
Introduction to the OCUDU Project

Presented by DeepSig

Follows the Duranta and OAI UE introduction from the OpenAirInterface side

DeepSig builds AI-native software for wireless, from spectrum sensing to the 5G PHY

Founded by wireless researchers to bring deep learning into the signal chain, in software that runs in production radios and networks.

OmniSIG®

Deep-learning spectrum sensing: detects, classifies and characterizes signals on live IQ; OmniSIG Studio trains on your data.

OmniPHY-5G®

AI-native 5G PHY: neural receivers and learned air-interface components replacing conventional signal processing.

Open source and standards

Founding member of the OCUDU Ecosystem Foundation and AI-RAN Alliance; active in 3GPP and O-RAN Alliance.

80+

patents in wireless sensing and 5G optimization

11,000+

citations of the team's AI-for-wireless research

Arlington, VA

headquarters; federal and commercial programs

OCUDU is an open-source 5G and early-6G CU/DU hosted by the Linux Foundation

Built on srsRAN

SRS's field-proven CU/DU codebase, relicensed BSD 3-Clause under neutral Linux Foundation governance.

Launched March 2026

v1.0 announced at MWC. Premier members: AMD, AT&T, DeepSig, Ericsson, Nokia, NVIDIA, SoftBank and SRS, plus 21 general members and 17 research institutions.

Contributor-led

Led by a Technical Steering Committee; time-boxed working groups graduate into SIGs; GitLab issues and merge requests are the record.

DeepSig's role

Founding member. On a three-year FutureG (OUSD R&E) and NSC program with SRS, DeepSig is responsible for the accelerated, AI-ready PHY and MAC: the CUDA L1 (WG-0001 HWACCEL) and the dApp/E3 framework (WG-0002 AI-RAN).

Current OCUDU features

CU-CP, CU-UP and DU, released each April and October. Teal = new in v26.10 (shipped October 2026).

Radio and physical layer

- FDD/TDD, all FR1 and FR2 bands; up to 100 MHz (FR1) / 400 MHz (FR2)
- 256-QAM, 8T8R MIMO
- Baseline beam management and Type-II codebook CSI
- Optimized LDPC/Polar codecs for ARM Neon and x86 AVX
- NTN GEO, Release 17 NTN

Protocol stack

- All RRC and MAC procedures
- All handover and mobility types over NG and Xn, incl. conditional HO
- RRC_INACTIVE, RoHC, RAN slicing
- Positioning: NRPPa with RSRP and SRS; baseline angle-based and DL-PRS
- 2-step random access, UL pre-scheduling, configured grant

Architecture, deployment and security

- CU/DU and CU-CP/CU-UP separation; F1, E1, E2, O1; splits 6, 7.2x, 8
- Open Fronthaul split 7.2a, baseline 7.2b; M-plane via helper components
- Hardware acceleration via DPDK BBDEV
- O-RAN TIFG E2E and WG11 security compliance, validated by VALOR labs
- DTLS and improved resiliency

OCUDU feature roadmap

Planned features by release, through the end of the NSC program in October 2028.

v26.04

Initial release · shipped

- FR2
- RoHC
- RRC inactive
- Xn
- M-plane

v26.10

October 2026 · shipped

- Beamforming
- 8T8R
- NR positioning
- O-RAN split 7.2b
- 2-step RACH
- Enhanced fault tolerance
- NTN Release 17

v27.04

April 2027

- MU-MIMO
- 64T64R
- Reciprocity beamforming
- SRS antenna switching
- EPS fallback
- NTN Release 18

v27.10

October 2027

- Multiple BWP
- Carrier aggregation
- Emergency call priority
- Coverage enhancements
- Rel-17 Type-II codebook
- RedCap

v28.04

April 2028

- Dual connectivity
- UL Tx switching
- S-NPN / PNI-NPN
- Non-3GPP access
- NSA

v28.10

October 2028

- Final optimizations
- External test and validation

Resourced and scheduled, but not fixed: added capacity and contributions can shift priorities and timelines. In-scope features that matter to you are open for discussion.

