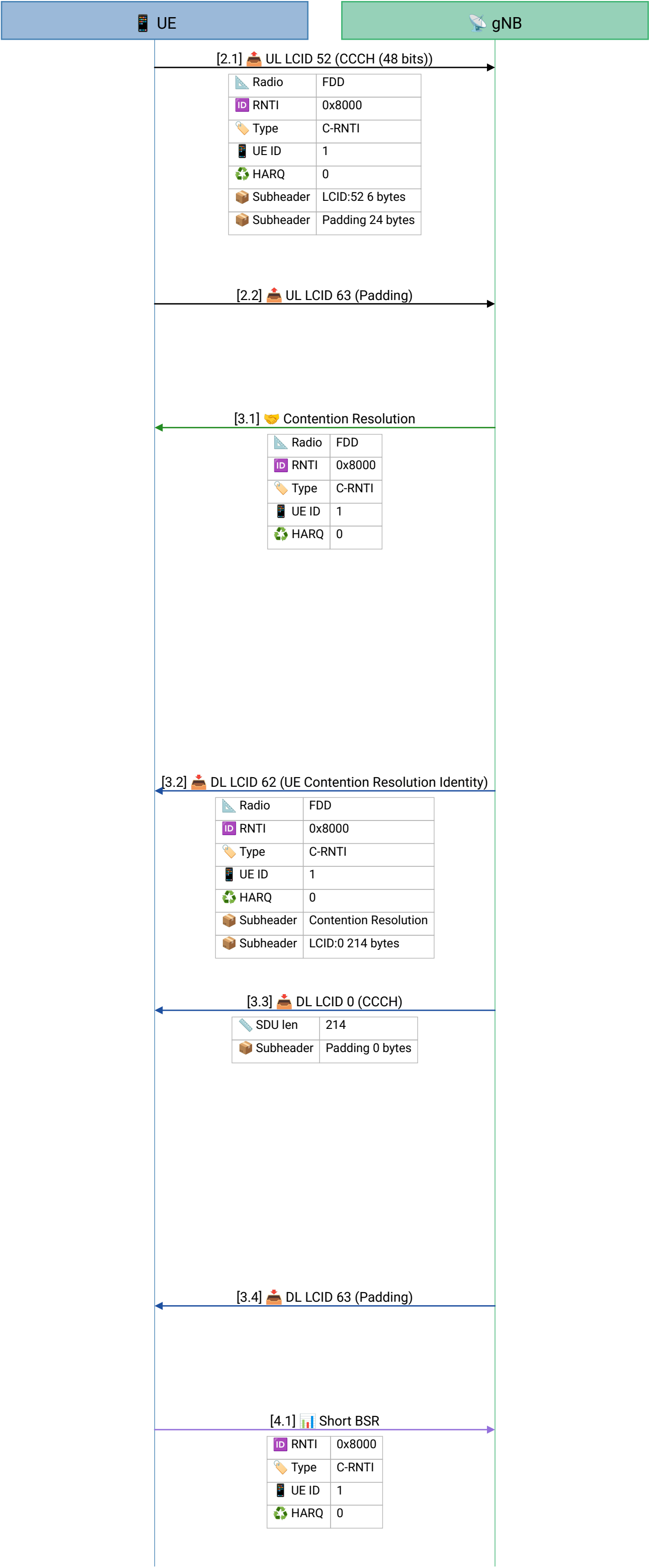




[Frame 2.1] Msg3 — the UE speaks for the first time. LCID 52 is UL-SCH shorthand for *CCCH of size 48 bits* (TS 38.321 Table 6.2.1-2), and it is the only logical channel a UE may use before the network has agreed who it is. The subheader carries no length field, because 48 bits *is* the length: six bytes, 1b34056b1846, which decode as an RRC RRCSetupRequest with ue-Identity randomValue b34056b184 and establishmentCause mo-Signalling.

Those six bytes are the whole point of the next frame. Several UEs can pick the same random-access preamble in the same slot, be given the same uplink grant, and transmit Msg3 on top of one another. Nothing so far has singled this UE out. Remember the byte string.

[Frame 2.2] 24 bytes of padding, LCID 63. The grant that carried Msg3 was sized by the Random Access Response — a message this session does not contain, because it is addressed to RA-RNTI 0x0500 rather than to a UE identity. The scheduler had no idea how large the UE's first message would be, so the fit is approximate and MAC fills the remainder. This is the first of 57 uplink padding subheaders.

[Frame 3.1] Msg4 — contention resolution. The gNB sends back a *UE Contention Resolution Identity* MAC control element whose payload is 1b34056b1846 : a byte-for-byte echo of the CCCH SDU the UE sent 0.708 ms earlier.

TS 38.321 §5.1.5 makes the rule explicit. A UE receiving this control element compares it against the CCCH SDU it transmitted in Msg3. On a match it declares contention resolution successful, stops *ra-ContentionResolutionTimer*, and — for a random access started from idle — promotes the Temporary C-RNTI it was given in the Random Access Response into its own C-RNTI. Any UE that transmitted a *different* Msg3 sees a mismatch, loses, and starts random access again.

One wrinkle worth naming, because it is easy to misread off the diagram. The Random Access Response in frame 1 assigns Temporary C-RNTI 0x1001, yet every frame from Msg3 onward is stamped C-RNTI 0x8000 in the capture's own MAC-NR context. The promotion is therefore *not* visible here as a change in the RNTI field. You read contention resolution from the identity echo, not from the identifier.

[Frame 3.2] The same control element seen as a subheader: LCID 62 on DL-SCH is *UE Contention Resolution Identity* (TS 38.321 Table 6.2.1-1) — a fixed-size control element with a six-octet payload and no length field.

This double arrow is a reading habit worth acquiring early. Every Short BSR, every Single-Entry PHR and both Contention Resolution elements in this flow are drawn twice: once as a dedicated control-element arrow with the fields expanded, and once as the generic per-LCID arrow that shows where the element sat in the MAC PDU. 105 of the session's 475 arrows are that second copy, so the real PDU count is 370 subheaders.

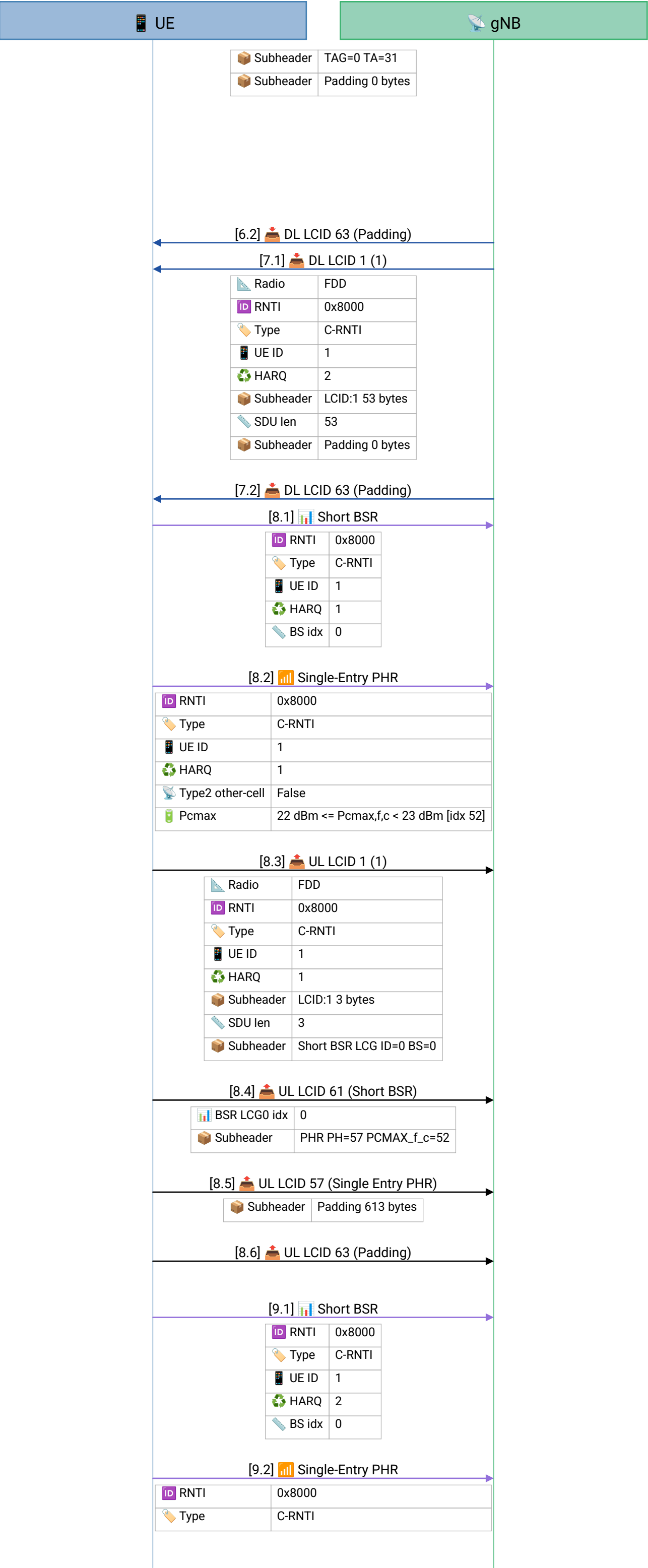
[Frame 3.3] And multiplexed behind the control element, in the same transport block: LCID 0 — CCCH on DL-SCH — carrying a 214-byte SDU. That SDU is the RRC RRCSetup, the message that gives the UE SRB 1 and, as this walk keeps coming back to, the entire MAC configuration it will run for the rest of the connection.

This is MAC's job stated in one frame. Contention resolution and connection setup are two unrelated procedures at two different layers, and MAC delivers them in one transmission because it can: a MAC PDU is a concatenation of subheaders, and each subheader independently names either a logical channel or a control element (TS 38.321 §6.1.2).

Note the order — control element first, SDU second. That is the downlink rule, and the uplink's is the mirror image. Watch for it in the next frame.

[Frame 3.4] The padding subheader that closes the block — and the first instance of a pattern that holds for the whole session. It is LCID 63 with **zero** bytes after it: one byte of R/R/LCID and nothing more. Add up every downlink padding subheader in these 144 frames and you get 64 of them and **0 bytes** of padding payload. The uplink total is 30955 bytes. The last section of this walk is about why.

[Frame 4.1] The first Buffer Status Report — and it asks for nothing. A Short BSR on Logical Channel Group 0 with buffer-size index 0, which the 5-bit level table in TS 38.321 §6.1.3.1 defines as literally BS = 0. That reads like a wasted control element until you notice what shares the transport block with it. The UE had one RLC PDU queued, the grant was large enough to take it whole, and MAC assembled both into the same PDU. By the time the buffer-size field was



Timing advance is how a cell keeps every UE's uplink arriving inside the same symbol boundary despite different propagation delays – the further away the UE, the earlier it must transmit. TS 38.213 §4.2 defines this control element's value as a *relative* adjustment around a midpoint of 31, so a command of 31 asks for no change at all.

All 23 Timing Advance Commands in this session carry 31, and they arrive on a metronome. Watch for them again at frames 120 to 127.

[Frame 7.1] 💡 DL-SCH MAC PDU – gNB->UE on PDSCH
(generic fallback for LCIDs not matched by a higher-priority template)

[Frame 8.1]  Short Buffer Status Report – UE reports queued UL data for one Logical Channel Group

[Frame 8.2] 💡 Power Headroom Report – UE reports transmit power margin remaining below P_{cmx}

[Frame 8.3] 💡 UL-SCH MAC PDU — UE->gNB on PUSCH
(generic fallback for LCIDs not matched by a higher-priority template)

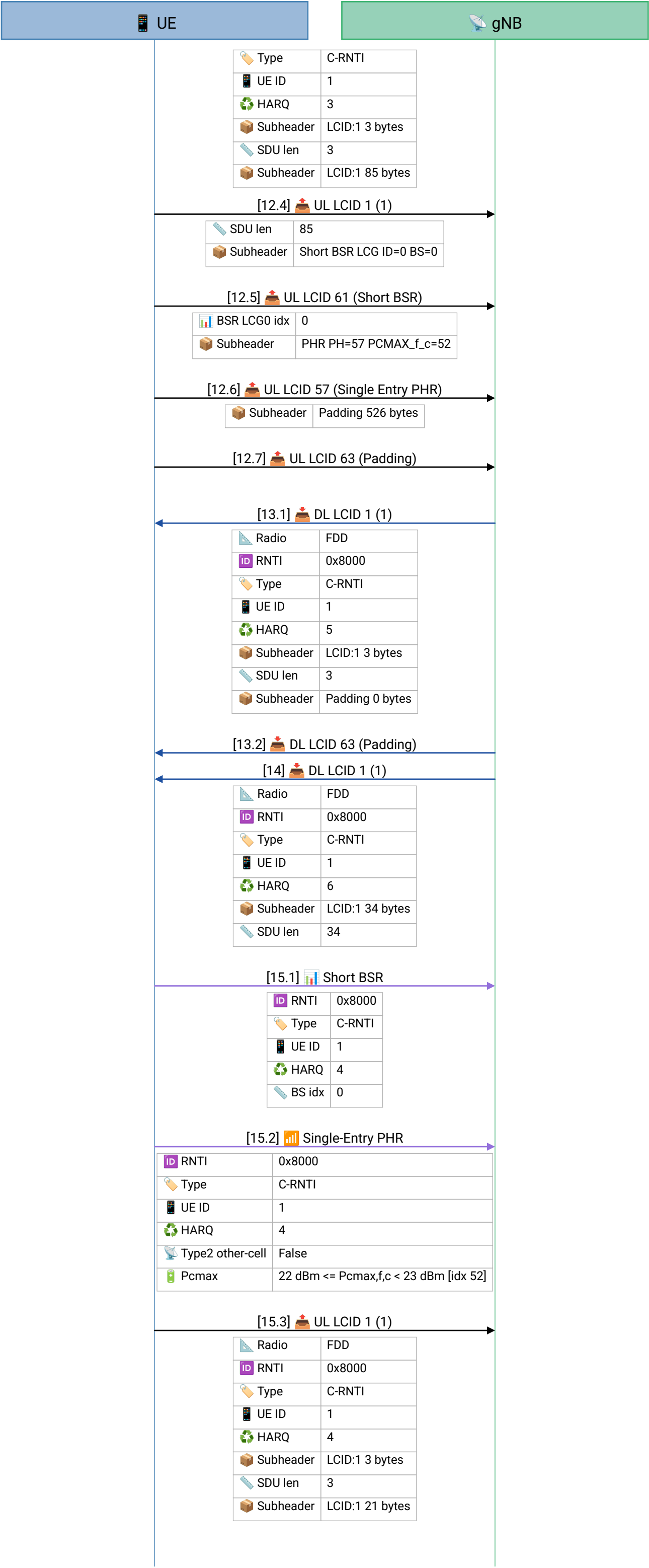
[Frame 8.4] 💡 UL-SCH MAC PDU – UE->gNB on PUSCH
(generic fallback for LCIDs not matched by a higher-priority template)

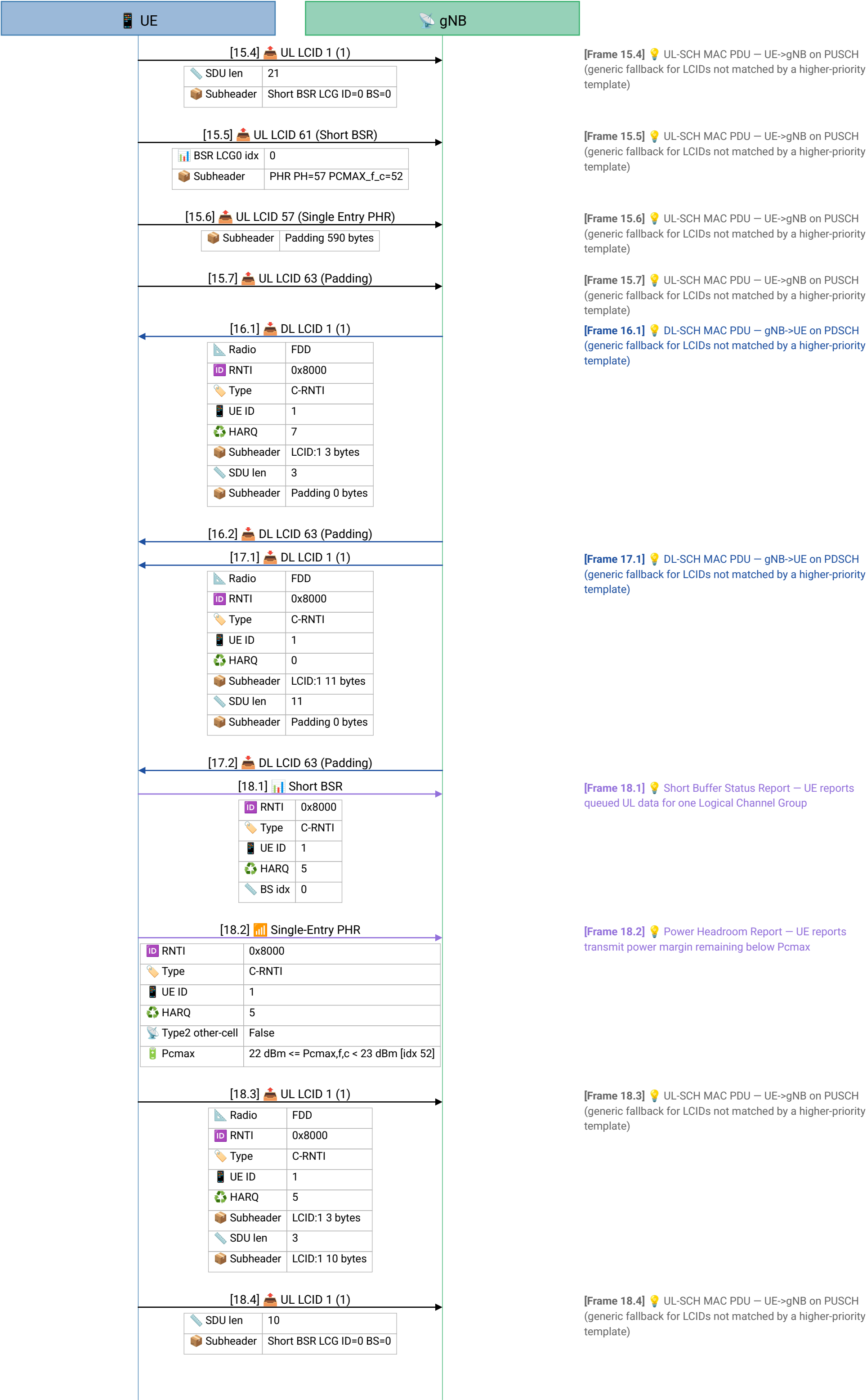
[Frame 8.5] 💡 UL-SCH MAC PDU – UE->gNB on PUSCH
(generic fallback for LCIDs not matched by a higher-priority template)

