How CNS Built its Novel RA-NN on OCUDU

Cognitive Network Solutions · OCUDU Ecosystem Foundation · 2026

ECOSYSTEM FOUNDATION

Executive Summary

Cognitive Network Solutions (CNS) and the OCUDU Ecosystem Foundation jointly demonstrate how an open, modifiable 5G CU/DU enables a new category of intrinsically intelligent RAN. By embedding a novel third CU functional split — the Intelligence Plane (CU-IP) — directly into OCUDU, CNS proved its RA-NN formulation can run inside a real protocol stack, validated that its unique PCI auto-retune algorithm can operate as a native protocol-stack function, and contributed its OCUDU stack scaffolding back upstream so every future OCUDU builder starts from a stronger baseline. Ultimately, this not only creates a new category of RAN software solution, but is proof that even a small market entrant like CNS can seamlessly innovate on top of the OCUDU open-source RAN platform, effectively leveling the playing field for both new and established players in the broader telecom ecosystem.

At a Glance

Partners	**Cognitive Network Solutions (CNS) · OCUDU Ecosystem Foundation**
Platform	OCUDU + RA-NN (OCUDU gNB · CU-IP · Kubernetes · USRP B210 · NVIDIA GPU)
Use Case	Intrinsically intelligent RAN — RA-NN / CU-IP on open 5G stack
PCI Confusions Resolved	815 —> 0 (100% elimination)
Execution Time	100 ms on a 1,500-cell, 48,320-relation network
Cells Changed	13.6% (surgical correction, not full replan)
Formulation Runtime	Protocol-stack resident — RT timescale, peer to the CU

The Challenge: Intrinsic Intelligence Has to Live Inside the Protocol Stack

The Radio Access Neural Network (RA-NN) is CNS's unique neural systems approach to a new intelligent RAN: a RAN that treats protocol interactions as the training signal for a shared latent representation of the network's physical structure.

Realizing the RA-NN vision requires it to live inside the protocol stack. To work the way it's designed to work, it imposes three hard requirements on whatever hosts it:

- It needs to consume protocol primitives, not KPIs: L3 measurement reports, RRCSetup exchanges, F1AP cell context updates — not per-cell aggregates that throw away the granularity the formulation needs.
- It must maintain a shared dynamic latent state of all network interactions: a live graph structure, consultable by handover and mobility logic at the same timescale the CU already runs at.
- It must operate at RT timescale: RA-NN intelligence functions need to emit decisions back into the protocol stack inside the same timescale the CU's protocol decisions operate in.

What Was Holding the Industry Back

Before OCUDU, four structural barriers prevented this category of work from existing at all:

- No open 5G CU/DU existed with a permissive license that could be modified and redistributed — incumbent stacks are proprietary.
- No open stack was architecturally modular enough to admit a new functional split alongside CU-CP and CU-UP without a full rewrite.
- Suitable protocol-stack scaffolding (cell-lifecycle gates, F1AP config-update procedures) that any AI-driven decision needs to safely act on a live cell simply did not exist in the open community.
- Without an active upstream community accepting contributions, any missing plumbing built in isolation remains fragile and benefits only its author.

> *"We know that carriers want this type of intelligence, but there is no way for them to deploy our solution when the 5G stacks they are buying are all proprietary."*
>
> **— Terence Li, co-founder, CNS**

The Solution: CU-IP — the Intelligence Plane

CNS implemented a third CU functional split — the Centralized Unit Intelligence Plane (CU-IP) — running alongside CU-CP and CU-UP. The CU-IP is the architectural home for the RA-NN formulation inside the RAN protocol stack. Two new internal interfaces connect it to the rest of the RAN:

- RANN-P (Radio Access Neural Network — Primitives): inbound, CU-CP -> CU-IP. Carries per-UE measurement reports, cell context, and neighbor state — the protocol primitives the formulation trains on.
- RANN-D (Radio Access Neural Network — Decisions): outbound, CU-IP -> Management Plane. Carries decisions out of the CU-IP into a declarative cell-state mechanism to apply them to the running CU and DU.