OCUDU and Duranta: two Linux Foundation projects, similar aims, different emphasis

	OCUDU	Duranta (OpenAirInterface)
Emphasis	Open CU/DU for commercial deployment, broad industry adoption and advanced R&D; BSD 3-Clause	Open, research-grade RAN and UE reference stack; CSSL v1.0
Lineage	Evolved from srsRAN; hosted by the Linux Foundation (OCUDU Ecosystem Foundation)	Evolved from OpenAirInterface; hosted by LF Networking (Duranta Project)
Scope	gNB (O-CU and O-DU); UE, core and RIC from anyone, FlexRIC included	gNB, eNB and UE (NR and LTE); core from anyone; FlexRIC (near-RT RIC) originated at OAI
Interfaces	F1, E1, E2, O1; splits 6, 7.2x and 8; ZMQ virtual radio; E3 dApp interface	F1, E1, E2; splits 7.2 and 8; RFsim and ZMQ simulators; E3 dApp interface



Where they meet: an open UE plus an open gNB gives an air interface open at both ends.

dApps live in the PHY and MAC, enabling AI-RAN use cases at every timescale

Compatible with Northeastern's dApp/E3 framework, also carried by OAI's E3 agent, as E3 moves through O-RAN Alliance standardization.

Class A · 10–500 μ s

Inline PHY

In-process call on resident data. No copy.

- Neural RX, CE, equalization
- Neural SRS sounding, PRACH
- Two-sided learned modems
- **ISAC waveform co-design**

Class B · 50 μ s – 1 ms

Real-time control

Bounded record in, validated intent out, in-slot.

- MAC scheduling and allocation
- HARQ and link adaptation
- Per-slot power control
- Beam selection and prediction

Class C · 0.2 ms – seconds

Observe and advise

Never blocks the producer. One copy at most.

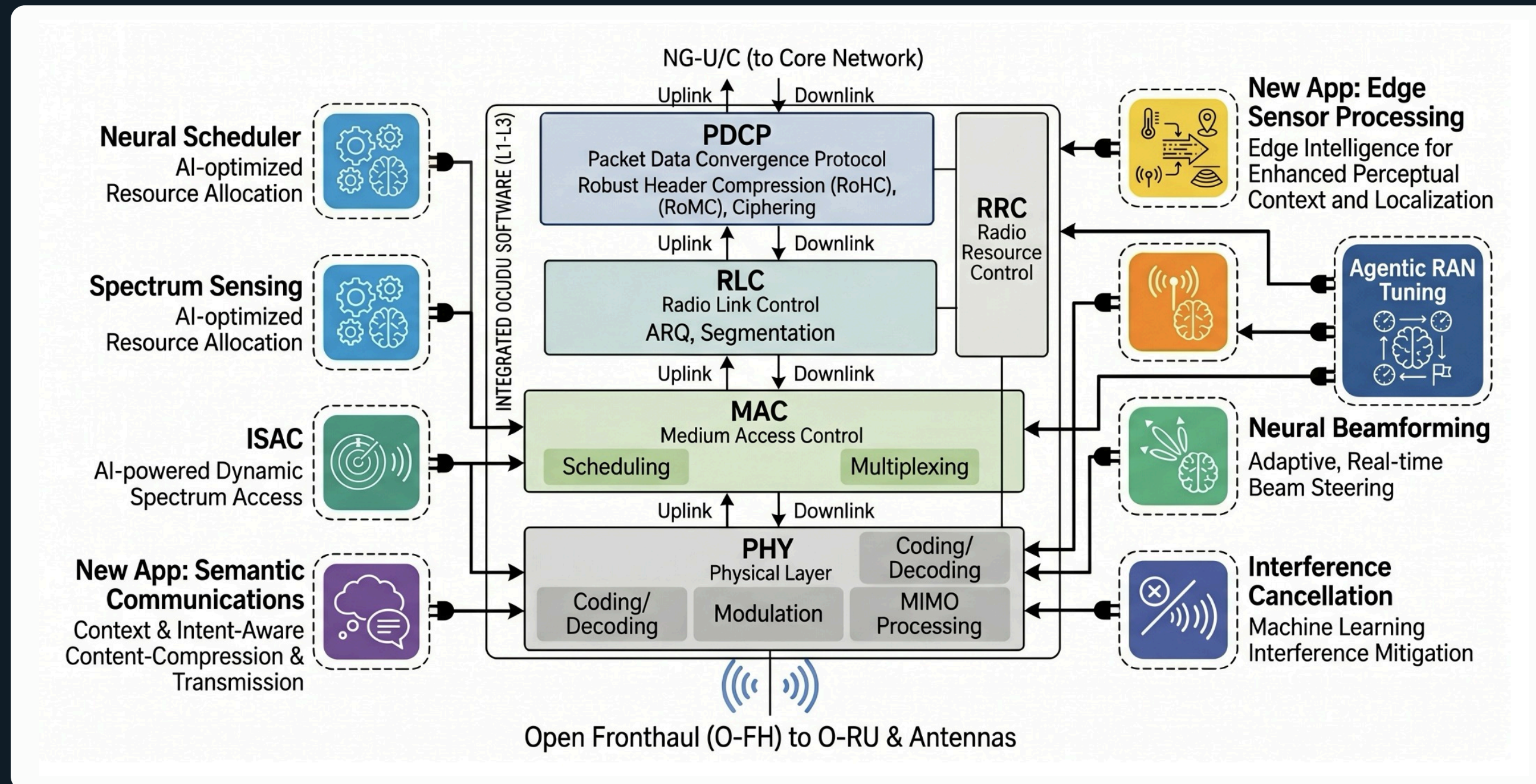
- Spectrum sensing and sharing
- **ISAC radar and sensing**
- **Positioning, channel charting**
- Uplink interference detection