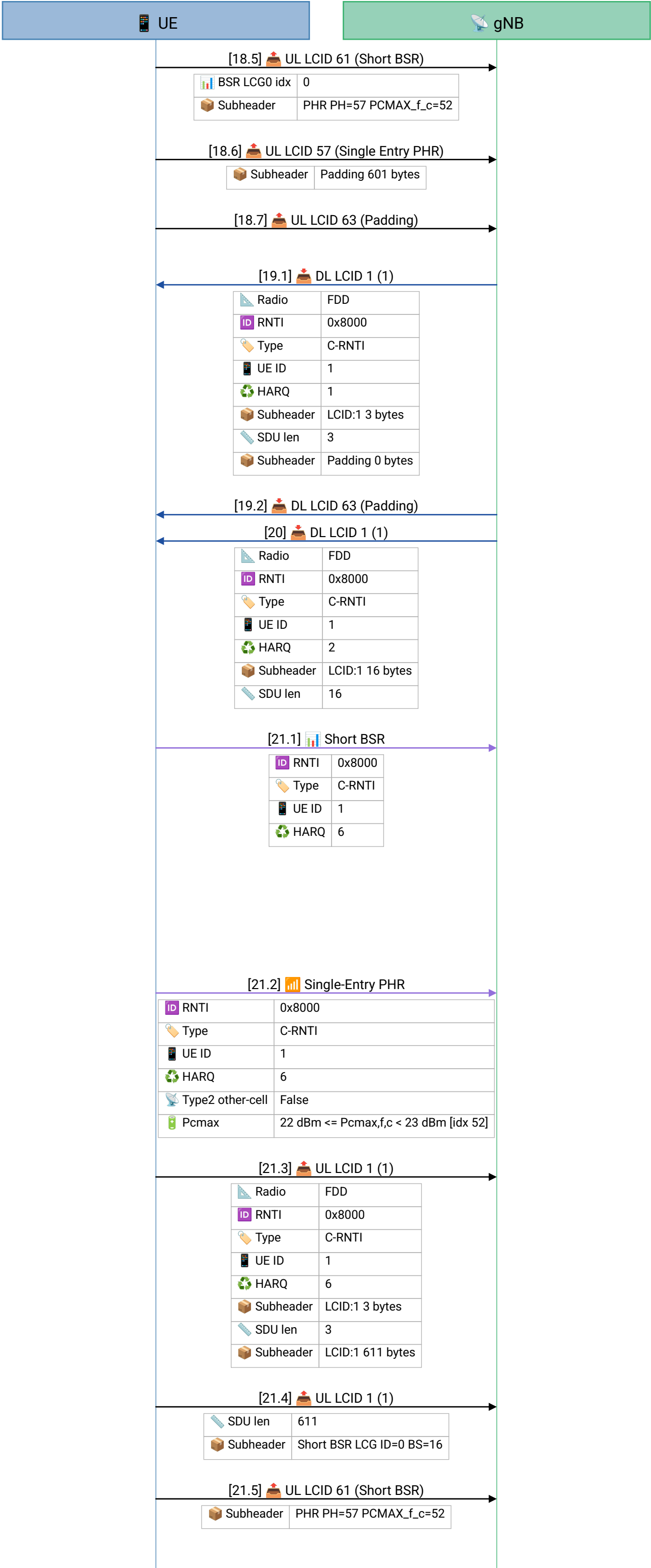
[Frame 8.6] 💡 UL-SCH MAC PDU – UE->gNB on PUSCH
(generic fallback for LCIDs not matched by a higher-priority template)

[Frame 9.1]  Short Buffer Status Report – UE reports queued UL data for one Logical Channel Group

[Frame 9.2] 💡 Power Headroom Report – UE reports transmit power margin remaining below P_{cmx}







[Frame 18.5] 💡 UL-SCH MAC PDU — UE->gNB on PUSCH (generic fallback for LCIDs not matched by a higher-priority template)

[Frame 18.6] 💡 UL-SCH MAC PDU — UE->gNB on PUSCH (generic fallback for LCIDs not matched by a higher-priority template)

[Frame 18.7] 💡 UL-SCH MAC PDU — UE->gNB on PUSCH (generic fallback for LCIDs not matched by a higher-priority template)

[Frame 19.1] 💡 DL-SCH MAC PDU — gNB->UE on PDSCH (generic fallback for LCIDs not matched by a higher-priority template)

[Frame 20] 💡 DL-SCH MAC PDU — gNB->UE on PDSCH (generic fallback for LCIDs not matched by a higher-priority template)

[Frame 21.1] The first Buffer Status Report that actually asks for something. LCG 0, buffer-size index 16, which the level table in TS 38.321 §6.1.3.1 reads as $1038 < BS \leq 1446$ bytes still queued after this transmission.

This is the BSR doing its real job. The UE has just been asked for its radio capabilities — a message far too large for one uplink grant — so RLC has cut it up and MAC is sending the first slice. The BSR riding alongside tells the scheduler that between one and one-and-a-half kilobytes remain, and the scheduler grants again immediately.

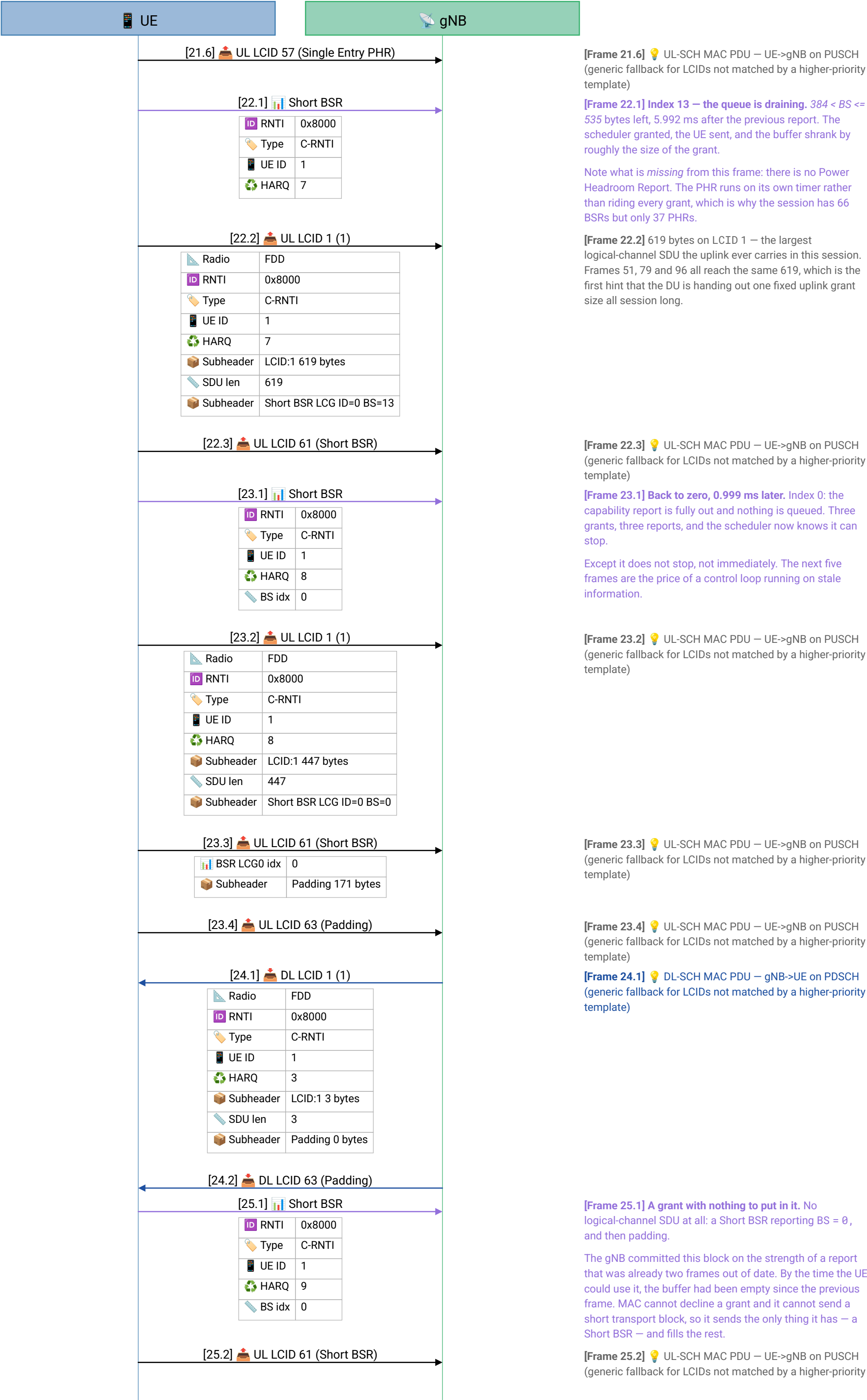
Follow the index down across the next two frames: 16, then 13, then 0. That three-step drain is the entire feedback loop in miniature.

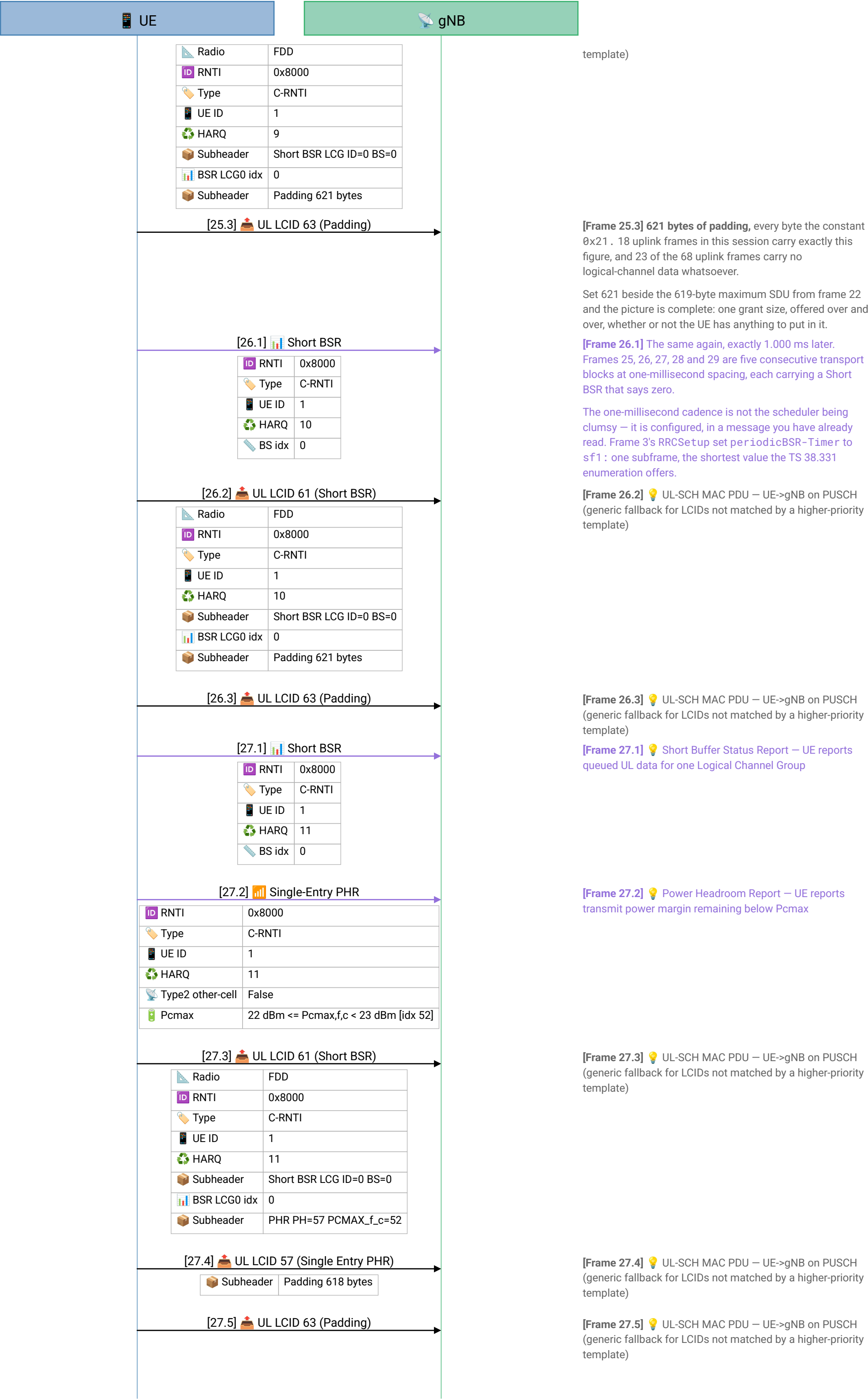
[Frame 21.2] 💡 Power Headroom Report — UE reports transmit power margin remaining below Pcmax

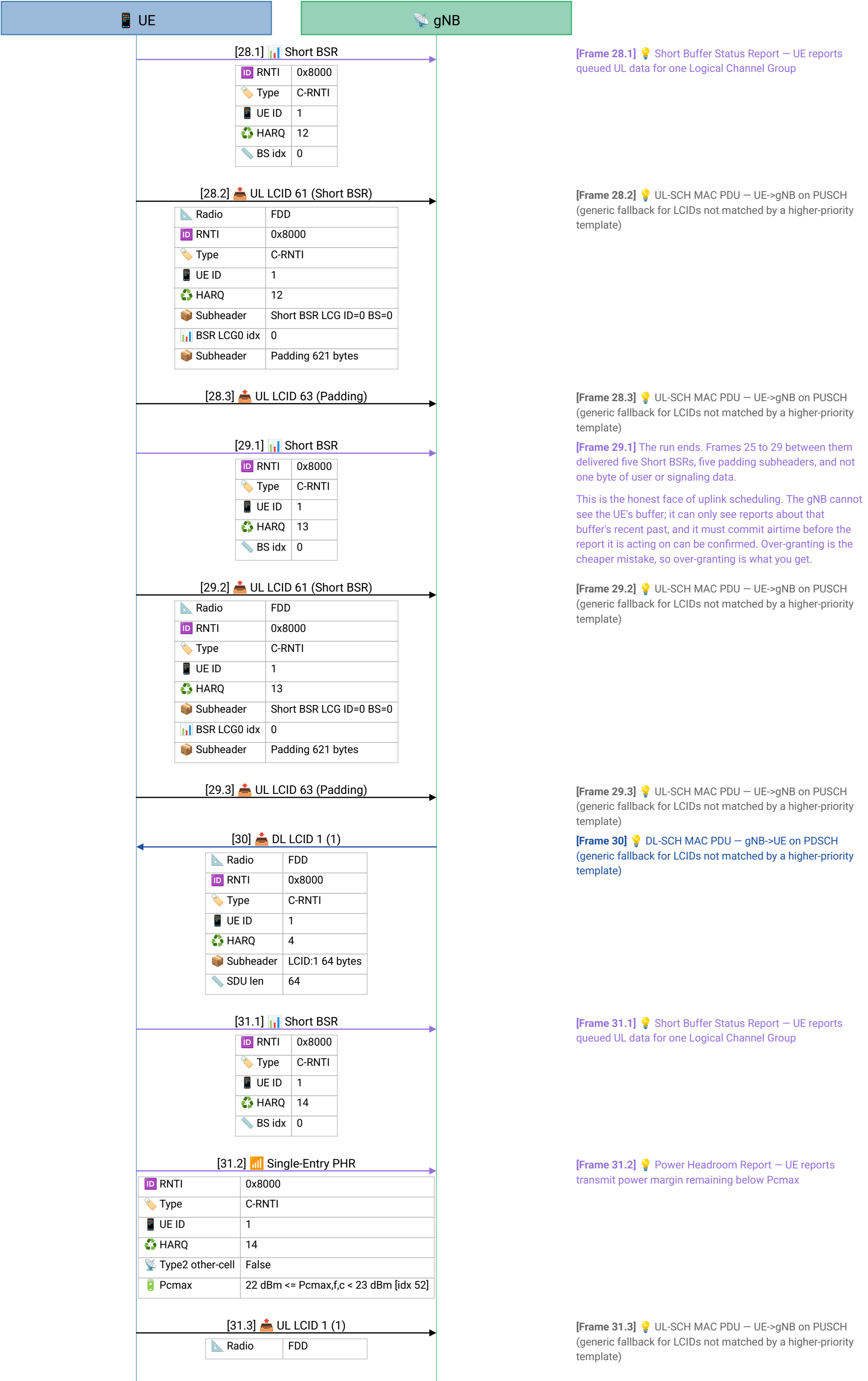
[Frame 21.3] The three-byte RLC status report riding along at the front of the PDU.

