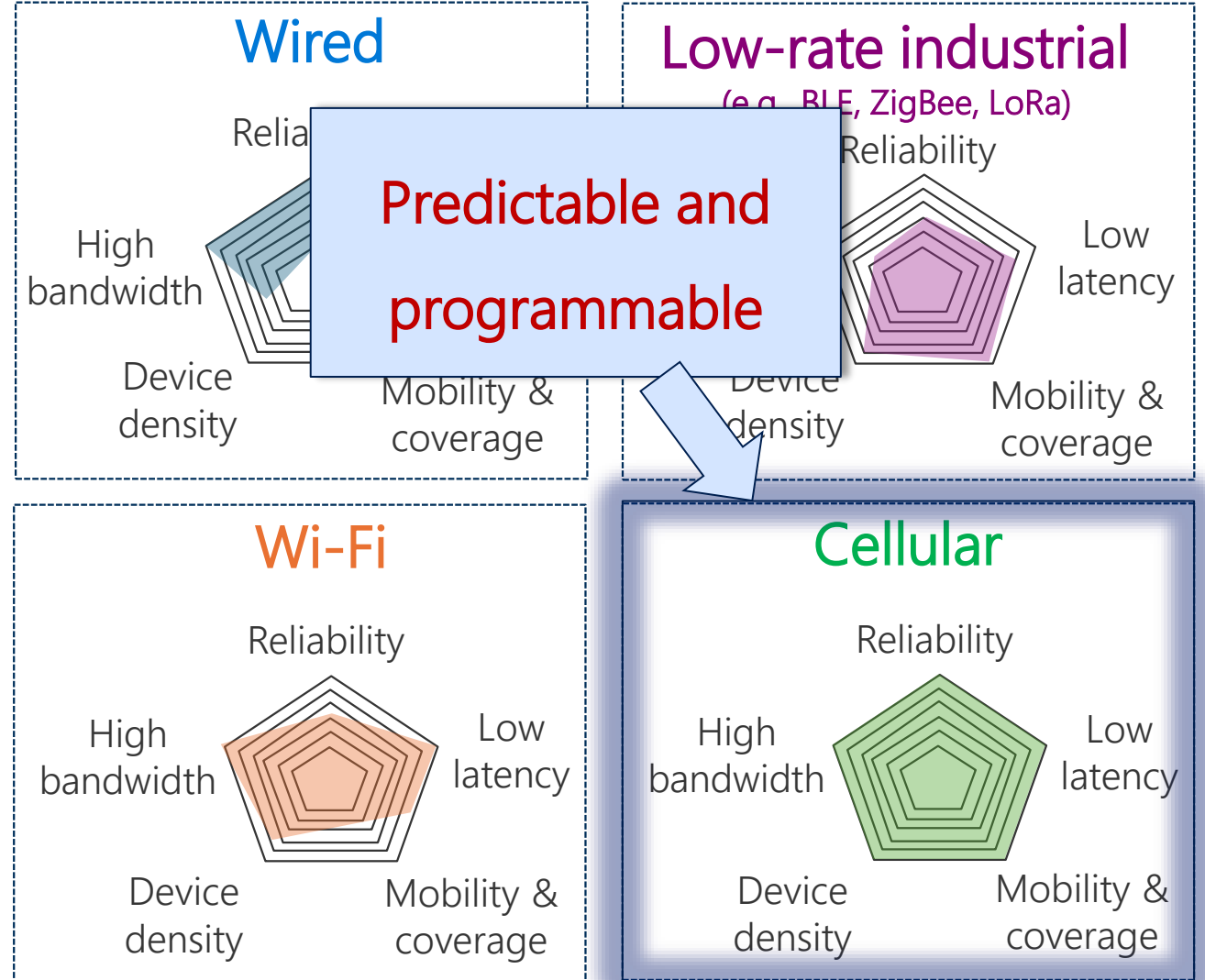
Mobility & Coverage: 

Device density: 