Available now

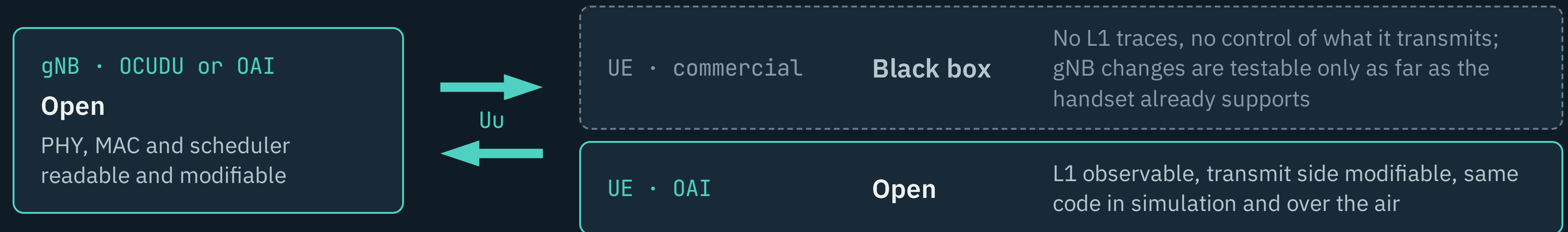
OCUDU TSC WG-0002,
BSD-3-Clause-Clear

gNB with embedded dApp runtime and E3 agent; 16-package reference SDK (CPU and CUDA); Docker quickstart with 13 tutorials, no radio or GPU needed; seven runnable use cases, incl. spectrum sensing, ISAC, inline neural receiver and learned scheduler.

Where dApps plug into the gNB



Why an open-source software UE matters



A software UE for development

- Runs without radios: full-L1 simulation or abstracted-PHY mode
- Reproducible 38.901 TDL/CDL channels and swept SNR for BLER curves
- Both ends visible: REs, LLRs, HARQ state, decoder output
- Labeled uplink traffic to train and test a neural RX, closed loop

