Overview of F1 and E1 and F1 handover

Webinar Chapter 12

Robert Schmidt (OSA)

December 4, 2024

Outline

Motivation

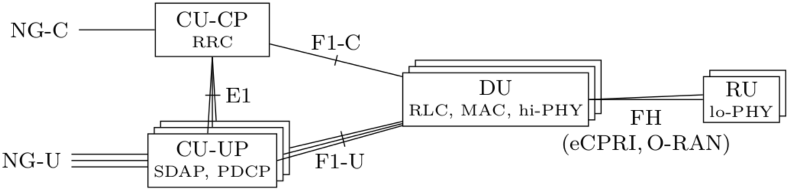
F1+E1 message examples

Implementation Status

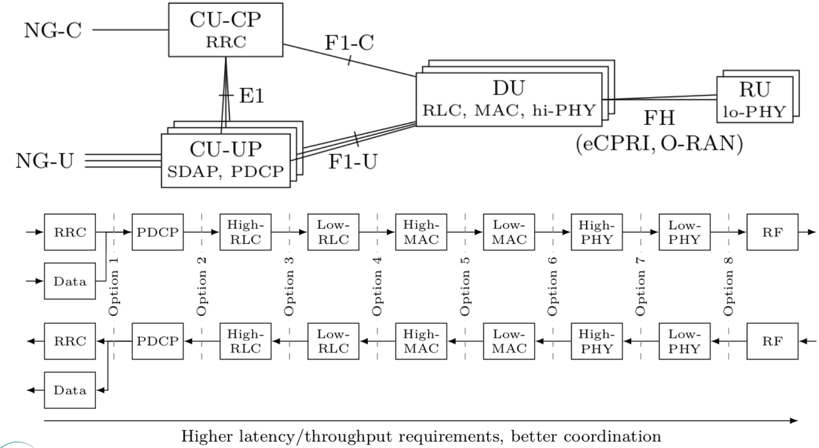
Demos: F1/E1 and Handover

Future work

What are the F1 and E1 splits?



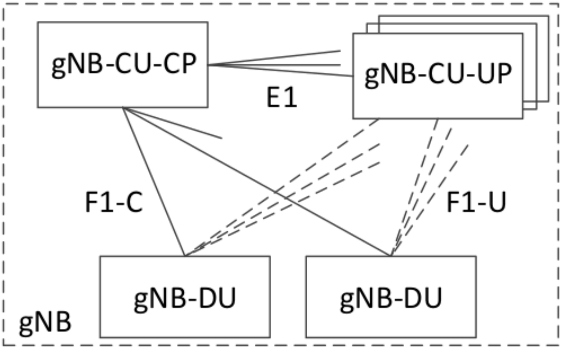
What are the F1 and E1 splits?



Why F1 split?

- TR 38.801 is study item:
  - *split architecture (between central and distributed units) allows for coordination for performance features, load management, real-time performance optimization, and enables NFV/SDN*
  - *Configurable functional splits enables adaptation to various use cases, such as variable latency on transport*
- From TR 38.801 (2016–2017) for split option 2 (i.e., F1):
  - *allow traffic aggregation from NR and E-UTRA transmission points to be centralized*
  - *should be the most straightforward option to standardize*
- Real-time requirements in DU, but not in CU
- Other splits already defined: (n)FAPI, 7.2/eCPRI

Why all this?



- F1 split: centralization and coordination
- E1 split: “load balancing” of user plane, hardware implementation
- Allow “intelligent” handovers, easier to implement as anchor remains
- Also: cloud deployments with relaxed timing requirements

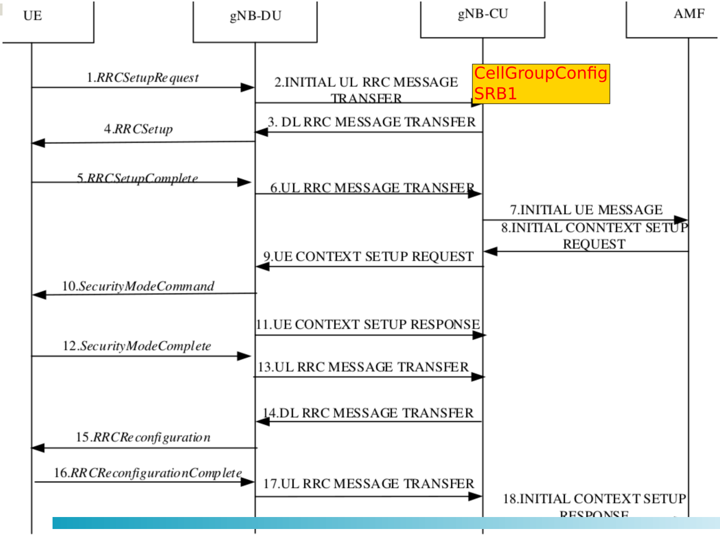
Status last year

- Basic connection setup using F1 internally (remove RRC from DU)
- Basic connection setup using E1
- Interoperability with Accelleran CU (simple use case)
- *Missing*:
  - UE capability handling for MIMO, 256-QAM, ...
  - Multi-DU handling
  - Reestablishment
  - Handover procedures
  - “Full” E1 support
- Challenge: ”overcome” old MAC/RRC/PDCP interface and replace with F1/E1 logic

Connection Setup through F1, TS 38.401

Note: connection setup might not always happen like this, but diagram is sufficiently concise to explain the basic setup.