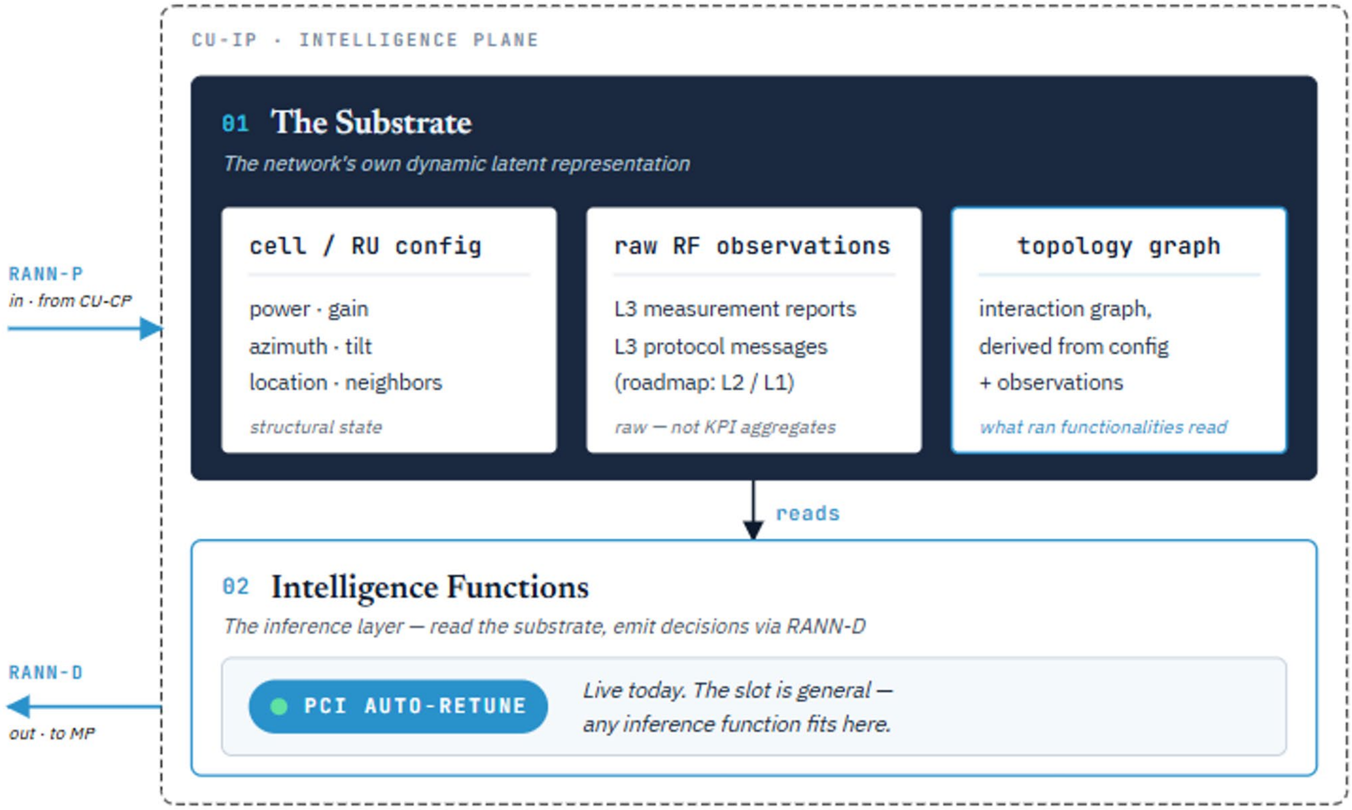

Fig 1) CU-IP internals: The RA-NN substrate (cell/RU config, raw RF observations, topology graph) feeding the intelligence functions, decisions out via RANN-D

Solution Components

Component	Description
cognitive-network-solutions/cns-ocudu/ocudu	CNS's downstream distribution of OCUDU. Carries the main OCUDU protocol-stack + CU-IP adapter libraries with additional runtime CU/DU configuration components.
cognitive-network-solutions/cns-ocudu/cuip	Public reference implementation of the CU-IP: RA-NN substrate + PCI auto-retune as the first intelligence function.

CNS implemented its solution as a split architecture on Kubernetes. Not only did CNS split the CU from the DU, but further split the CU into its three separate components for a truly modular deployment of the gNB’s logical functions. The Kubernetes Management Plane mediates the substrate’s decisions to make them persistent, not ephemeral: each decision is written back to the deployment manifests, so any change survives pod restarts and carries forward into every new rollout.