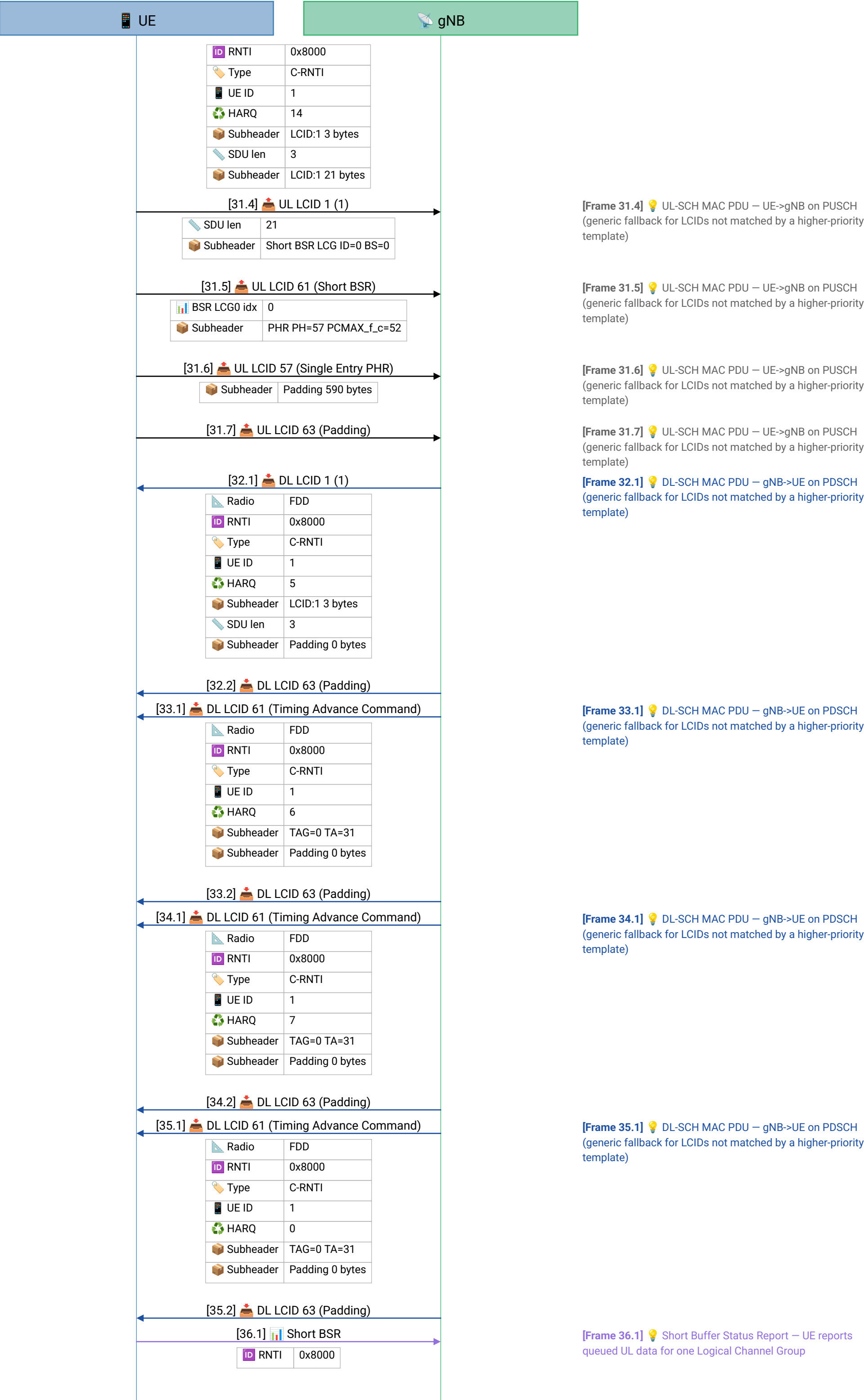
[Frame 21.4] 611 bytes on LCID 1 — the first slice of the capability report, and the largest single logical-channel SDU the uplink has carried so far.

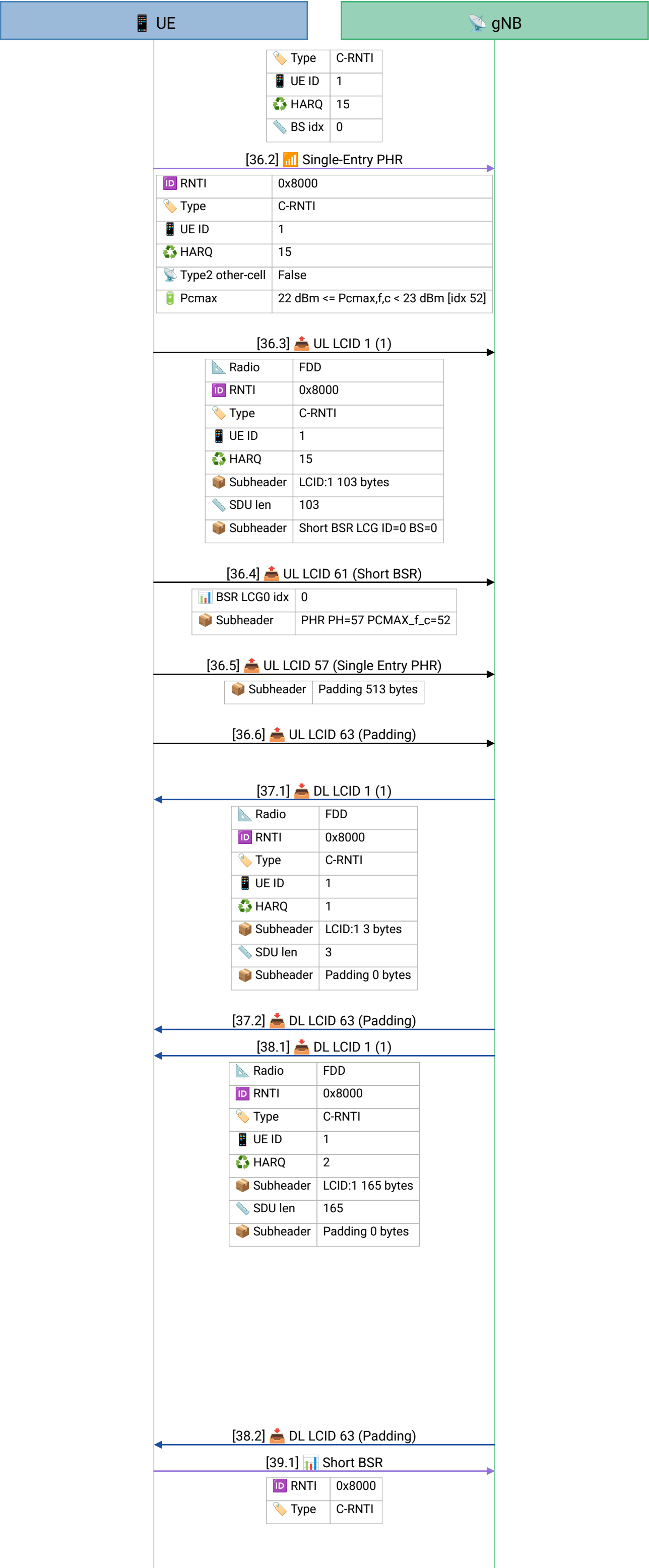
[Frame 21.5] 💡 UL-SCH MAC PDU — UE->gNB on PUSCH (generic fallback for LCIDs not matched by a higher-priority template)











[Frame 36.2] 💡 Power Headroom Report – UE reports transmit power margin remaining below P_{cmx}

[Frame 36.3] 💡 UL-SCH MAC PDU — UE->gNB on PUSCH
(generic fallback for LCIDs not matched by a higher-priority template)

[Frame 36.4] 📡 UL-SCH MAC PDU — UE->gNB on PUSCH
(generic fallback for LCIDs not matched by a higher-priority template)

[Frame 36.5] 💡 UL-SCH MAC PDU — UE->gNB on PUSCH
(generic fallback for LCIDs not matched by a higher-priority template)

[Frame 36.6] 💡 UL-SCH MAC PDU — UE->gNB on PUSCH
(generic fallback for LCIDs not matched by a higher-priority template)

[Frame 37.1] 💡 DL-SCH MAC PDU – gNB->UE on PDSCH
(generic fallback for LCIDs not matched by a higher-priority template)

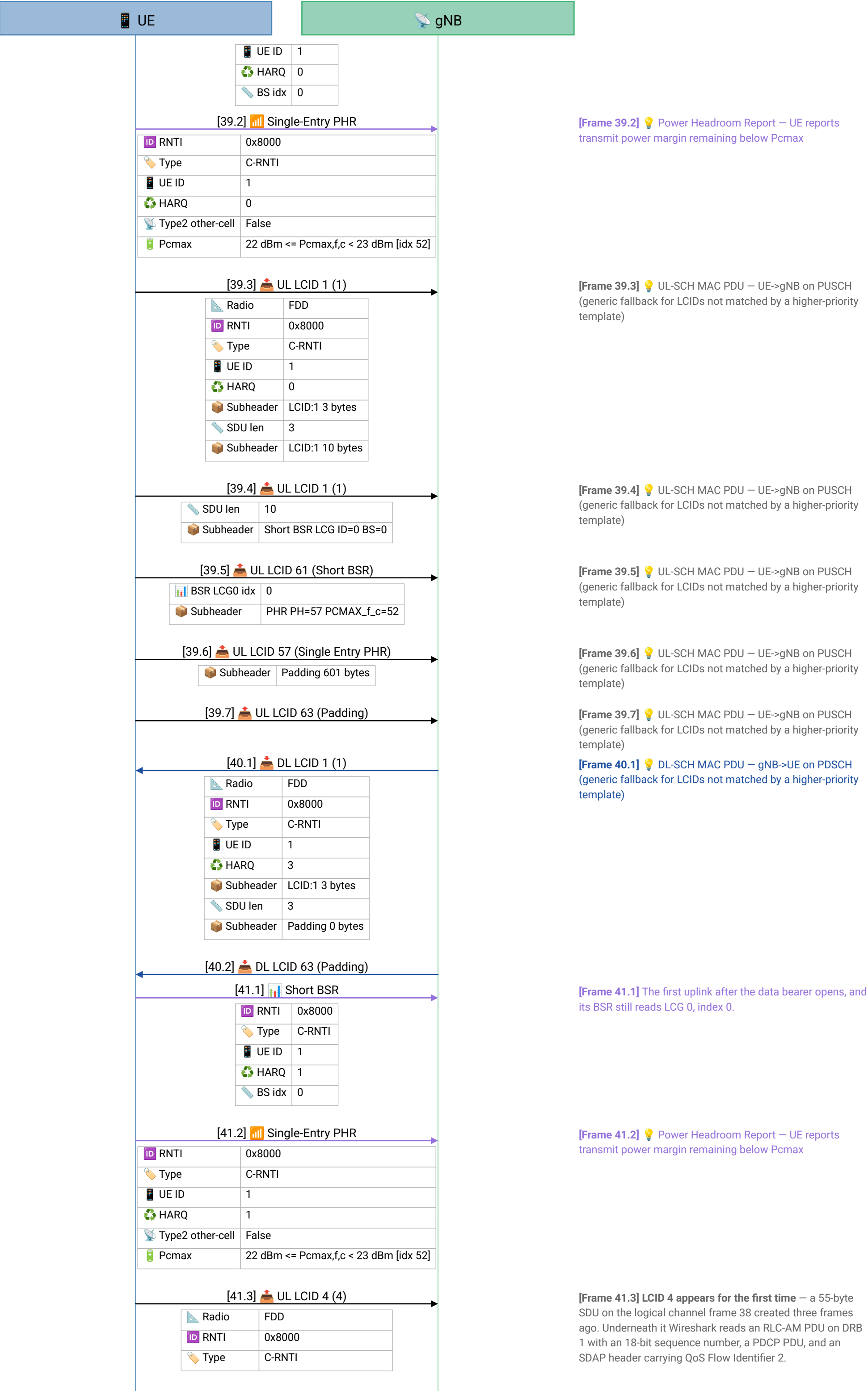
[Frame 38.1] The reconfiguration that mints a new logical channel. A 165-byte SDU on LCID 1, decoding to an RRC Reconfiguration. Two entries in it matter to MAC:

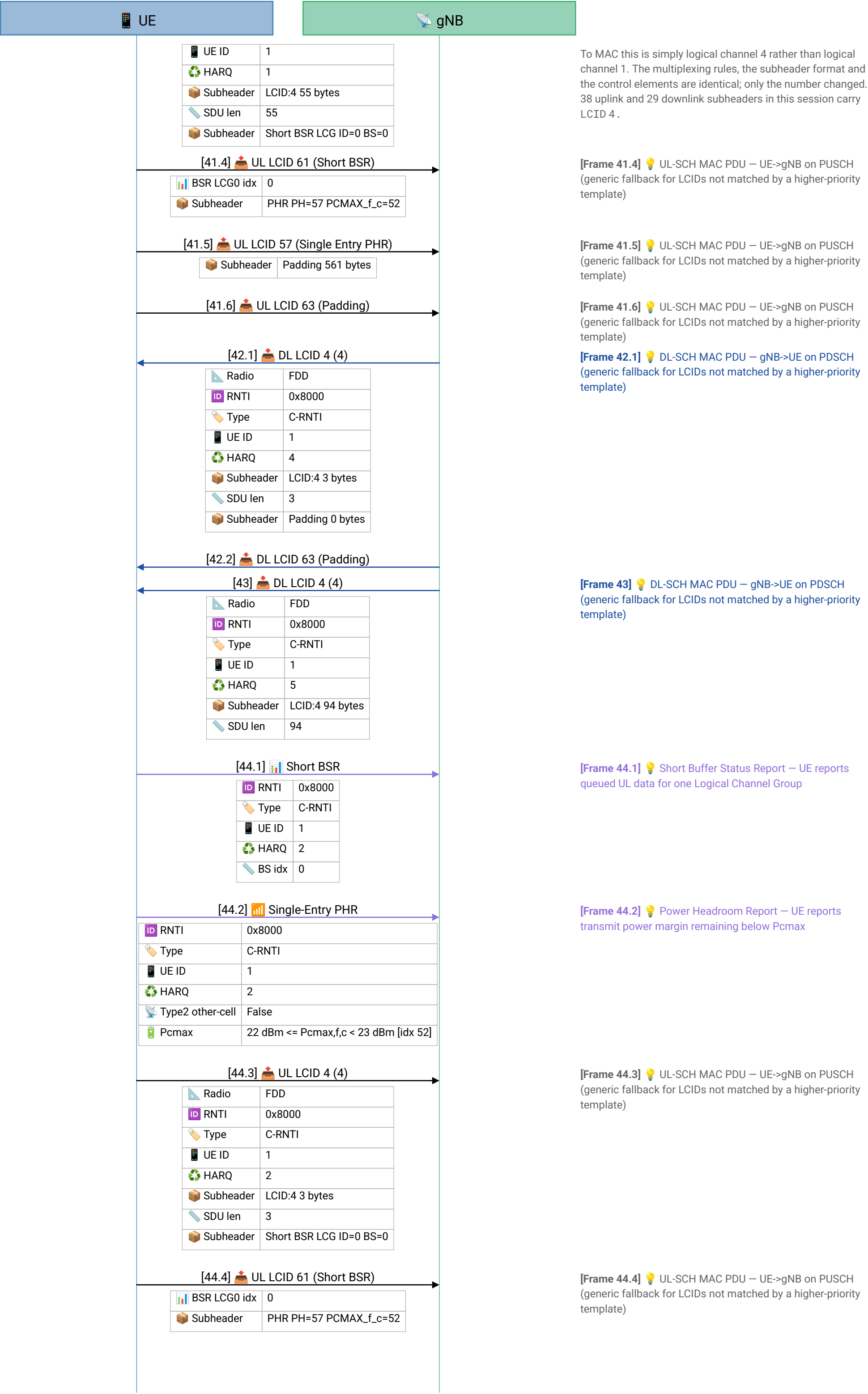
- logicalChannelIdentity 4, servedRadioBearer drb-Identity 1, priority 1, bucketSizeDuration ms50, logicalChannelGroup 1, schedulingRequestID 0
- logicalChannelIdentity 2, priority 3, bucketSizeDuration ms50, logicalChannelGroup 0, schedulingRequestID 0