See documentation



What are F1 messages? TS 38.473

Interface Management

- F1 Setup Request/Response
- F1 Reset (DU handling only)

UE Context Management

- UE Context Setup Request/Response/Failure
- UE Context Release Request/Command/Complete
- UE Context Modification Request/Response/Failure
- UE Context Modification Required/Confirm/Refuse

RRC Message Transfer Messages

- Initial UL RRC Message
- DL/UL RRC Message Transfer

What are E1 messages? TS 38.463

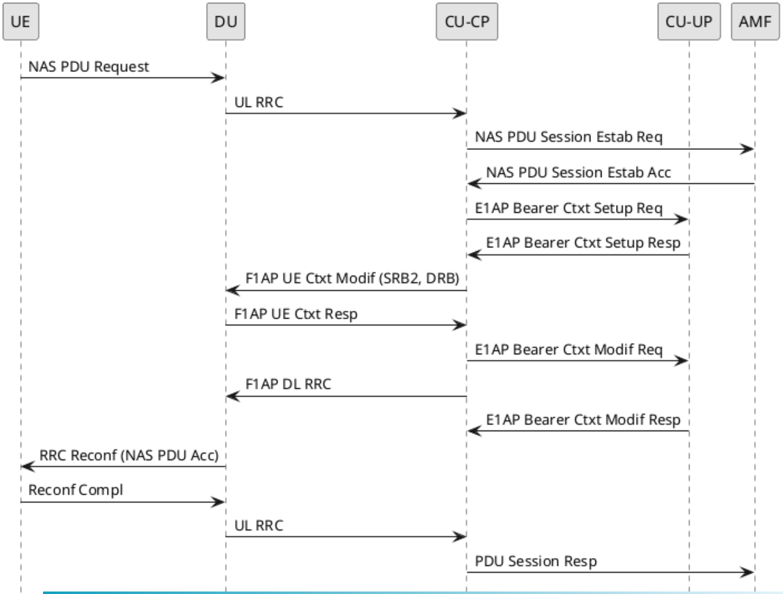
Interface Management

- E1 Setup Request/Response

Bearer Context Management

- Bearer Context Setup Request/Response/Failure
- Bearer Context Release Request/Command/Complete
- Bearer Context Modification Request/Response/Failure (CU-CP/CU-UP initiated)

PDU Session Setup



Current Status

F1

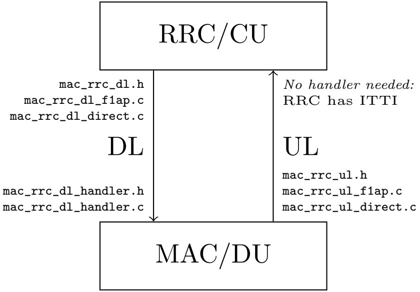
- F1 Interface Management
- F1 RRC Message Transfer
- F1 UE Context Management (Setup, Modification, Release)
- F1 Reset
- Procedures for Reestablishment
- Handling of multiple DUs
- Procedures for intra-CU mobility

E1

- E1 Interface Management
- E1 Bearer Context Management (Setup, Modification, Release)
- Handling of multiple CU-UPs

Implementation

- F1 message types to exchange messages
- E.g., RRC sends ”raw” DL RRC Message
  - Monolithic: calls to handler which puts RRC message into RLC (thread-safe)
  - F1: fills ITTI message to CU task, encoding + send, DU receives messages, calls handler
- F1 is used even in monolithic
- E1: is the same



Implementation – Call Flow Monolithic vs. F1

```
f1_ue_data_t ue_data = cu_get_f1_ue_data(ue_p->rrc_ue_id);
RETURN_IF_INVALID_ASSOC_ID(ue_data.du_assoc_id);
f1ap_dl_rrc_message_t dl_rrc = {
  .gNB_CU_ue_id = ue_p->rrc_ue_id,
  .gNB_DU_ue_id = ue_data.secondary_ue,
  .rrc_container = buf,
  .rrc_container_length = size,
  .srb_id = CCCH
};
rrc->mac_rrc.dl_rrc_message_transfer(ue_data.du_assoc_id, &dl_rrc);
```