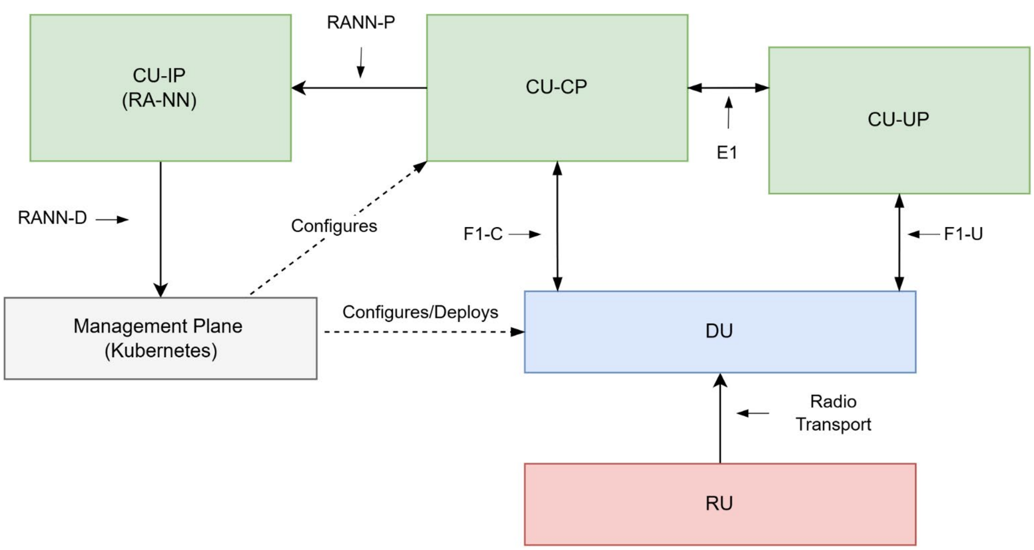

Fig 2) CNS RA-NN deployed using the CU-IP as an OCUDU split mode architecture on Kubernetes

OCUDU
ECOSYSTEM FOUNDATION

OCUDU's Critical Contribution

OCUDU is the first open 5G CU/DU stack modular enough for a third functional split to be added by an outside contributor. It provided the open, permissive-licensed protocol stack that accelerated interoperability on real hardware and COTS UEs. It served as the foundation that demonstrated real workload portability in a live RAN environment. Without it, RA-NN would have remained a published paper.

Results

The original RA-NN-driven PCI retune algorithm was benchmarked against standard graph-coloring algorithms on a 1,500-cell simulated network with 48,320 neighbor relations, starting from a deliberately degraded state (815 confusions, 46 collisions):

Algorithm	Result	What It Means
RA-NN (this work)	815 -> 0 confusions · 100 ms · 13.6% cells changed	Complete elimination — surgical correction, not a full replan
Greedy coloring	815 -> 1,500 confusions · 33 ms	Drives every node to confused — wrong objective
DSATUR (C++)	815 -> 1,500 confusions · 102 ms	Same failure mode as greedy at similar speed
TabuCol	815 -> 703 confusions · 342 ms	Cannot see confusions — stops with 703 remaining
BRKGA	815 -> 608 confusions · 1.77 s	Best classical result, still 608 confusions remaining

Since the RA-NN is already proven to handle the scale, CNS created a smaller hybrid deployment scenario to validate this intelligence can be intrinsic to the OCUDU protocol stack.

One cell broadcasting from a USRP B210 with a COTS UE attached, end-to-end, plus nine virtual neighbor cells with realistic signal metrics computed using NVIDIA's Sionna framework. The virtual neighbor signal measurements were injected directly into the real L3 measurement report the COTS UE was reporting, making the UE appear as if it was measuring 10 unique cells to the CU-IP.