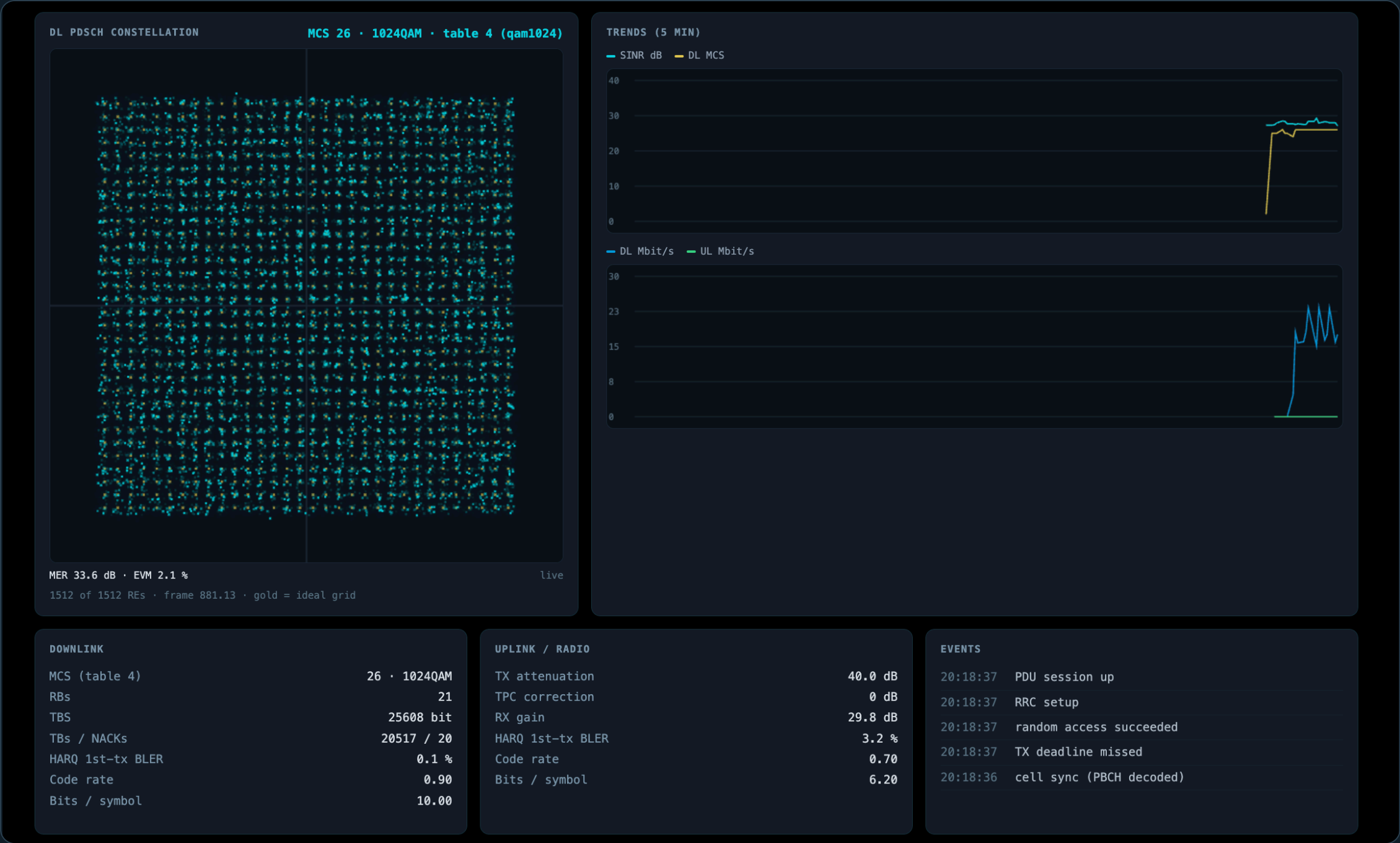
An over-the-air UE for ISAC

- Same code on USRP-class SDRs: sim and OTA loops match
- Transmit-side control: DMRS, SRS, pilot density, waveform
- Range and Doppler from the OFDM RE grid, mono- or bistatic
- Time-synchronized cooperative UE with known position and motion

An integration point for 6G PHY

- Two-sided learned waveforms: neural TX in the UE, neural RX in OCUDU
- Constellation shaping today; CSI compression and prediction next
- Co-training needs UE TX updates a closed handset won't allow
- With an open gNB (OCUDU or OAI), the air interface is open at both ends

Instrumenting the OAI UE for R&D and demos



ISAC work and next steps

Presented by DeepSig

Follows the OAI UE interoperability walkthrough and the local and over-the-air demos

ISAC with the OAI UE and an OCUDU gNB: three roles for an open UE

In a sensing experiment the UE can be the target, a second sensor, or one end of a new waveform.