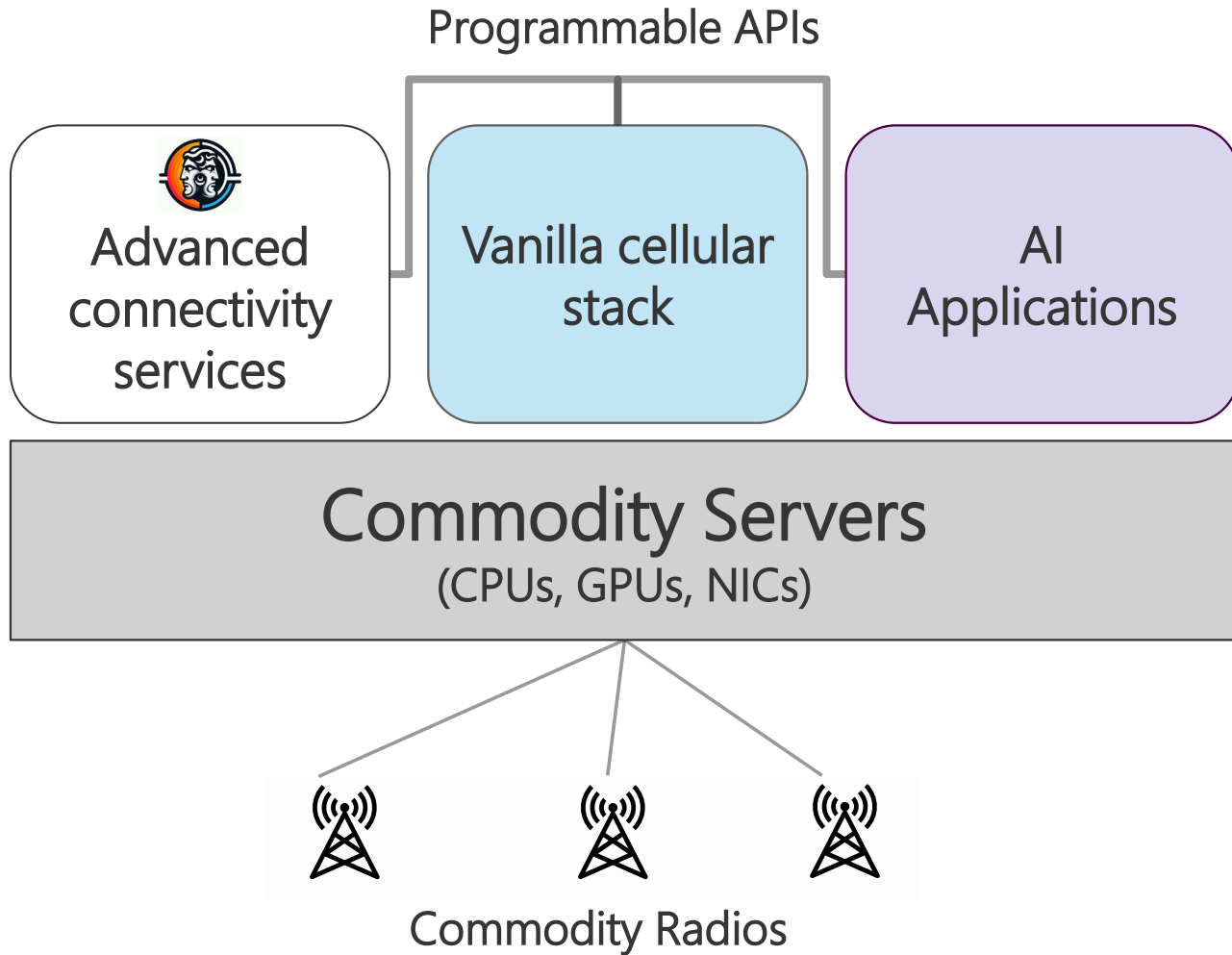
Reliability: 99,999%

Low latency: < 100ms

Security and privacy: 



Our Vision: A Converged Industrial AI Edge



New communication stack for low-latency AI applications

Benefits

- Controlled latency, throughput
- API to applications
- Low-cost deployments

Conclusions

- Cellular radio networks are exciting
 - They are virtualized and programmable
- It is easy to build a network yourself
 - It is a software system building exercise (no RF knowledge required)
- Getting the right data on time is key
 - It can enable many new AI applications
- Next-generation cellular networks can enable low-latency AI
 - Fine-grained, application-level network control





*Try our open-source dynamic probes
and real-time controller*

<https://github.com/microsoft/jbpf>



Thank you!