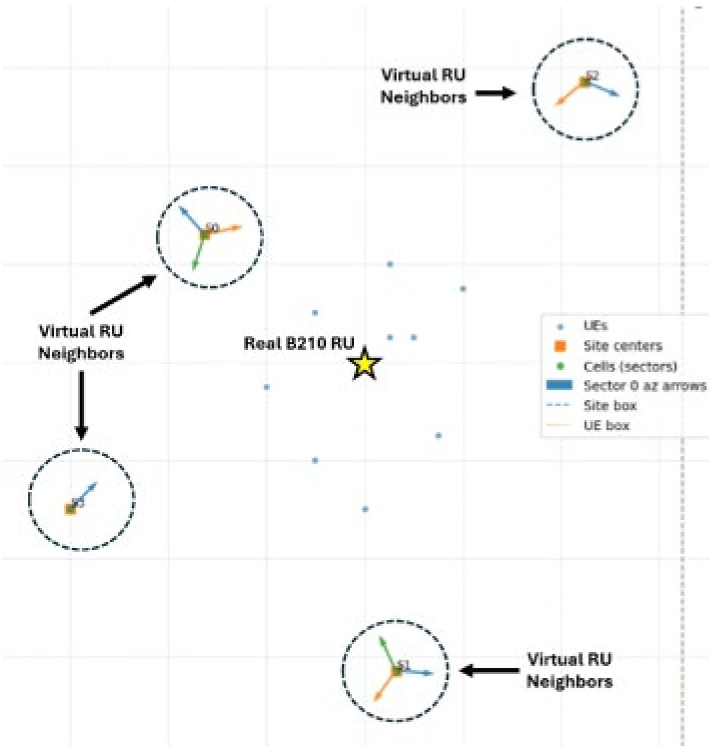

Fig 3) Hybrid-b210 scenario layout: real B210 RU at center, virtual neighbor RUs at defined positions/azimuths.

The RA-NN detects any PCI related issue on the next report cycle and emits a RANN-D decision to update the config:

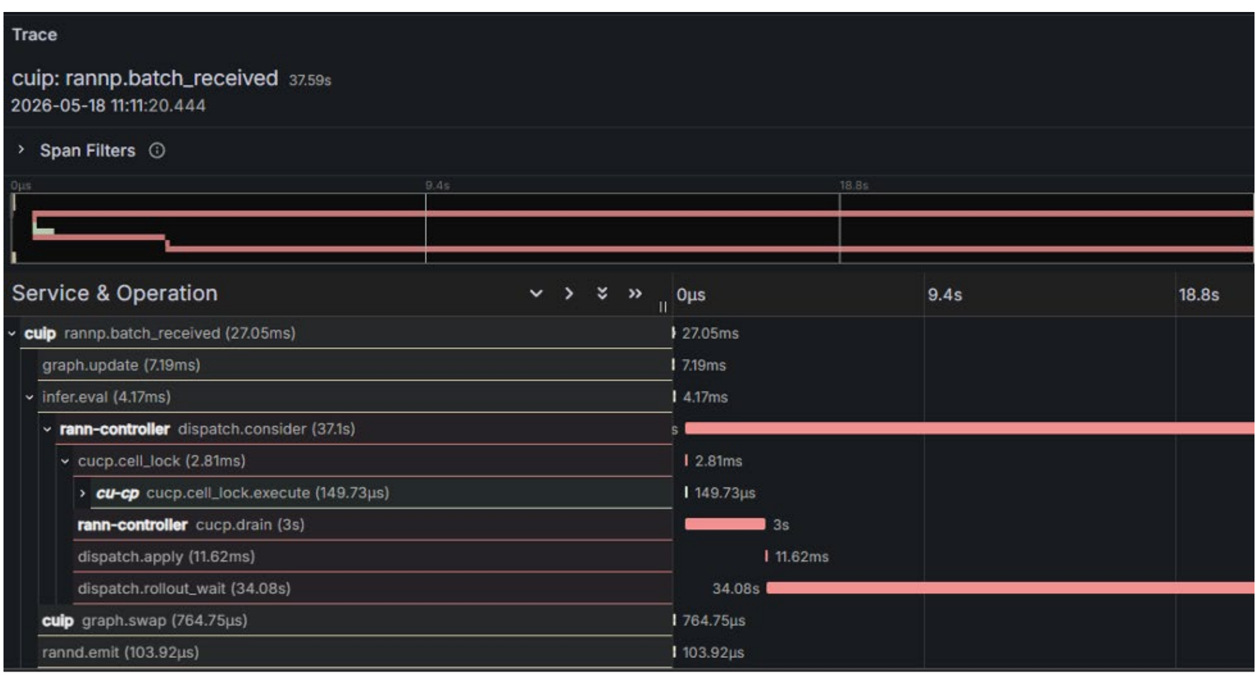

Fig 4) Live OTel trace of the CU-IP protocols interwoven with the 3GPP protocols of the CU-CP during an automatic PCI retune

The decision-making chain is fast:

SPAN	DURATION	WHAT IT DOES
cuip rannp.batch_received	27.05 ms	Substrate ingests a RANN-P batch of L3 primitives end-to-end
graph.update	7.19 ms	Updates the live interaction graph from the new primitives
infer.eval	4.17 ms	The PCI confusion inference itself
graph.swap	764 µs	Atomic swap of the substrate's served snapshot
rannd.emit	104 µs	Decision dispatched out of CU-IP to the Management Plane via RANN-D