Role 1 · Cooperative target

Known position, known waveform

- SRS and DMRS are the sensing waveform, seen by the gNB every slot
- SRS periodicity, bandwidth, comb and ports set from UE config
- UE-reported position and motion labels as ground truth
- Mono- and bistatic geometries with a time-synchronized UE

Role 2 · Second sensor

Both ends of the channel visible

- UE-side RX grid, CSI-RS and DMRS estimates readable
- gNB and UE channel views captured on the same slot timing
- Measurement reporting extended beyond today's CSI framework
- 38.901 simulation checked against the measured channel

Role 3 · Waveform co-design

Transmit side open to change

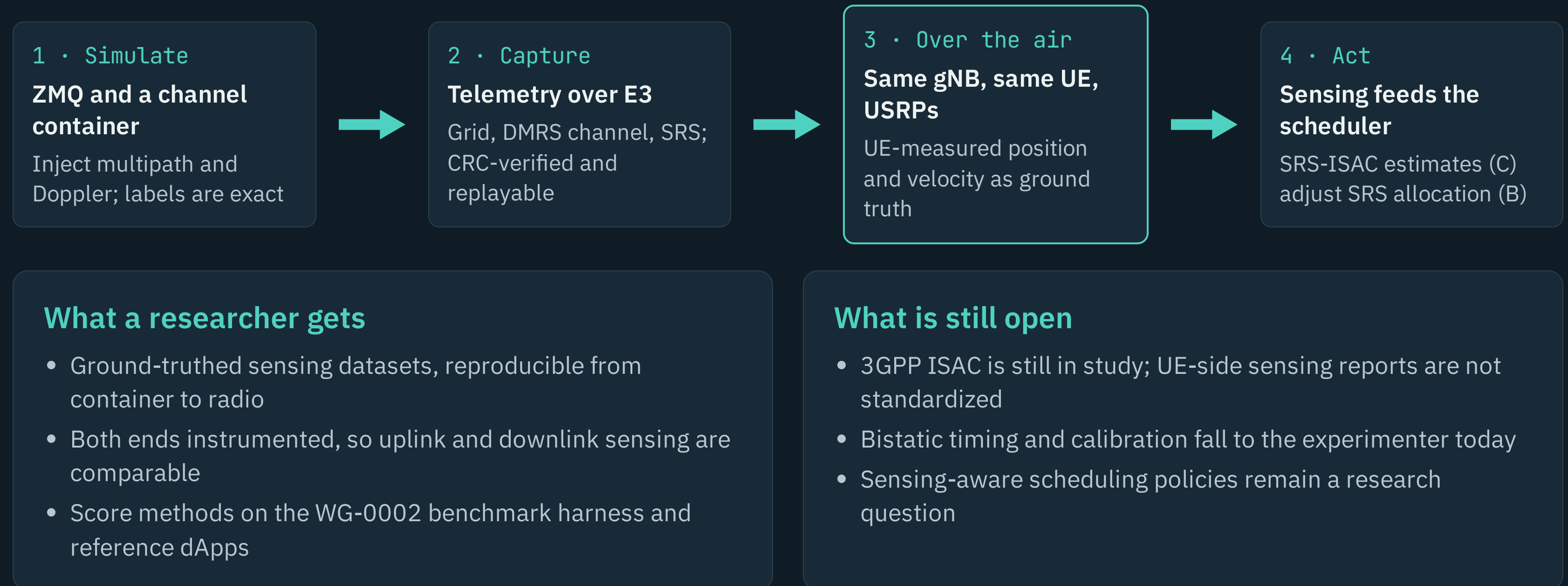
- Custom reference signals, pilot density and waveform on the UE
- Sensing-optimized SRS patterns matched to a Class A gNB dApp
- UE TX changes paired with a gNB dApp, tested end to end OTA
- Pre-standard ISAC ideas tested as code before study items close