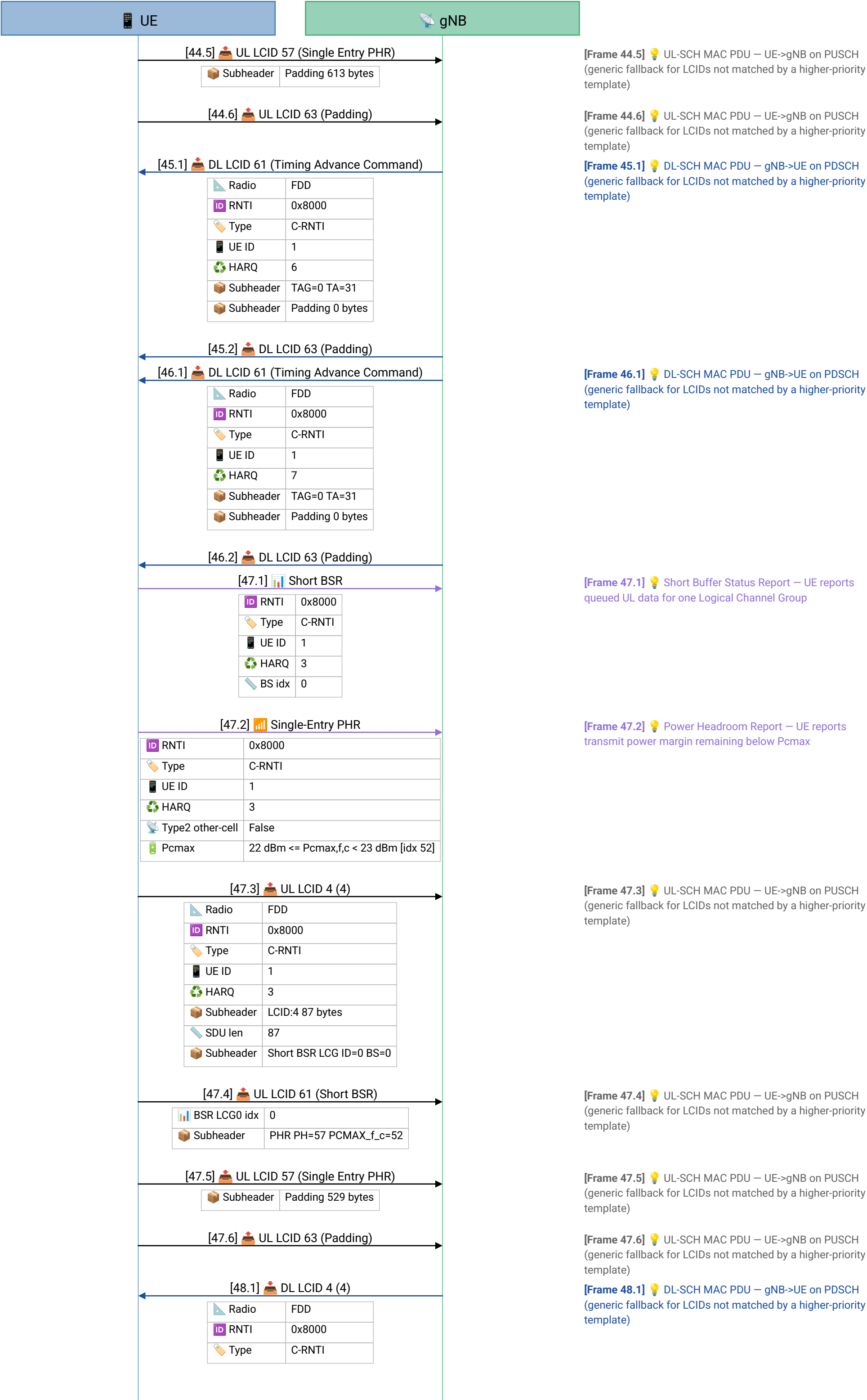
The Logical Channel Group is the field to hold on to. A Buffer Status Report does not report per logical channel – it reports per *group* – and this frame is where the mapping is decided. From here on, a BSR naming LCG 1 is talking about the data bearer and one naming LCG 0 is talking about signaling. Every LCG 1 report in the session appears after this frame.

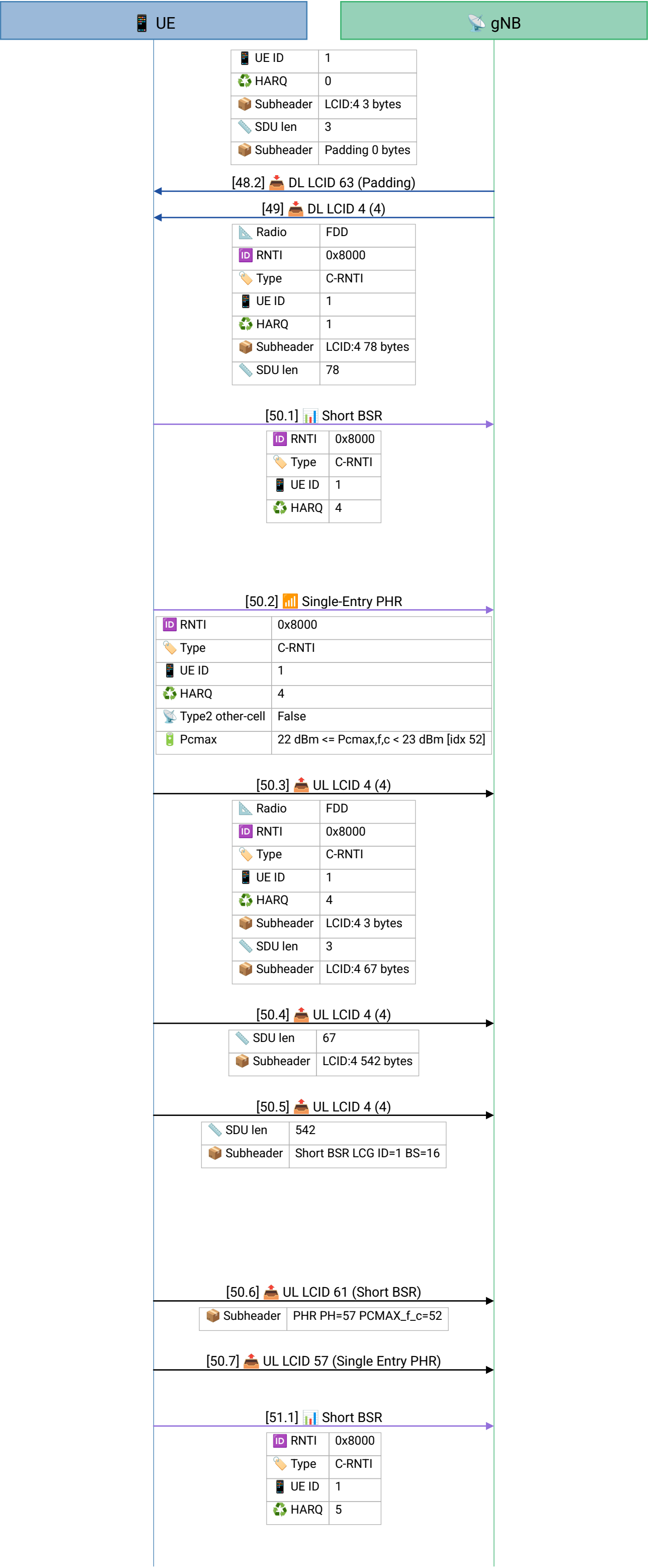
Note also what this message does **not** contain:
mac-CellGroupConfig is absent. MAC itself is never
reconfigured after frame 3.

[Frame 39.1]  Short Buffer Status Report – UE reports queued UL data for one Logical Channel Group









[Frame 49] 💡 DL-SCH MAC PDU – gNB->UE on PDSCH
(generic fallback for LCIDs not matched by a higher-priority template)

[Frame 50.1] A Buffer Status Report on Logical Channel Group 1. Index 16 – 1038 < BS <= 1446 bytes queued – and this time the queue belongs to the data bearer, because frame 38 put LCID 4 in LCG 1.

That distinction is what makes grouping useful. The gNB now knows not just *how much* the UE has to send but *what kind*: signaling backlog and user-plane backlog arrive as separately numbered reports, and a scheduler that wants to protect latency-sensitive signaling can act on them differently.

[Frame 50.2] 💡 Power Headroom Report – UE reports transmit power margin remaining below P_{cmx}

[Frame 50.3] Three SDUs and two control elements in one transport block – the busiest PDU shape in the session.

This is the first of the three, a three-byte RLC status report on LCID 4. Only 13 of the 144 frames reach five subheaders, and this is one of them. There is no padding subheader at all: three SDUs plus the BSR plus the PHR filled the grant exactly.

[Frame 50.4] 67 bytes on LCID 4, the second of three.

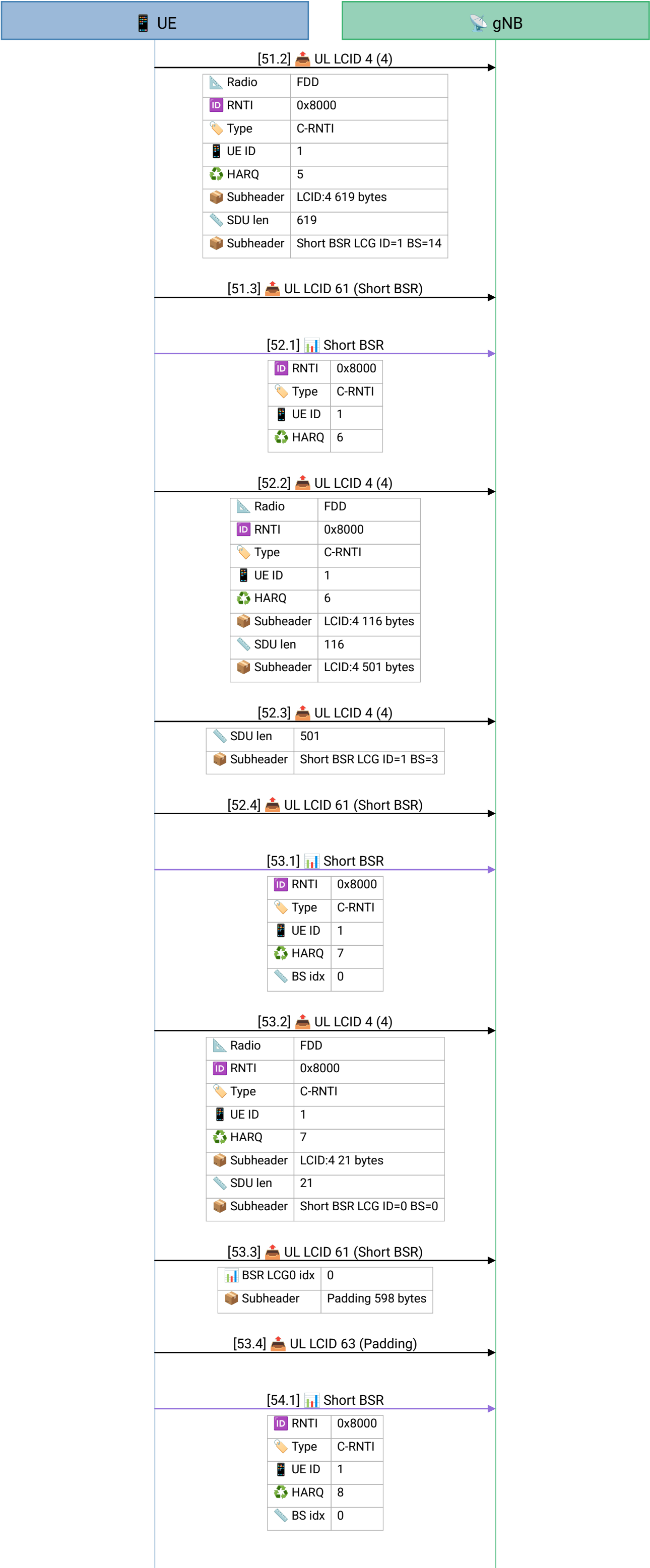
[Frame 50.5] 542 bytes on LCID 4, the third — and the reason the BSR alongside still had something to report. RLC had more queued than the grant could take, so it segmented, and MAC packed what fitted.

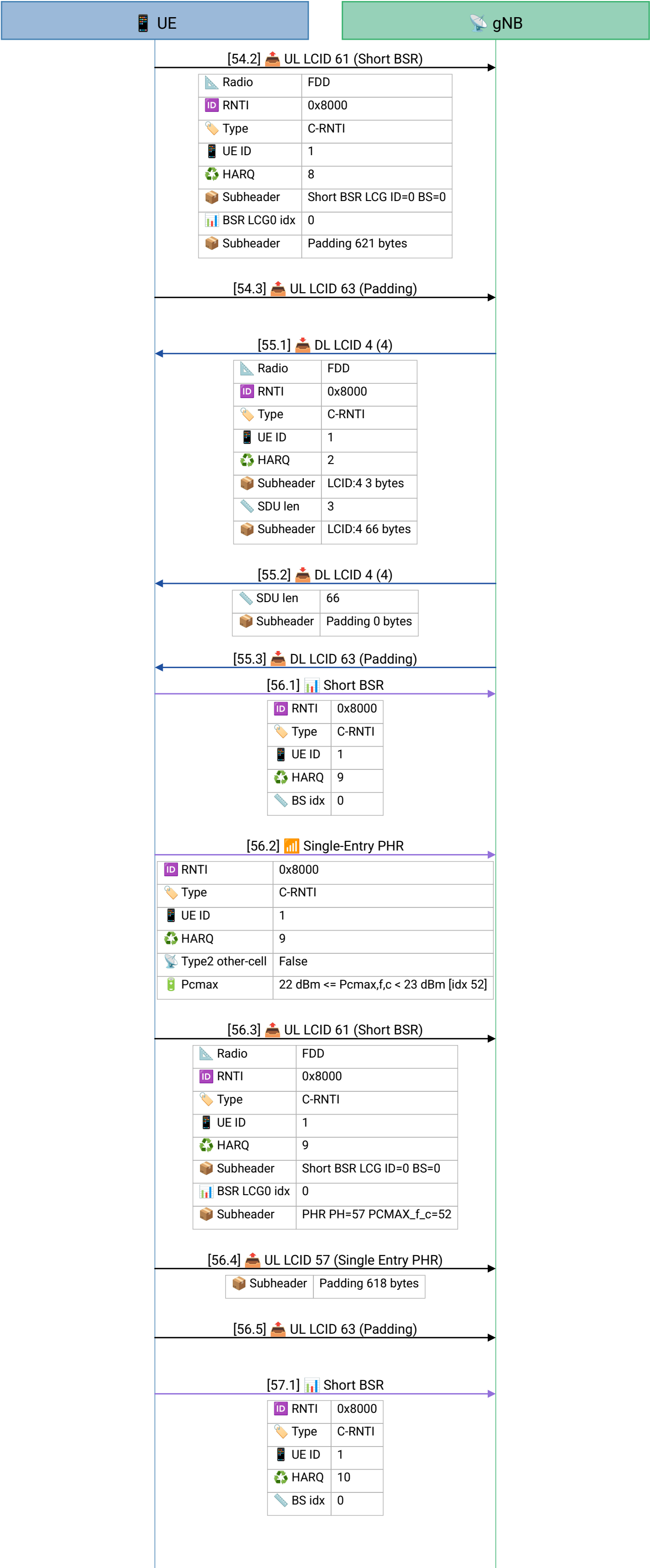
Read the five subheaders in order and the assembly rule is plain: SDU, SDU, SDU, Short BSR, PHR. Uplink control elements come after every SDU (TS 38.321 §6.1.2), and the BSR is deliberately the second-to-last thing built, so it can describe the buffer *after* the scheduler's grant has been consumed.