Applying this decision is where the operational reality of a structural RF change shows up. CNS implemented a new cell locking mechanism to the OCUDU stack to protect user experience during service impacting changes. They are actively contributing this mechanism upstream to OCUDU:

SPAN	DURATION	WHAT IT DOES
cucp.cell_lock	2.81 ms	Controller initiates the lock
cucp.cell_lock.execute	**149.73 µs**	Cell-deactivate signal firing inside CU-CP
cucp.drain	~3 s	UE drain — gated by 3GPP RRC procedures, not by us
dispatch.apply	11.62 ms	New PCI applied, cell reactivated
dispatch.rollout_wait	~34 s	K8s rollout wait for the DU pod to pick up the new config

What this validates:

- Standard 3GPP procedures (RRC, mobility, F1AP, NGAP) run normally while the CU-IP/RA-NN is active.
- The CU-IP ingests protocol primitives at the rate the CU-CP produces them without applying back-pressure.
- RA-NN decisions dispatching from CU-IP via RANN-D do not starve or deadlock the protocol path.
- Structural RF config changes to the CU/DU deployments can be made at runtime without impacting the customer experience.

> *"Our goal isn't intelligent applications on top of the protocol stack, rather a network that learns the nuances of its operating environment and adapts to them on its own."*
>
> **— Michael Chiaramonte, co-founder, CNS**

What Makes This Approach Different

Current RAN intelligence relies on external application surfaces and offline planning tools — and RF engineers are still performing periodic online fine-tuning after all of that. This work breaks that model:

- **Protocol-stack residency:** Intelligence lives inside the CU, not above it — operating at the same timescale as the protocol stack itself, not on a separate observability pipeline.
- **Surgical decisions:** RA-NN changes only the cells that need to change with the precision of a human engineer, rather than replanning the entire network — a consequence of its dynamic substrate.
- **Task absorption, not augmentation:** PCI planning hasn't become better-managed — it has stopped being an activity at all. Whatever else the RA-NN subsumes follows the same pattern.
- **Vendor-agnostic and open:** The CU-IP consumes 3GPP-defined protocol primitives, not vendor-specific extensions or KPIs. The same CU-IP runs identically regardless of which radio, UE, or 5GC is in the loop.
- **Upstream contributions benefit everyone:** Cell safe shutdown features, new config-update procedures, and runtime control command surfaces needed for intelligent RAN are general-purpose to any OCUDU deployment. By contributing these back upstream, anyone that wants to mutate cell parameters under live traffic can use the same mechanisms CNS built for its RA-NN.

OCUDU
ECOSYSTEM FOUNDATION

Next Steps: From Proof of Concept to Full Deployment Package

This category has gone from 'impossible outside the incumbents' to 'buildable in the open.' And buildable isn't finished: an open protocol stack with an intelligence plane inside it is the foundation of a network, not yet the whole E2E package. From here, CNS is building in three directions:

1. **Advancing the RA-NN:** The RA-NN must continue to improve upon its strong foundation to tackle more critical RAN functionality. Next on the roadmap: mobility management, load balancing, smart beamforming and cross-cell interference coordination.
2. **Building Out a Full E2E Distribution Package:** CNS is building Racora: an open E2E deployment distribution with OCUDU as its core protocol stack and RA-NN driving the intelligence. Because the RA-NN is capable of maintaining the live state of every cell and neighbor relation, it can progressively absorb the management and interoperability surface, simplifying a full open RAN software deployment to a single installable package.
3. **Contributing Upstream:** CNS believes that to drive industry adoption of the RA-NN and open source RAN in general, much of its work should be contributed back upstream to the OCUDU EF. CNS is actively contributing its cell-lock lifecycle mechanism and runtime remote-command surfaces, and plans to further its contributions by generalizing the RANN-P/RANN-D interfaces to be more implementation-agnostic and proposing its CU-IP as a reference architecture for AI-native RAN under OCUDU.

> *"The interesting thing isn't that we validated the RA-NN approach, it's that we couldn't have built it without the existence of OCUDU."*
>
> **— Michael Chiaramonte, co-founder, CNS**

GET INVOLVED

Join the OCUDU community at ocudu.org to help drive towards a new open source standard for the next generation of telecom deployments!

Interested in how CNS' intelligence functions on the roadmap and what the CU-IP and RA-NN can do for your intelligent-RAN approach? Try an early version of Racora on CNS's GitLab here.

OCUDU EF's Getting Started Guide | ocudu.org | cnscellular.com