The gNB side

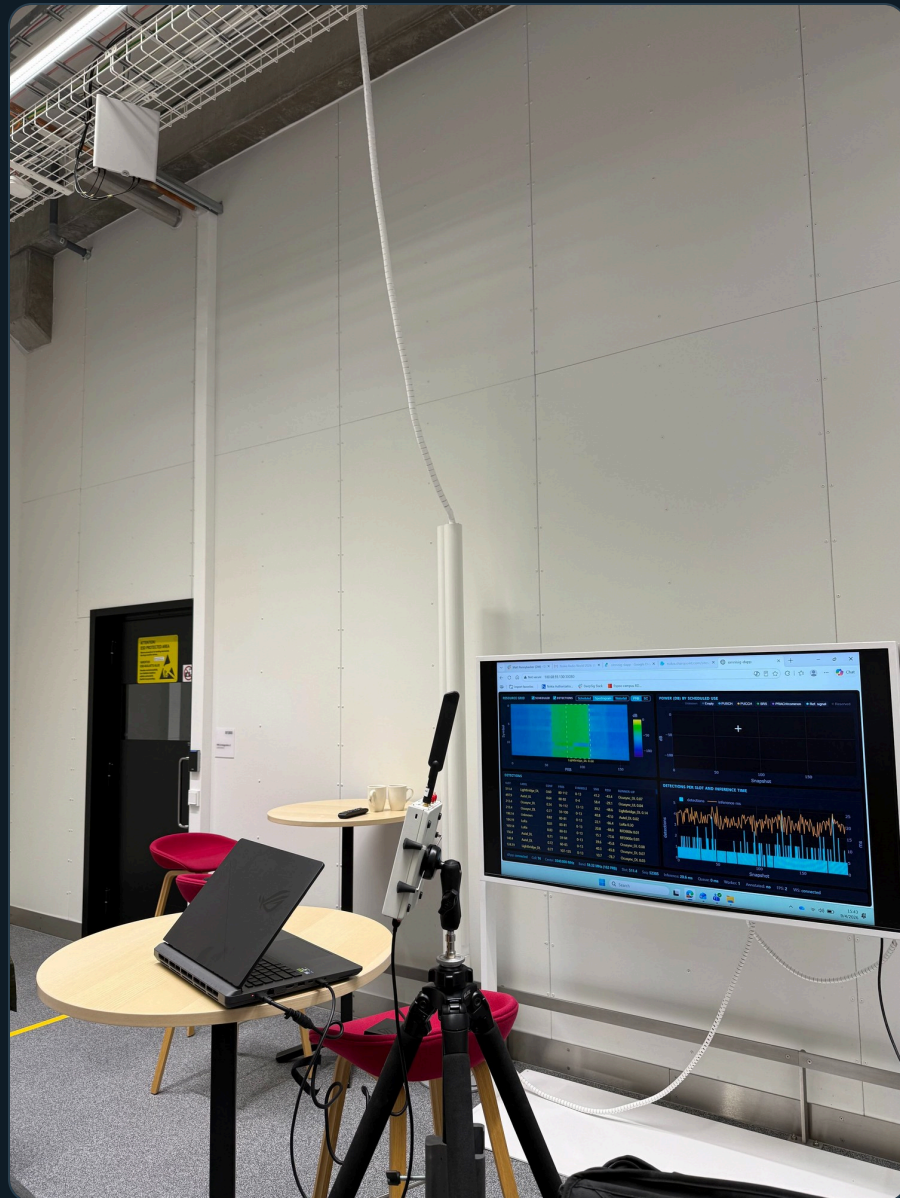
already in the dApp release

Class C SRS-ISAC dApp reads the uplink grid and SRS estimates, publishes range and Doppler over E3; SRS allocation is a Class B scheduler knob, so sensing can steer what the UE sounds.