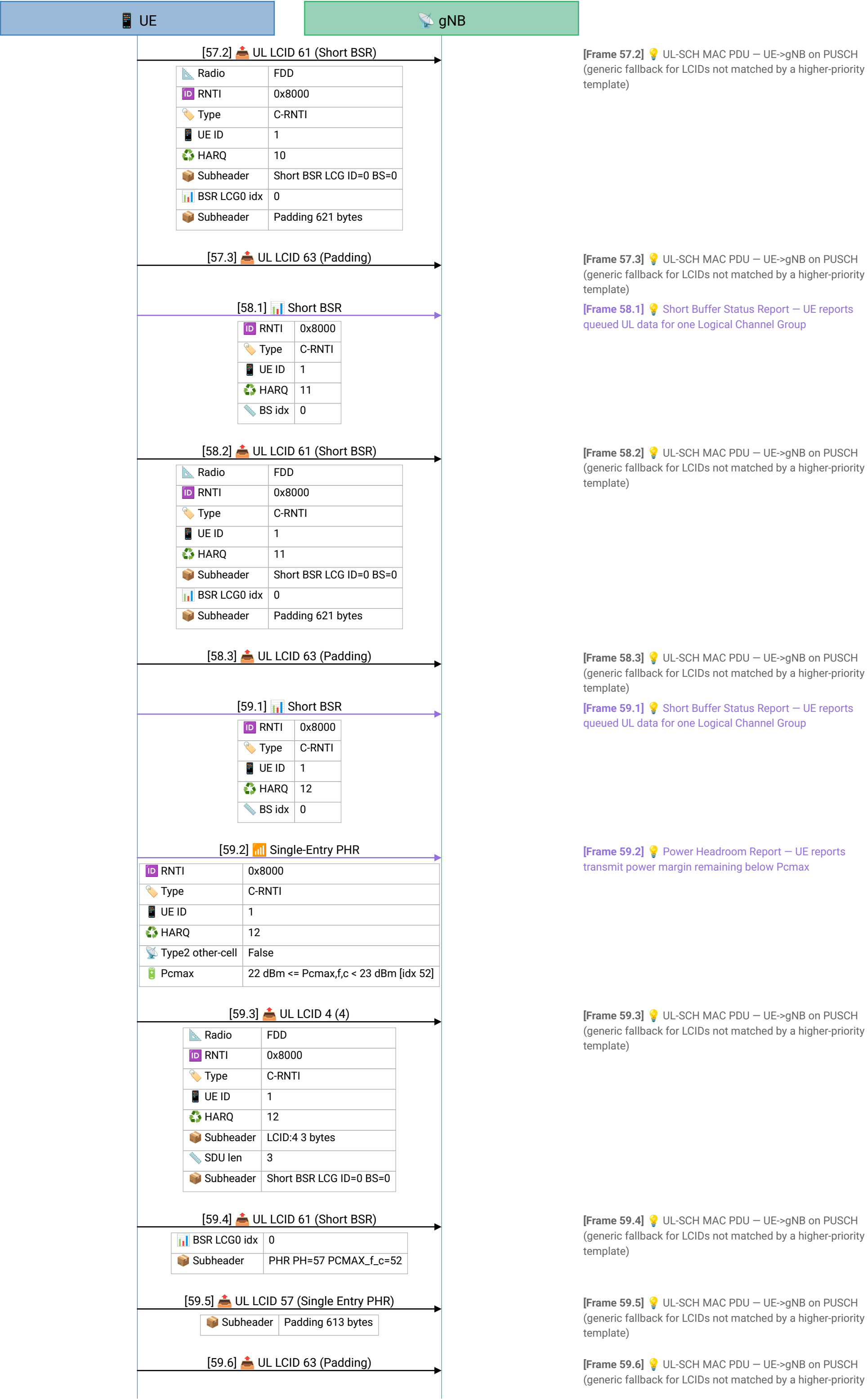
[Frame 50.6] 💡 UL-SCH MAC PDU – UE->gNB on PUSCH
(generic fallback for LCIDs not matched by a higher-priority template)

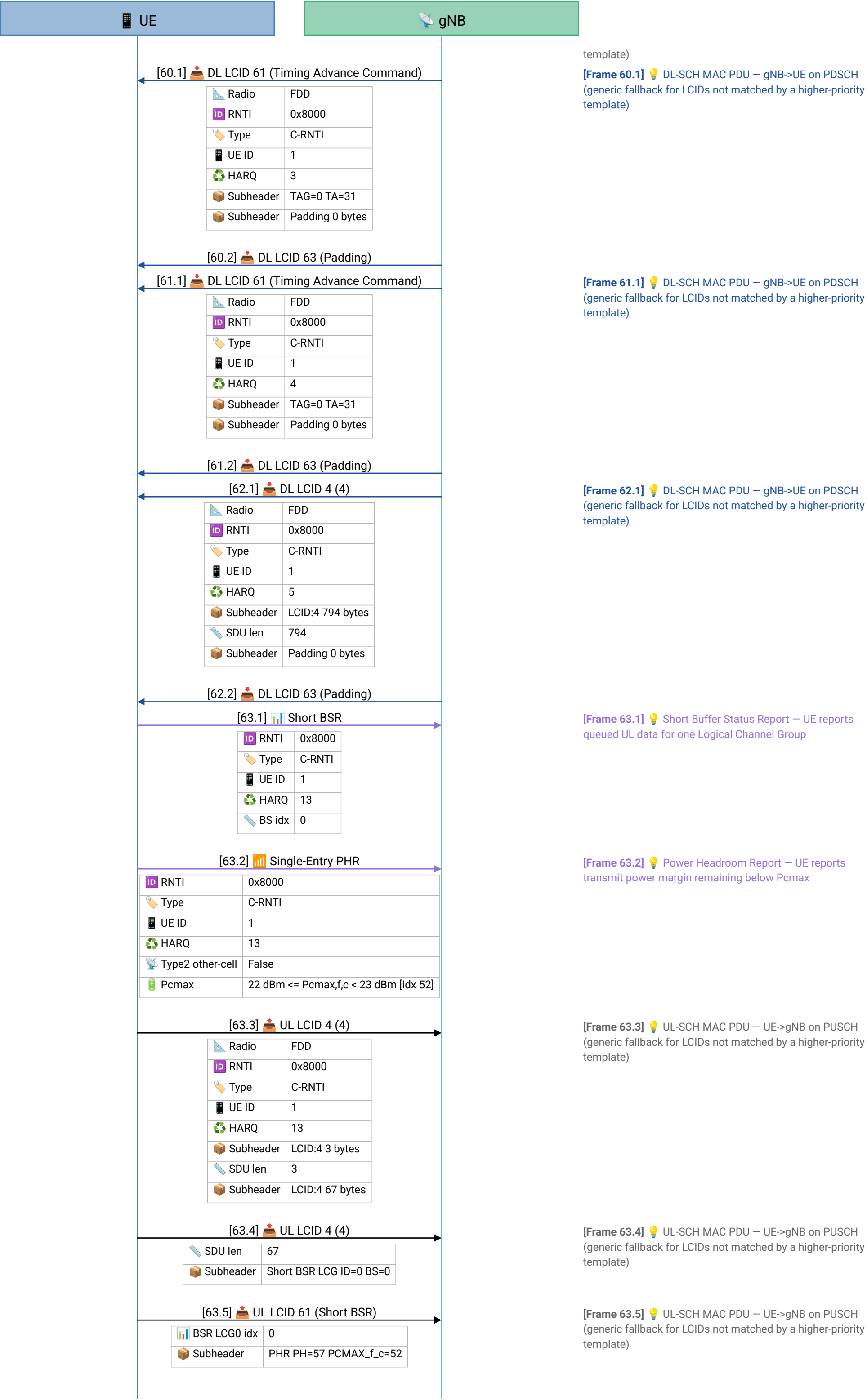
[Frame 50.7] 💡 UL-SCH MAC PDU – UE->gNB on PUSCH
(generic fallback for LCIDs not matched by a higher-priority template)

[Frame 51.1] 💡 Short Buffer Status Report – UE reports queued UL data for one Logical Channel Group