↓ function pointer

dl_rrc_message_transfer_direct()

↓ direct function call

dl_rrc_message_transfer()

Implementation – Call Flow Monolithic vs. F1

```
f1_ue_data_t ue_data = cu_get_f1_ue_data(ue_p->rrc_ue_id);
RETURN_IF_INVALID_ASSOC_ID(ue_data.du_assoc_id);
f1ap_dl_rrc_message_t dl_rrc = {
  .gNB_CU_ue_id = ue_p->rrc_ue_id,
  .gNB_DU_ue_id = ue_data.secondary_ue,
  .rrc_container = buf,
  .rrc_container_length = size,
  .srb_id = CCCH
};
rrc->mac_rrc.dl_rrc_message_transfer(ue_data.du_assoc_id, &dl_rrc);
```

↓ function pointer

dl_rrc_message_transfer_direct()

↓ direct function call

dl_rrc_message_transfer()


```
f1_ue_data_t ue_data = cu_get_f1_ue_data(ue_p->rrc_ue_id);
RETURN_IF_INVALID_ASSOC_ID(ue_data.du_assoc_id);
f1ap_dl_rrc_message_t dl_rrc = {
  .gNB_CU_ue_id = ue_p->rrc_ue_id,
  .gNB_DU_ue_id = ue_data.secondary_ue,
  .rrc_container = buf,
  .rrc_container_length = size,
  .srb_id = CCCH
};
rrc->mac_rrc.dl_rrc_message_transfer(ue_data.du_assoc_id, &dl_rrc);
```


↓ function pointer

dl_rrc_message_transfer_f1ap()

↓ ITTI message & handler

CU_send_DL_RRC_MESSAGE_TRANSFER()

↓ ASN.1 encoded msg sent via socket

DU_handle_DL_RRC_MESSAGE_TRANSFER()

↓ decoded message, direct function call

dl_rrc_message_transfer()

Implementation: Improvements since July 2023

- Adopt F1 Setup procedure in monolithic (MR !2304)
  → Simplification of configuration + use UE capabilities
- Change UE ID used at RRC (MR !2214) and handle multiple DUs (MR !2426)
- Adopt E1 logic in monolithic (MR !2383)
  → E1 release, Use of (multiple) CU-UPs based on NSSAI selection
- Implement Reestablishment using F1 (MRs !2304 and !2594)
- Align with O-RAN.WG5.C.1-v11 (MR !2594)
  → Interoperability with COTS O-DU, Improve reestablishment procedure
- Implement F1 handover procedures (MRs !2950+!2786)
  → Allows handover with COTS UE and also OAI UE in RFsim (experimental!)
- Unit tests for F1/E1 message encoding/decoding (MRs !2890, !2985, ...)

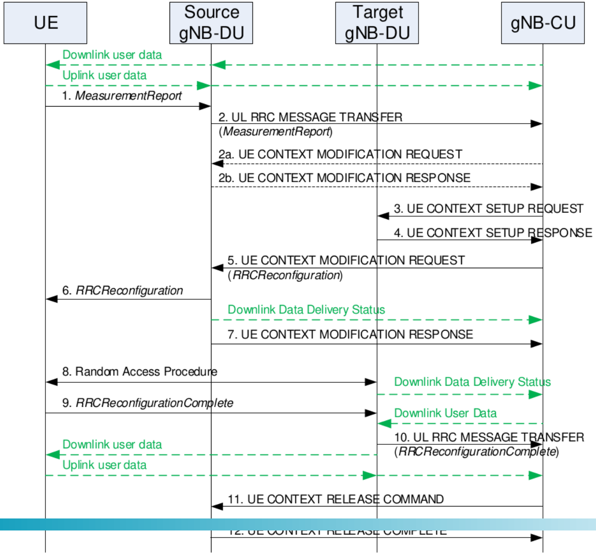
F1/E1 multi-DU/multi-CU-UP demo

See video

Handover: Case study

Note: For more information, see

- handover-tutorial.md
- handover part in rrc-dev.md



F1 handover demo

- Issues last year:
  - Configuration cumbersome
  - Illogical messages, e.g., two NG Setup Requests
  - Unreliable: connection does not always work, release does not always work
  - Data traffic does not seem to work
- None of these should be an issue now