One sensing pipeline, from ZMQ to over the air



A complementary OmniSIG® passive sensing dApp, demonstrated with Nokia



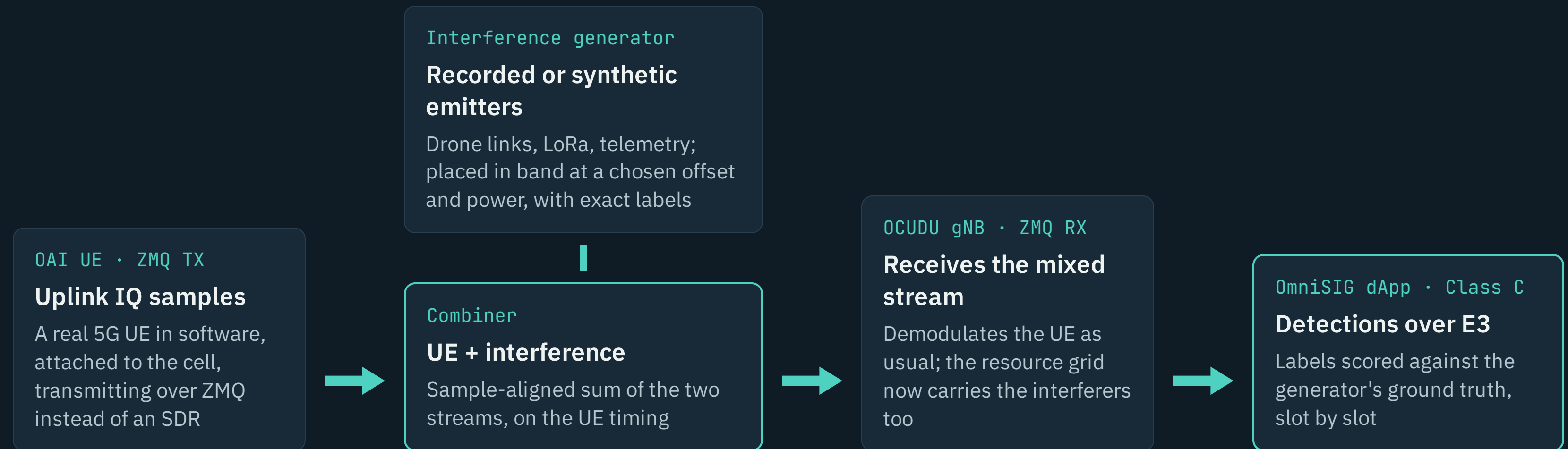
What the dApp does

- Class C dApp: reads the uplink resource grid over E3, no radio of its own
- Detects and classifies emitters sharing the band, per slot: drone control links, LoRa, telemetry radios
- Detections overlaid on the scheduled grid, with label, PRB and symbol span, SNR and confidence
- Inference in the tens of milliseconds per slot alongside the running cell

What was shown

- Developed on OCUDU, demonstrated on Nokia's gNB: both expose the same E3 interface, so the dApp ran unchanged
- Emitters brought into the band from the tripod SDR; detections appear as they transmit
- Same dApp code and SDK as the WG-0002 release

ZMQ pipeline: OAI UE plus interference into the gNB



Because the UE is open and speaks ZMQ, its transmit stream can be intercepted and mixed before the gNB sees it. The dApp is developed, regressed and scored on labeled interference in a container, then moved to the air unchanged.

Both projects are in the open and take contributions

Duranta (OpenAirInterface)

- github.com/duranta-project/openairinterface5g: code, issues, discussions, CONTRIBUTING.md
- openairinterface.org and lfnetworking.org/projects/duranta
- Mailing lists via the repository wiki; docs under doc/ (BUILD, RUNMODEM, FEATURE_SET)

OCUDU

- gitlab.com/ocudu: code; community/discussions is the record
- ocudu.org: project, members and releases
- Working groups (HWACCEL, AI-RAN): gitlab.com/ocudu/work_groups; calls open to all

The OAI UE + OCUDU tutorial

- docs.ocudu.org/tutorials/oaiue
- Covers the ZMQ virtual-radio and over-the-air modes shown today
- Tutorial feedback: gitlab.com/ocudu/ocudu_docs; code issues: OAI or OCUDU trackers

What helps most

- Run the tutorial and report what breaks, on either side of the link
- Source code on either project: UE and gNB features, reference dApps, benchmarks, platform ports
- Use cases and interface critique while the dApp interfaces are still forming

Extra: 3GPP constellation shaping

Presented by DeepSig

Backup material for Q&A or if time allows

3GPP constellation shaping: an open benchmark of 63 methods, and a result that ran as real code