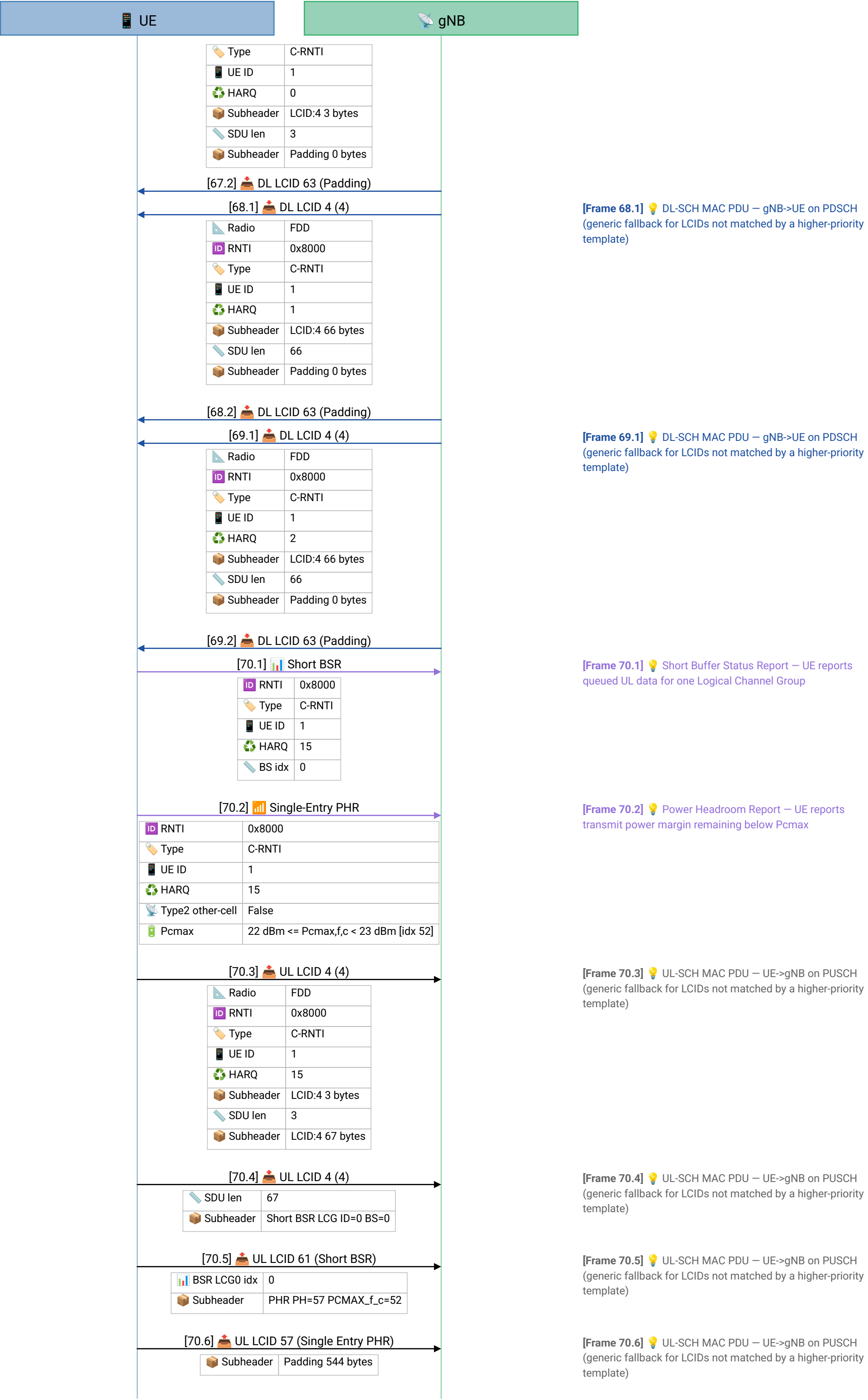


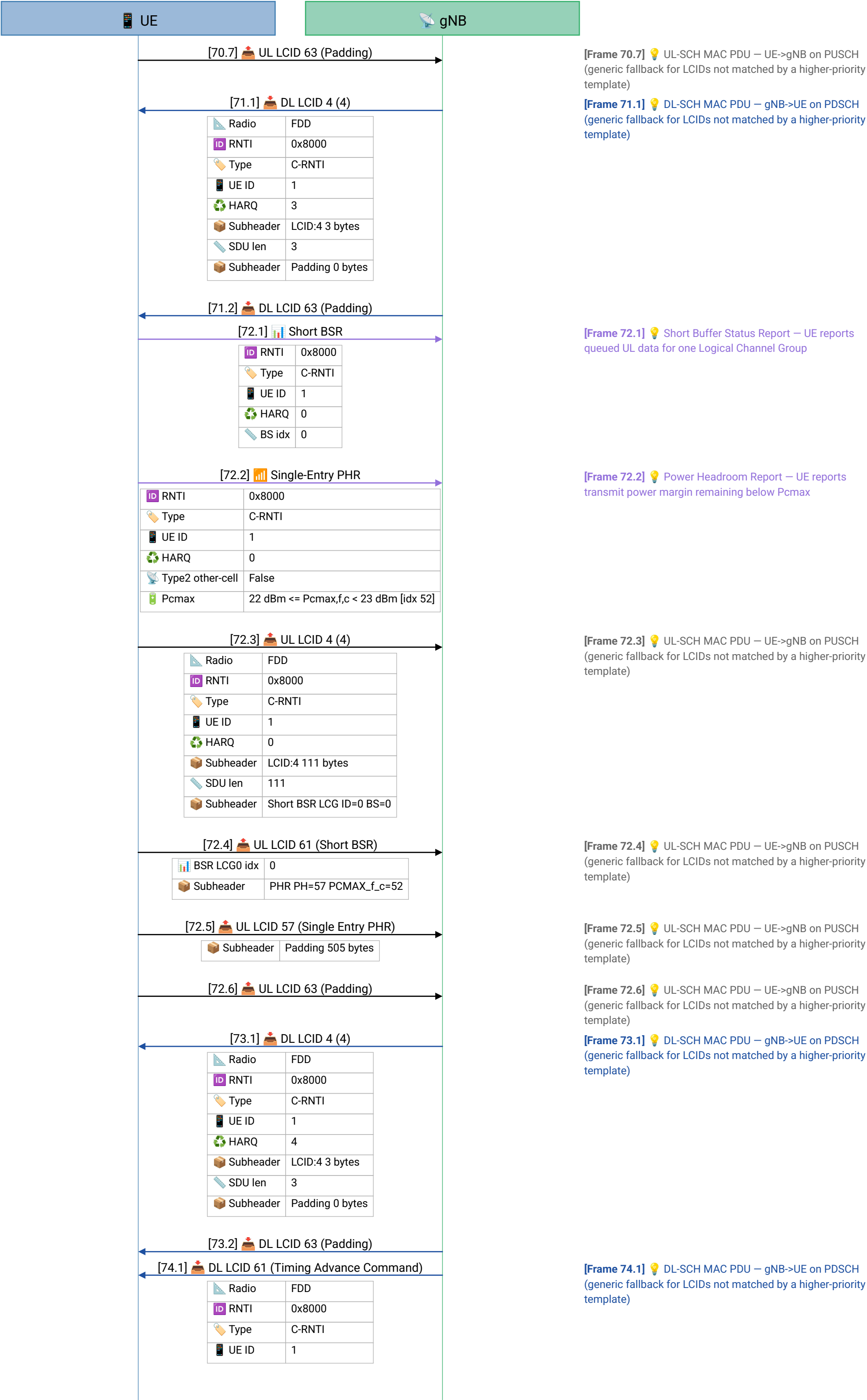


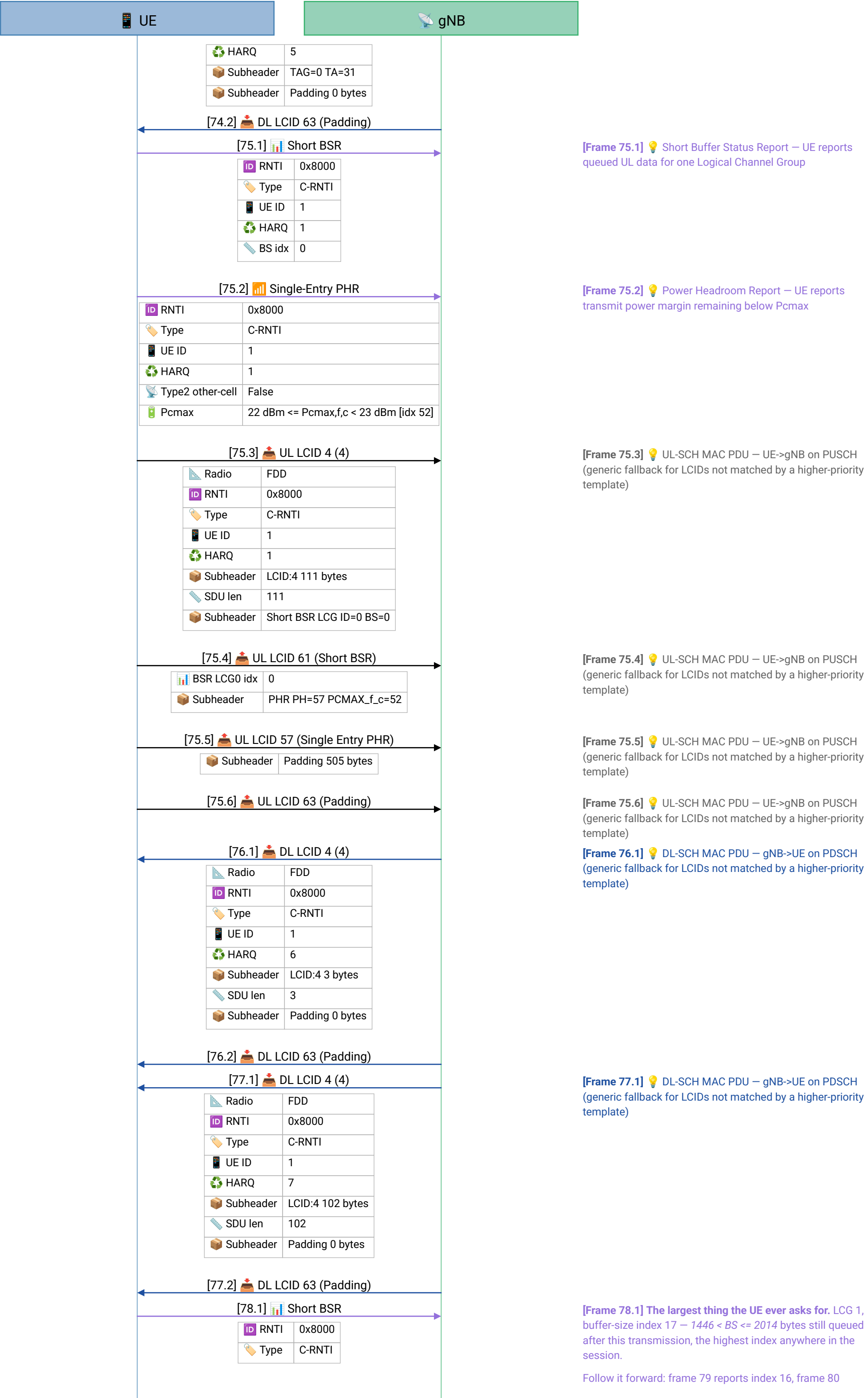


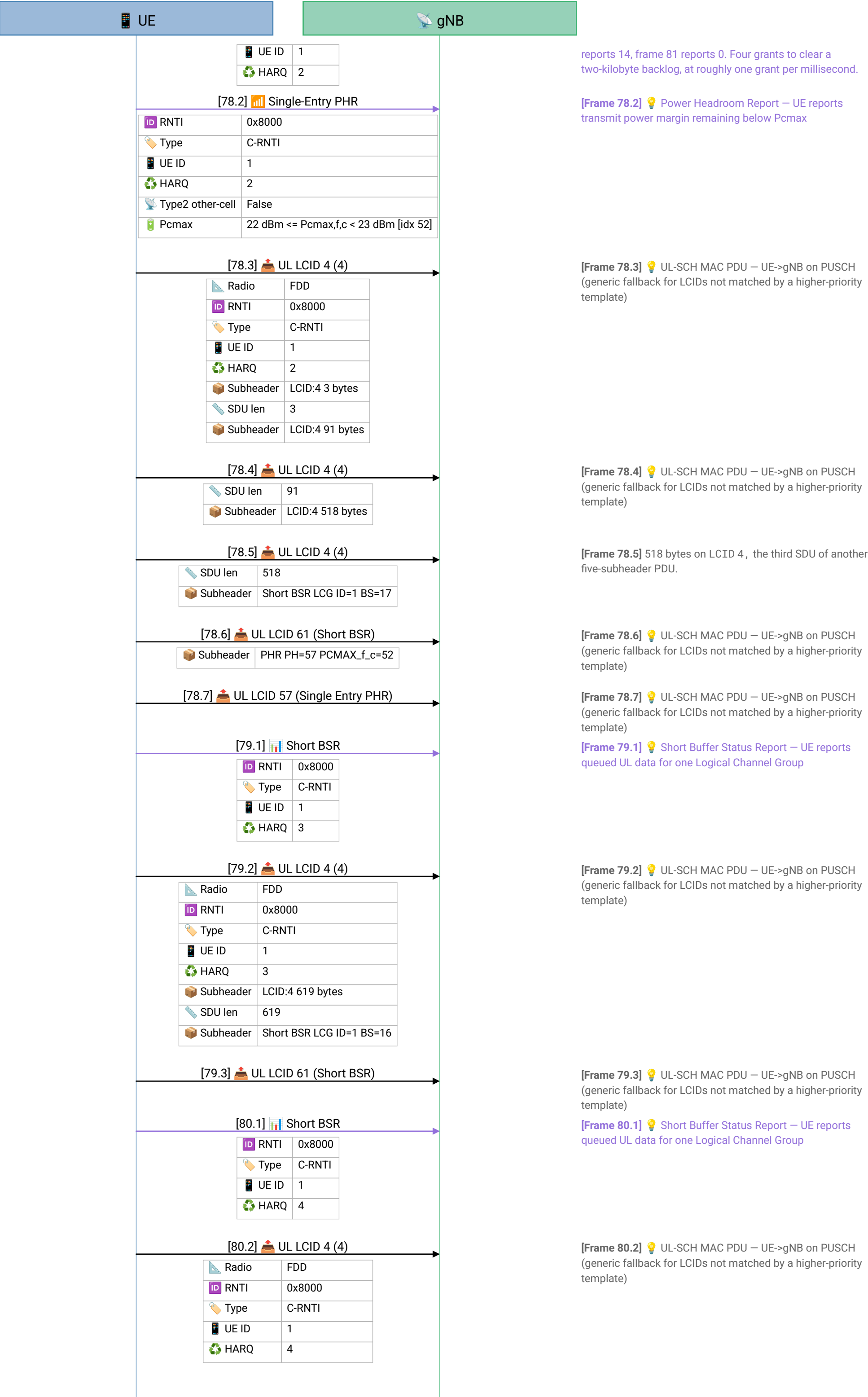


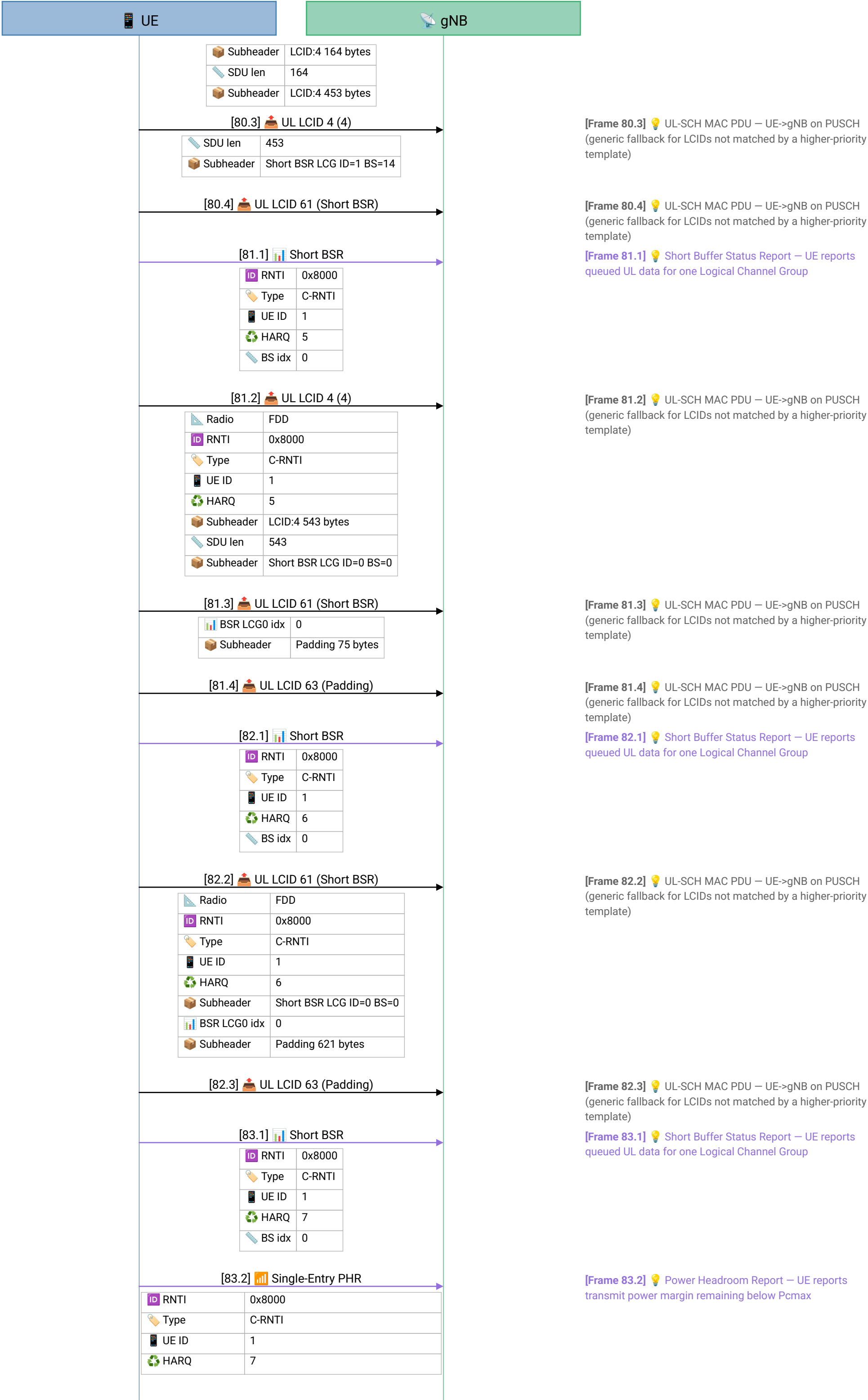


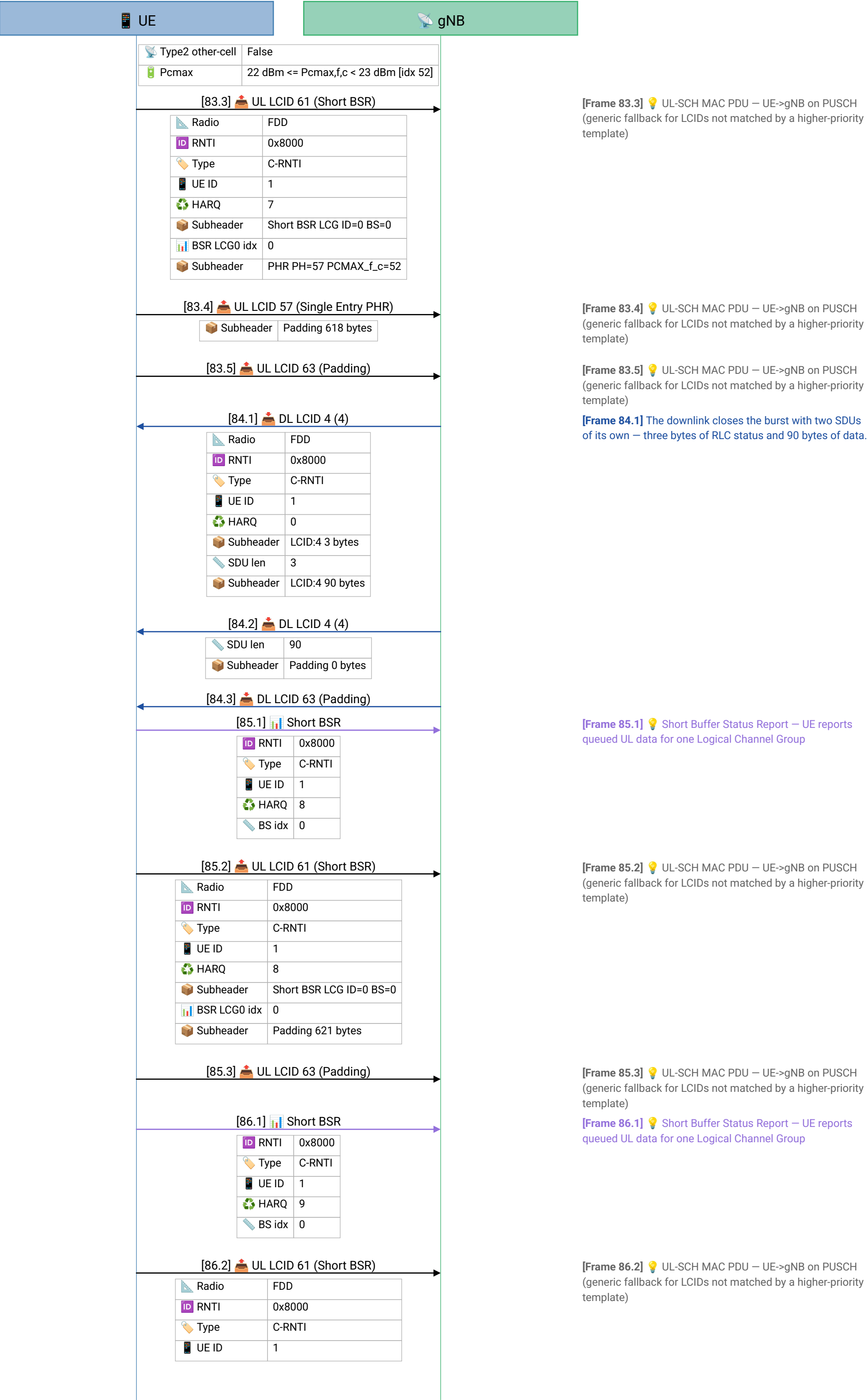


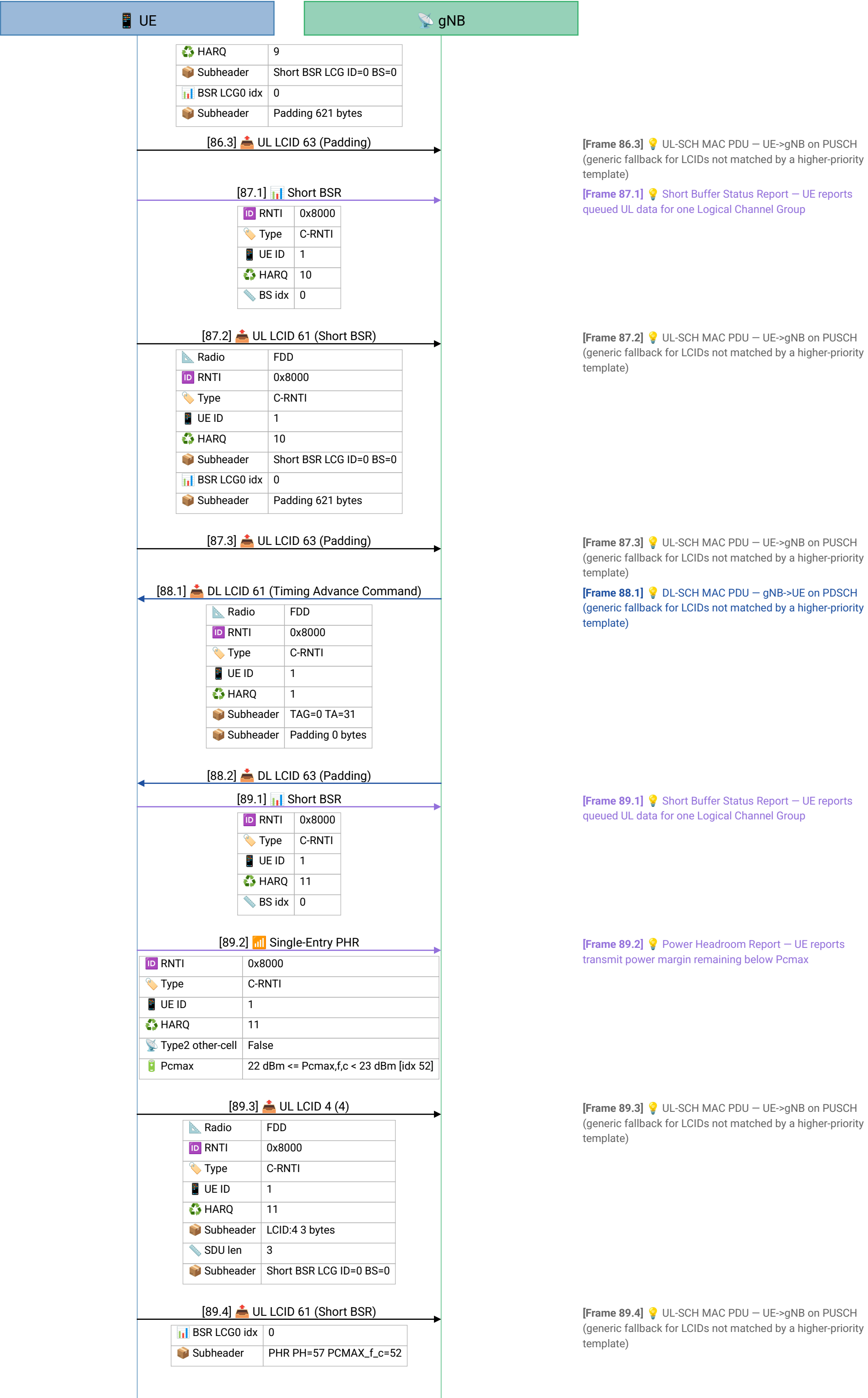


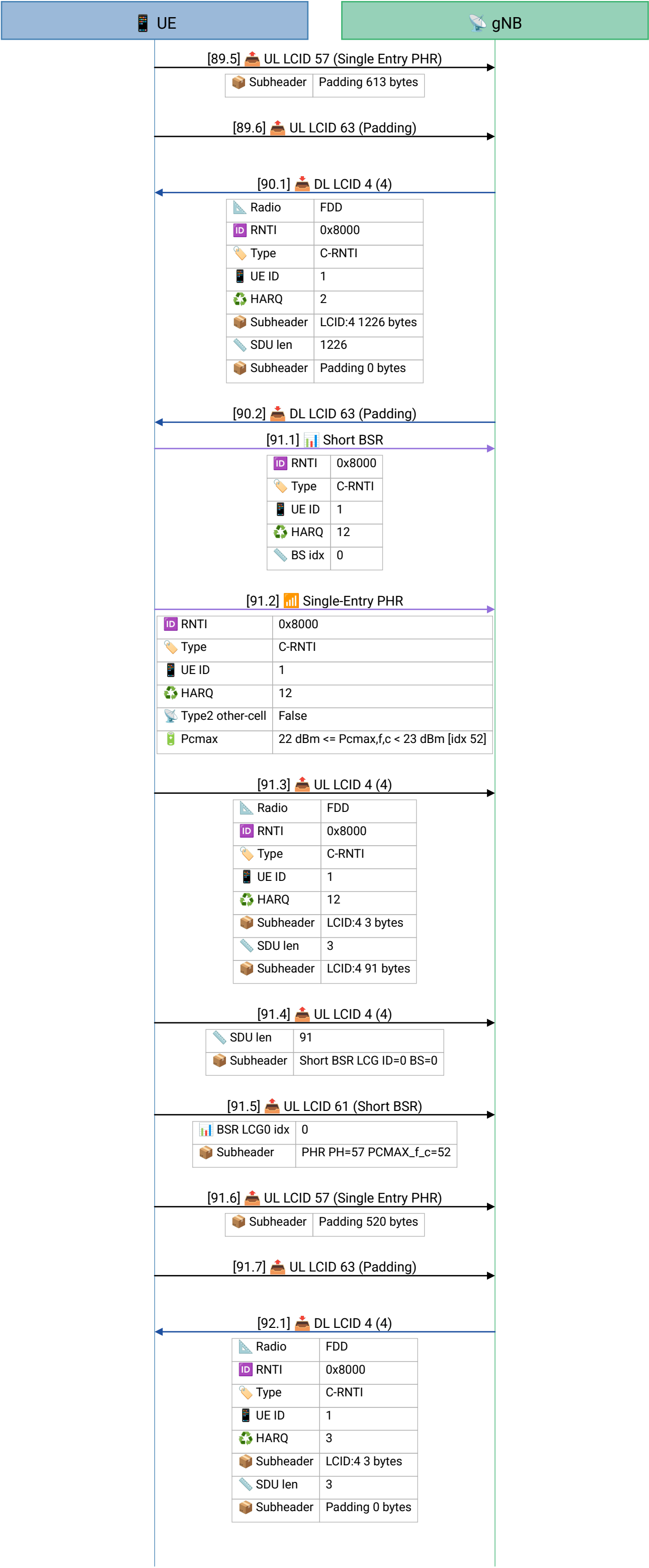












[Frame 89.5] 💡 UL-SCH MAC PDU — UE->gNB on PUSCH (generic fallback for LCIDs not matched by a higher-priority template)

[Frame 89.6] 💡 UL-SCH MAC PDU — UE->gNB on PUSCH (generic fallback for LCIDs not matched by a higher-priority template)

[Frame 90.1] A 1226-byte downlink SDU, carried whole. One subheader, one logical channel, no control elements, and a zero-byte padding subheader behind it.