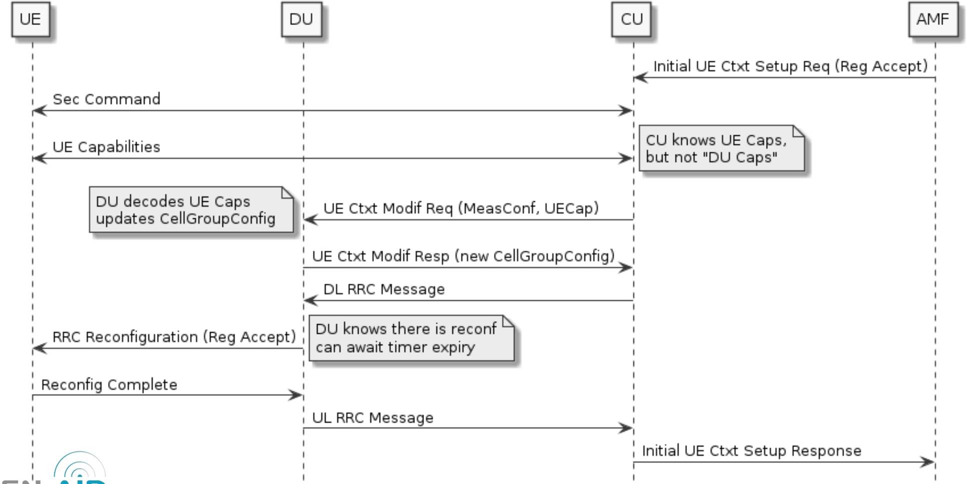
Future work

1. More unit tests for F1, E1
2. NSA?
3. Various issues to resolve for improvements, e.g.
   - Issue 855: Open developments for handover, RRC timers, ...
   - Issue 837: Better handling of CellGroupConfig
4. F1-U improvements: packet loss on F1-U can lead to connection loss, requires NR user plane protocol (TS 38.425)

Thank you for your attention.
Do you have questions? → Q&A

Robert Schmidt (robert.schmidt@openairinterface.org)

Management of UE Capabilities

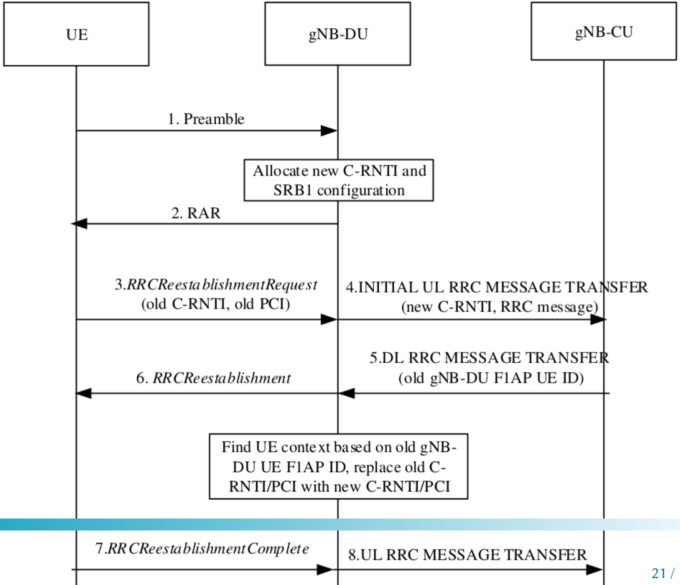


Handling of UE capabilities

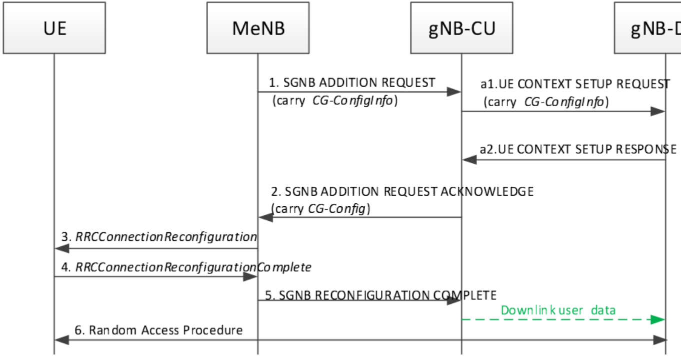
- Currently, RRC handles radio parameters (CellGroupConfig, ServingCellConfigCommon, ServingCellConfig, ...) and UE capabilities
- Example: the CU does not know if DU is able to fulfil radio UE capabilities (256-QAM, etc)
- Insight: Radio-related parameters need to go from CU to DU
- This also affects MIB & SIB1: to be handled at DU, but currently at CU $\rightarrow$ **need to use F1 Setup internally**

Handling of reestablishment and UE IDs

- Reestablishment: MAC direct call
- Handling of IDs must improve: the DU/CU task handles IDs
- Approach: for each UE, store primary UE ID, lookup secondary UE ID when sending F1 message to include these IDs
- Example reestablishment: Message handler updates RNTI, impossible with only the RNTI



Impact on OAI modes other than SA



- Other modes:
  - phy-test
  - do-ra
  - NSA
- They are all similar in nature