The benchmark

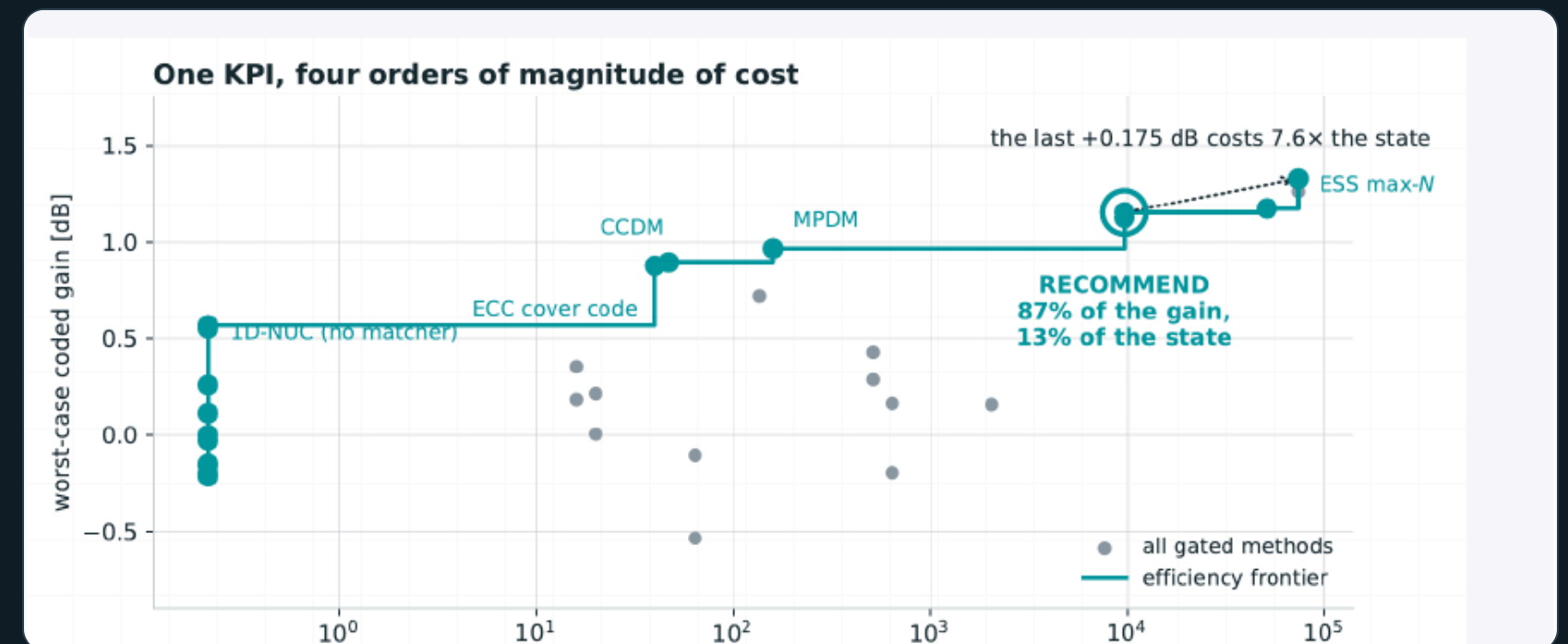
Shaping is a Release 20 study item; 51 RAN1 contributions each report gain against their own baseline. OCUDU TSC WG-0002 ran 63 methods from 17 companies in one open NR link, ranked on worst-case coded gain at 10% BLER.

The result

The answer is a frontier: the leaders sit within 0.10 dB of each other across four orders of magnitude of transmitter state. The recommended profile keeps 87% of the best gain at 13% of the state.

Ran as real code, against the OAI UE

Implemented as a per-grant mode in the OCUDU gNB, CPU and CUDA, with the OAI UE and the scheduler in the loop: +0.85 to +1.22 dB from 64- to 1024-QAM. Submitted as evidence to RAN1 #126.



Worst-case coded gain against transmitter matcher and LUT state (bytes); all gated methods, with the efficiency frontier.



A shaped constellation is a two-sided change, so the UE has to adopt it too. The OAI UE's open-source license let us add the shaping capability on the UE side ourselves.