Compare the uplink's ceiling of 619 bytes. The difference is not a protocol limit. The gNB sizes its own transport blocks against a buffer it can see directly, while the uplink runs on a grant fixed before the UE's data existed.

[Frame 91.1] 💡 Short Buffer Status Report — UE reports queued UL data for one Logical Channel Group

[Frame 91.2] 💡 Power Headroom Report — UE reports transmit power margin remaining below Pcmax

[Frame 91.3] 💡 UL-SCH MAC PDU — UE->gNB on PUSCH (generic fallback for LCIDs not matched by a higher-priority template)

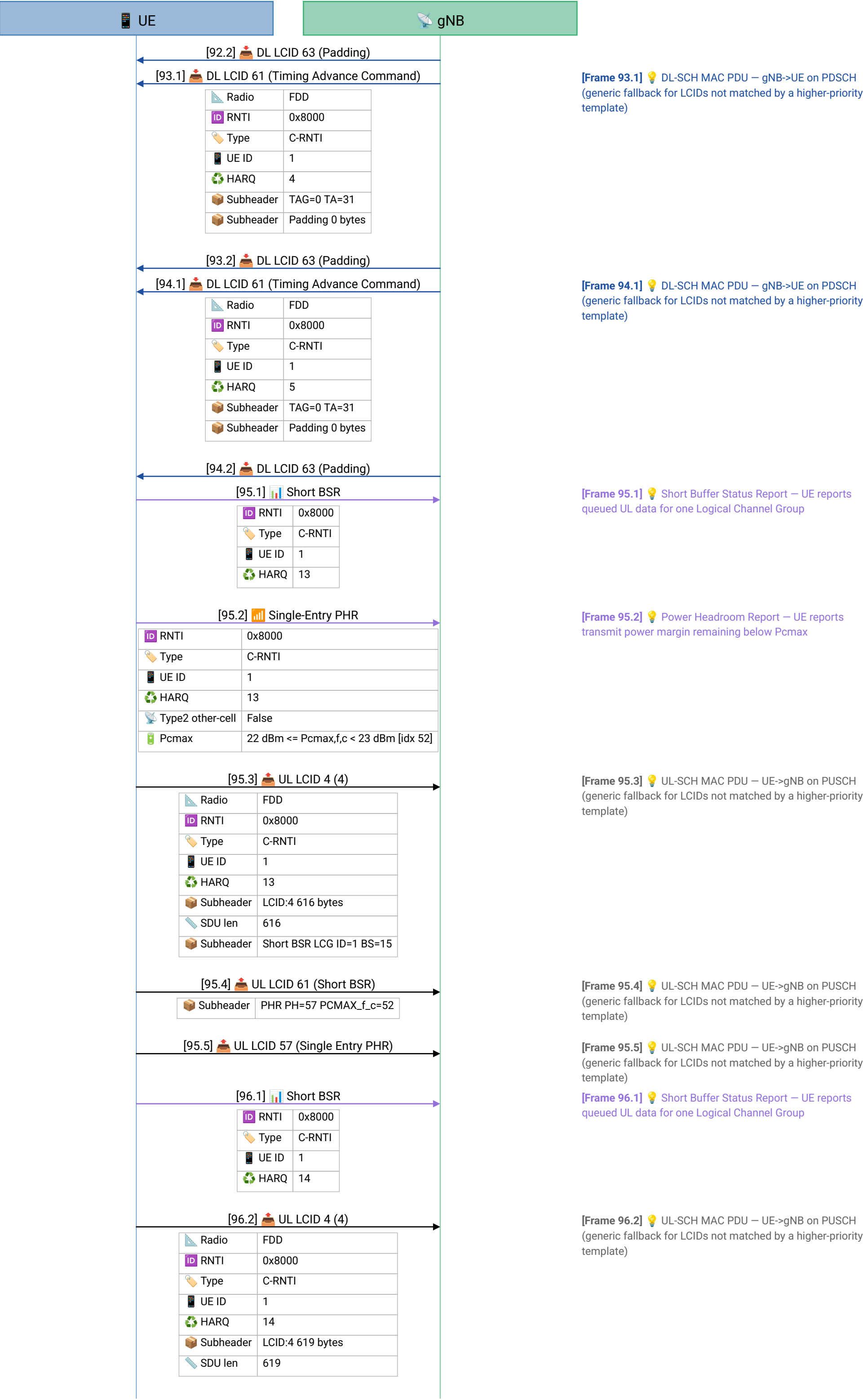
[Frame 91.4] 💡 UL-SCH MAC PDU — UE->gNB on PUSCH (generic fallback for LCIDs not matched by a higher-priority template)

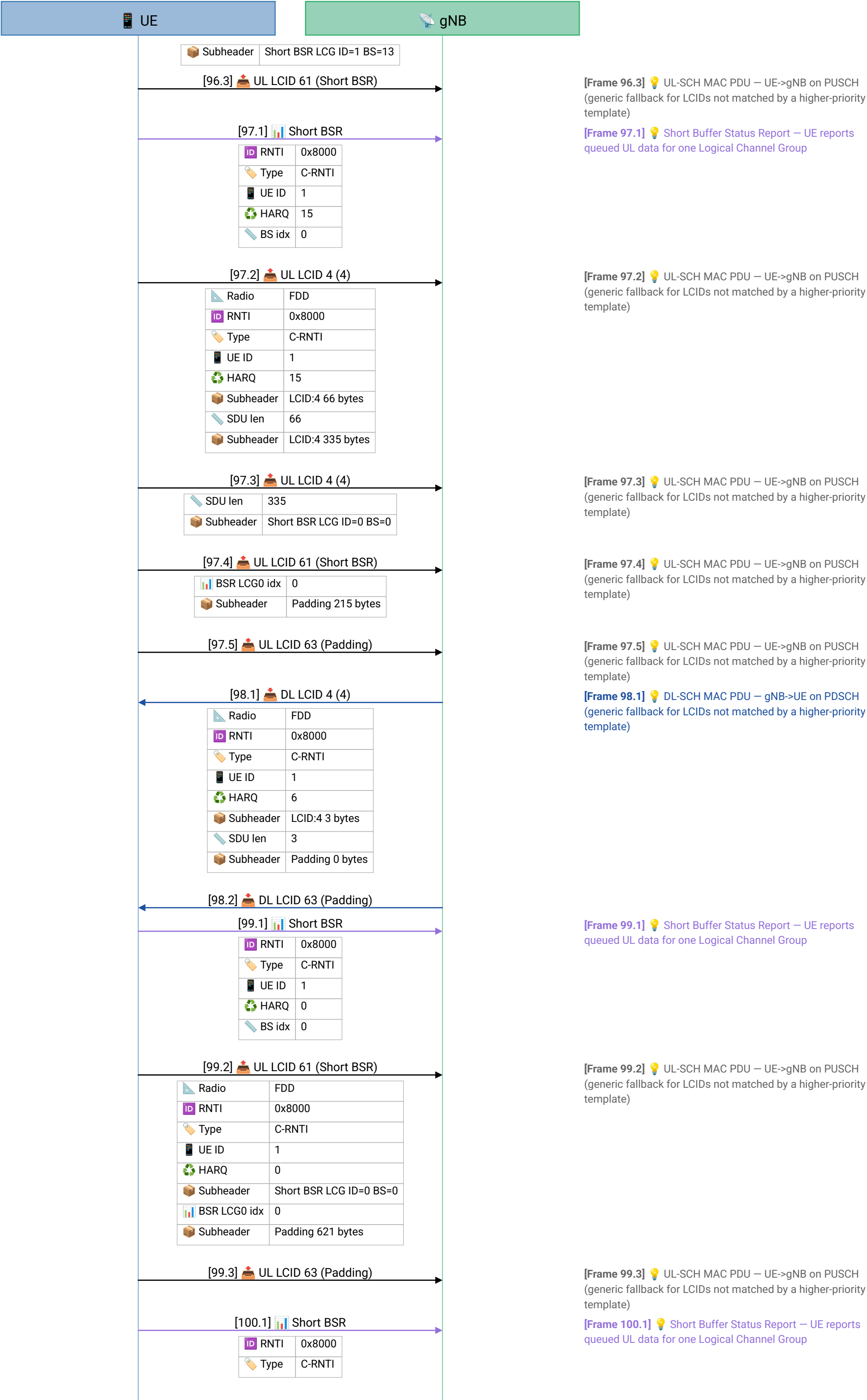
[Frame 91.5] 💡 UL-SCH MAC PDU — UE->gNB on PUSCH (generic fallback for LCIDs not matched by a higher-priority template)

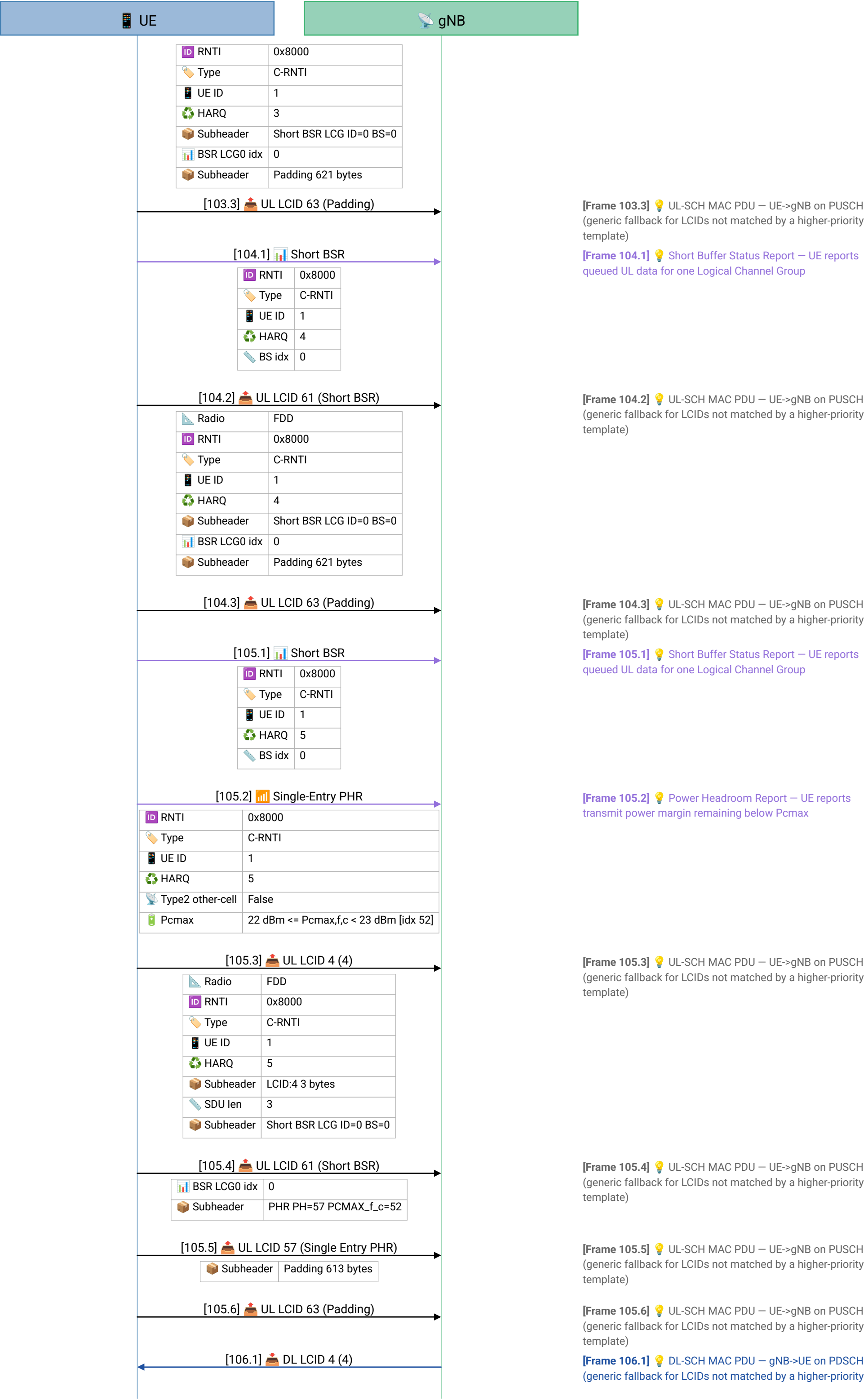
[Frame 91.6] 💡 UL-SCH MAC PDU — UE->gNB on PUSCH (generic fallback for LCIDs not matched by a higher-priority template)

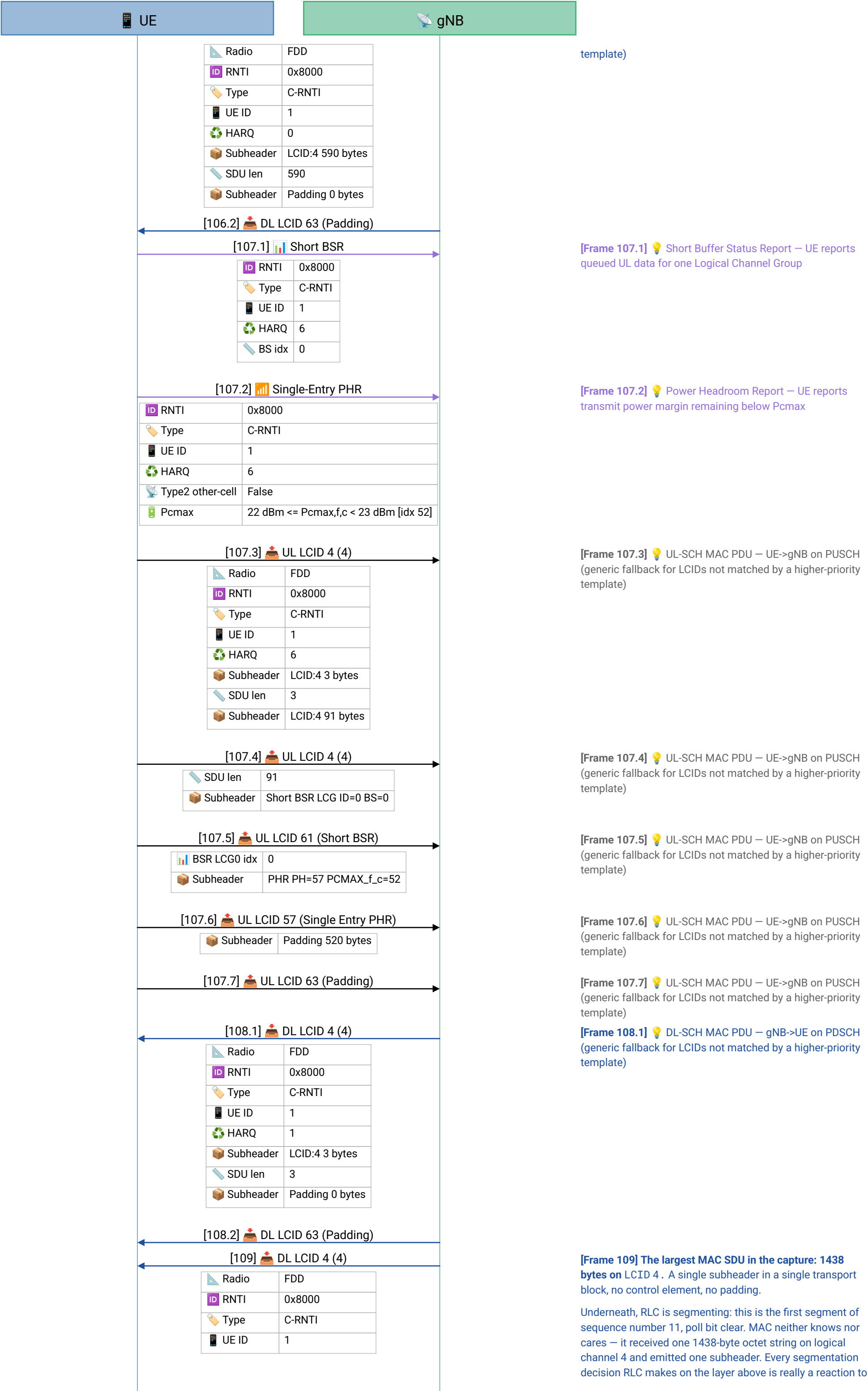
[Frame 91.7] 💡 UL-SCH MAC PDU — UE->gNB on PUSCH (generic fallback for LCIDs not matched by a higher-priority template)

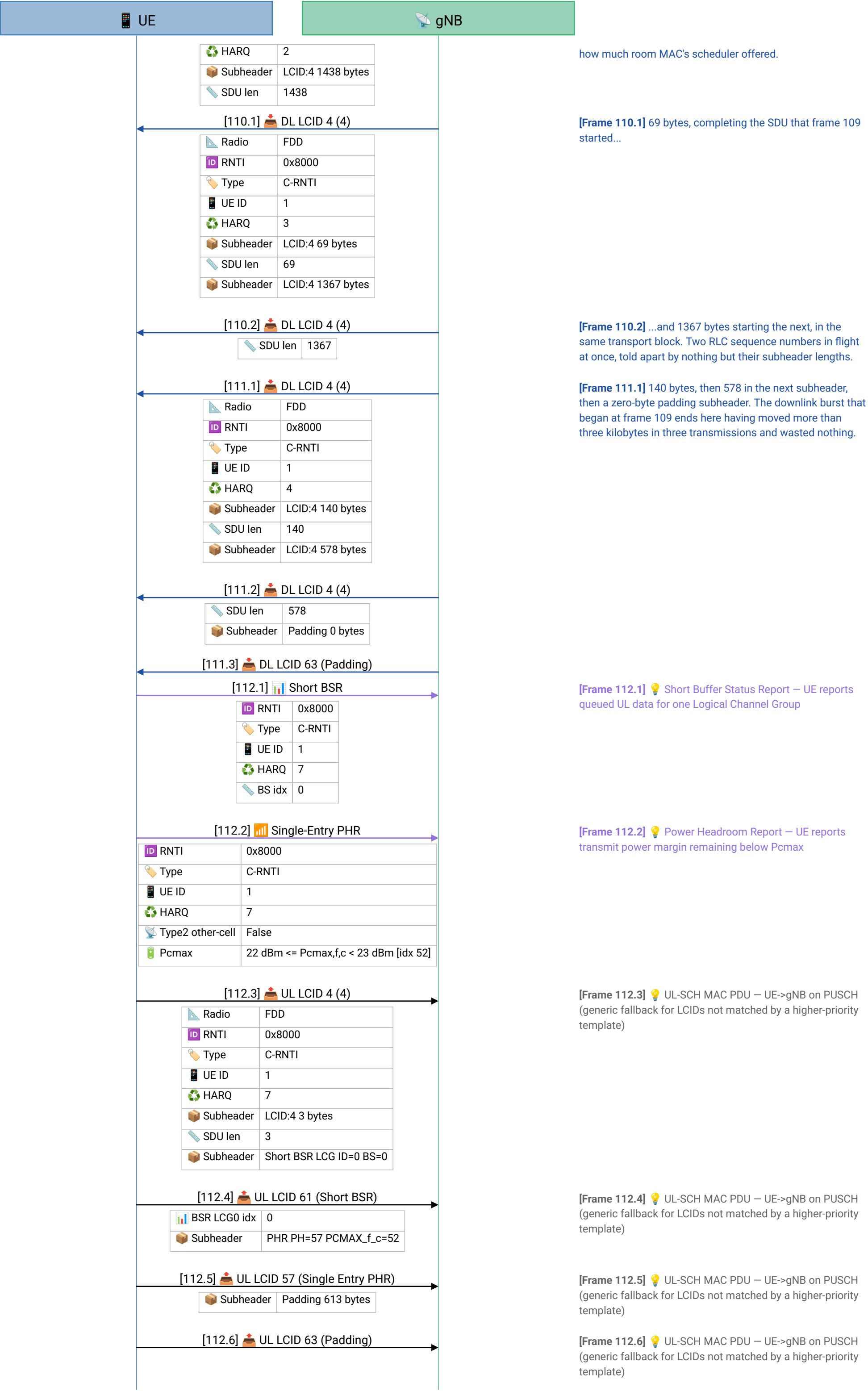
[Frame 92.1] 💡 DL-SCH MAC PDU — gNB->UE on PDSCH (generic fallback for LCIDs not matched by a higher-priority template)





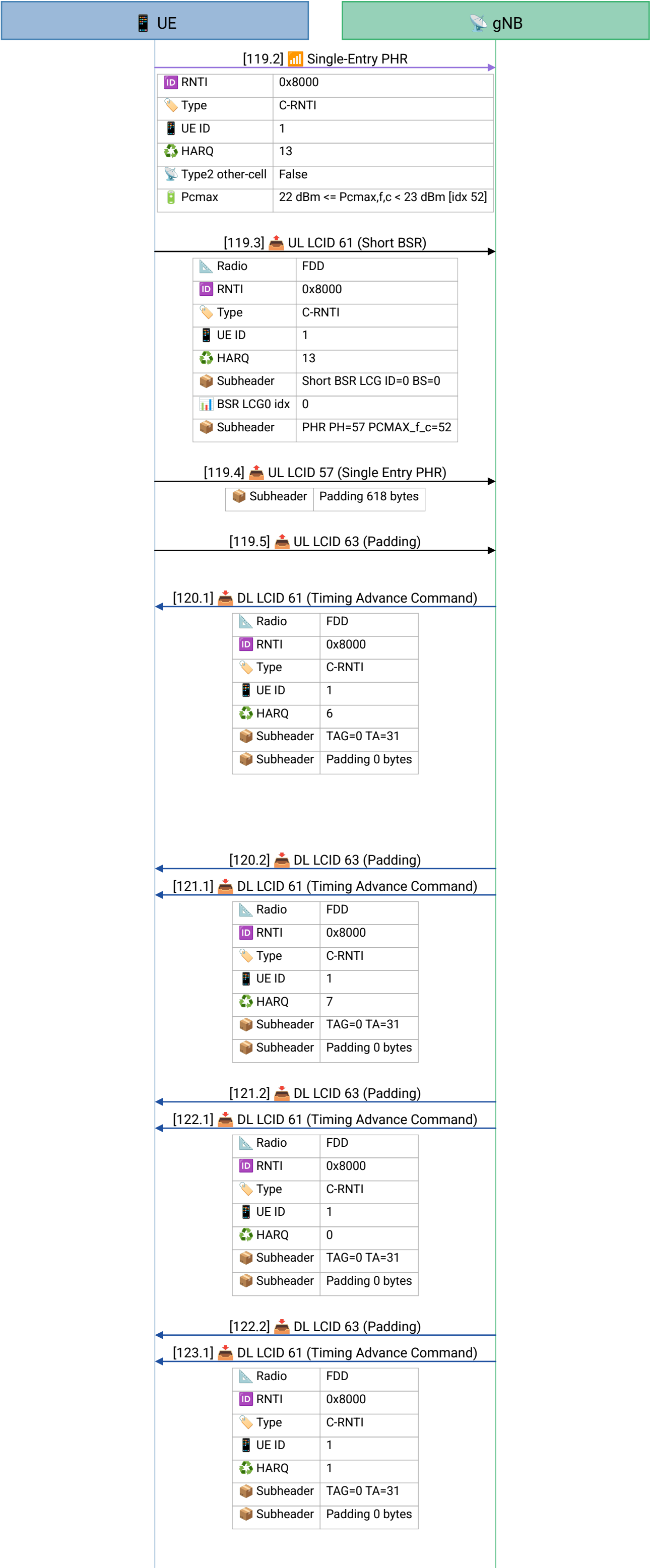


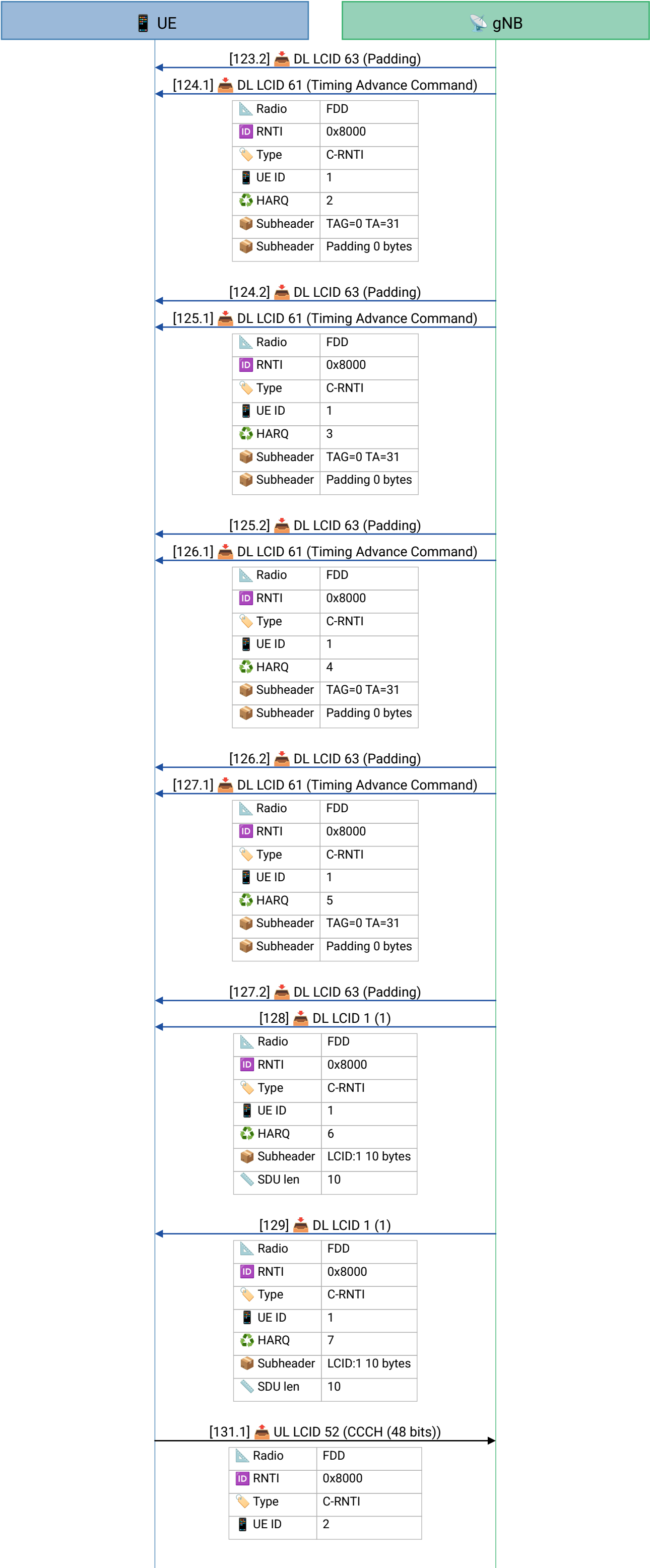


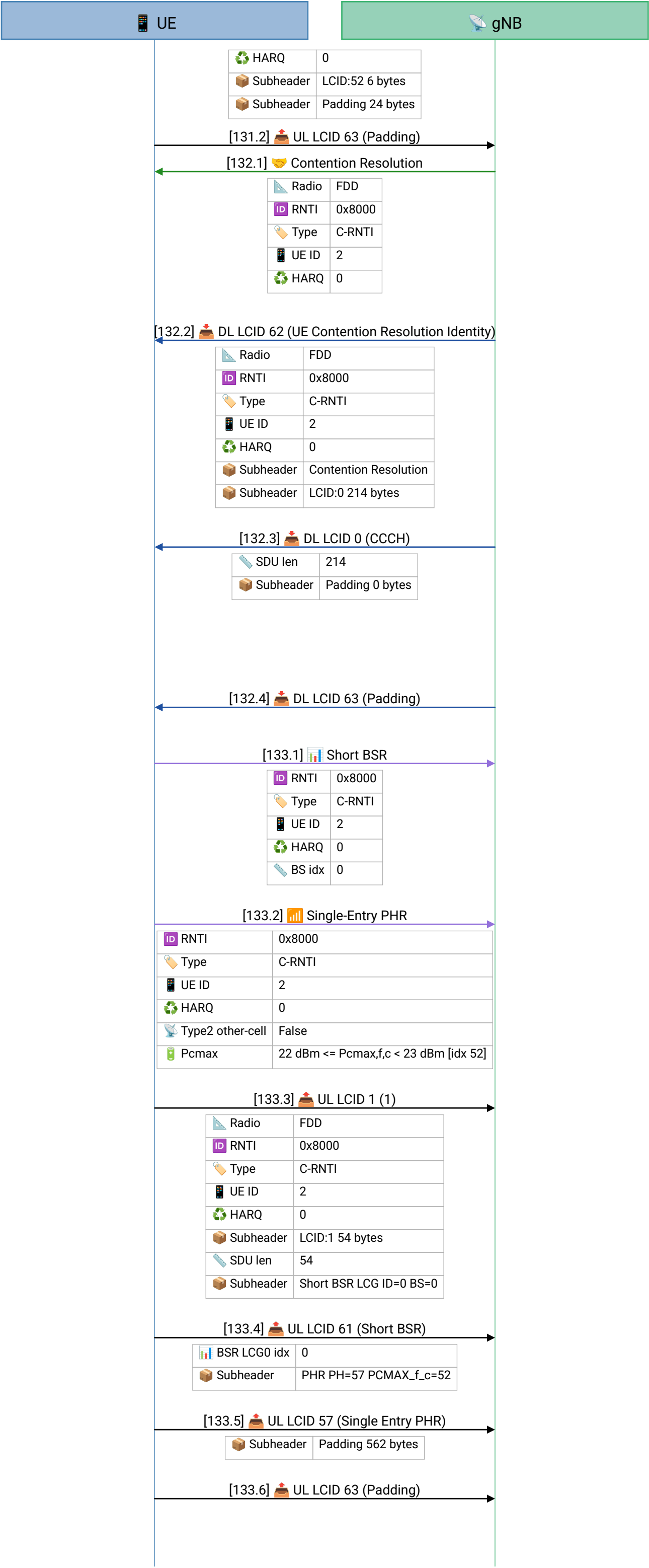












tells them apart: UEId 1 for frames 2 to 129, UEId 2 from here on.

[Frame 132.1] Contention resolution again – and the identity is different. The echoed UE Contention Resolution Identity is 0838468acff04, not the 1b34056b1846 of frame 3.

That difference is the proof this is a genuinely new random access rather than a continuation. The identity is whatever Msg3 the UE actually sent this time, and a new random value produces a new echo. Everything frames 2 and 3 taught applies unchanged, one connection later.

[Frame 132.2]  DL-SCH MAC PDU – gNB->UE on PDSCH (generic fallback for LCIDs not matched by a higher-priority template)

[Frame 132.3] A 214-byte CCCH SDU — the same size as frame 3's, because it is the same RRCSetup carrying the same mac-CellGroupConfig: periodicBSR-Timer sf1, retxBSR-Timer sf80, phr-PeriodicTimer sf10, phr-ProhibitTimer sf0, phr-Tx-PowerFactorChange dB1, timeAlignmentTimer infinity, and no drx-Config. The second connection therefore behaves exactly like the first, and for a readable reason rather than a guessed one.

[Frame 132.4] 💡 DL-SCH MAC PDU — gNB->UE on PDSCH (generic fallback for LCIDs not matched by a higher-priority template)

[Frame 133.1] The new connection's first uplink, and the four-subheader shape of frame 4 returns intact: a 54-byte LCID 1 SDU, a Short BSR reading zero, a Single-Entry PHR reading the same index 57 and index 52 as all 36 before it, and 562 bytes of padding.

[Frame 133.2] 💡 Power Headroom Report – UE reports transmit power margin remaining below P_{cmx}

[Frame 133.3] 📡 UL-SCH MAC PDU — UE->gNB on PUSCH
(generic fallback for LCIDs not matched by a higher-priority template)

[Frame 133.4] 💡 UL-SCH MAC PDU – UE->gNB on PUSCH
(generic fallback for LCIDs not matched by a higher-priority template)

[Frame 133.5] 💡 UL-SCH MAC PDU – UE->gNB on PUSCH
(generic fallback for LCIDs not matched by a higher-priority template)

[Frame 133.6] 💡 UL-SCH MAC PDU – UE->gNB on PUSCH
(generic fallback for LCIDs not matched by a higher-priority template)

