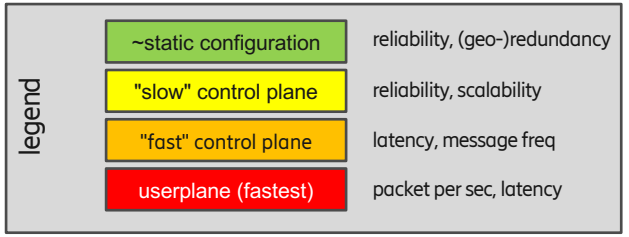
What's next for P4? A journey around missing features

Gergely Pongracz – Ericsson Research

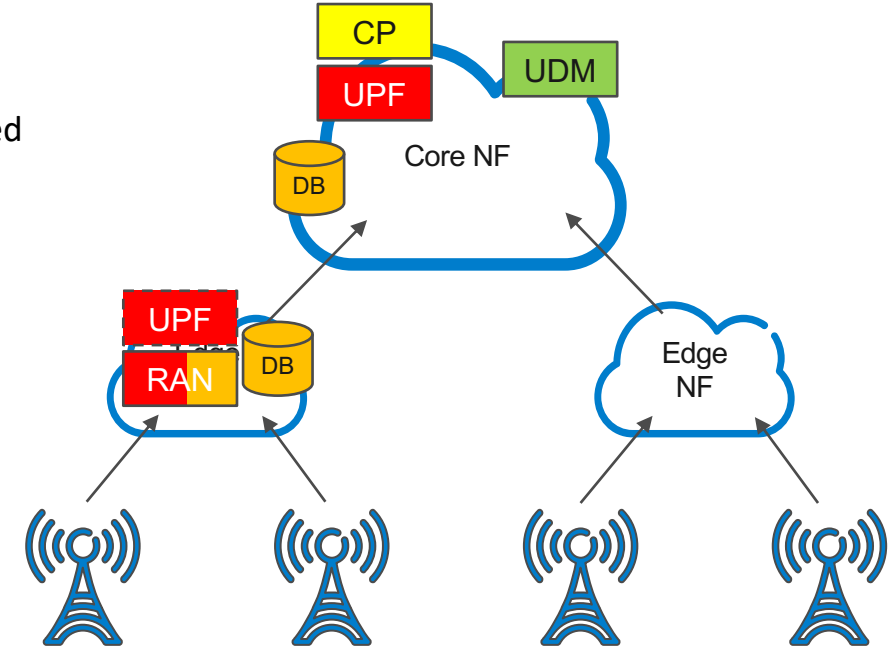
Team members:
Laszlo Molnar – Ericsson Research
Sandor Laki, Peter Voros – Eotvos University (ELTE)

Background: (5G) mobile network

functional view (simplified)



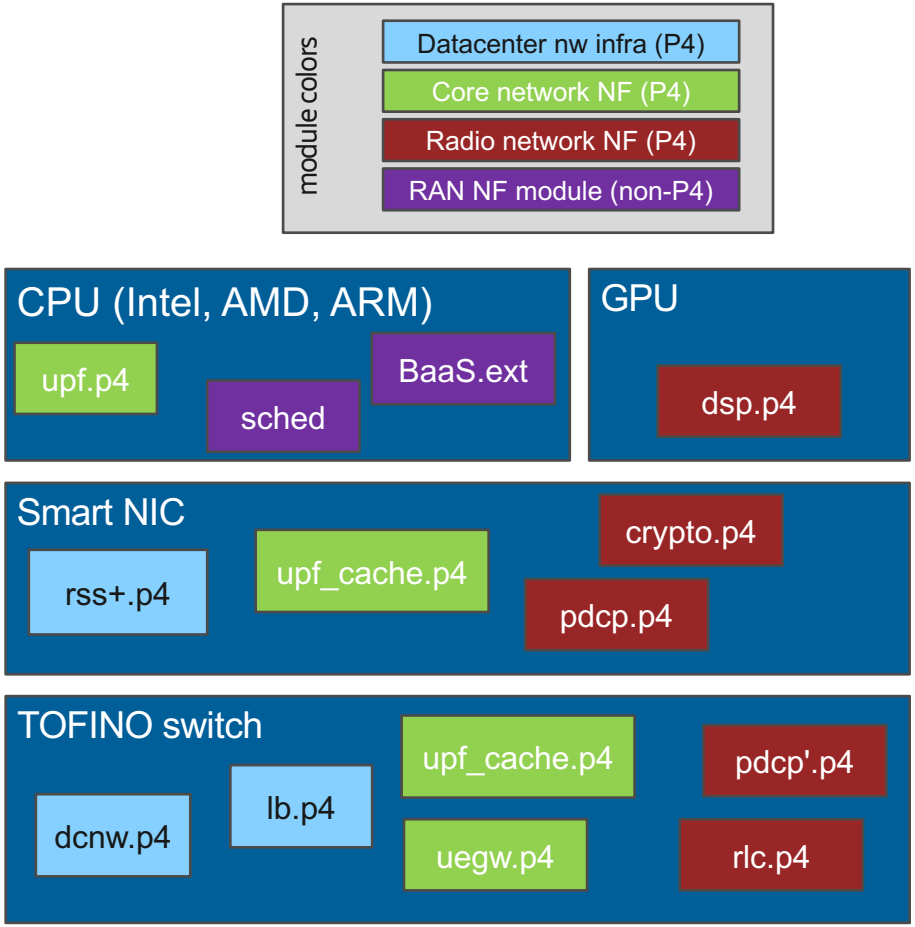
- RAN userplane
  - RLC framing, buffering and retransmission in acknowledged mode (RLC-AM)
  - PDCP framing, air crypto
  - optional hdr compression
  - GTPU encapsulation towards GW
- Core userplane
  - GTPU handling
  - service identification
  - counters, charging
  - optional rate limiting



Which component where?

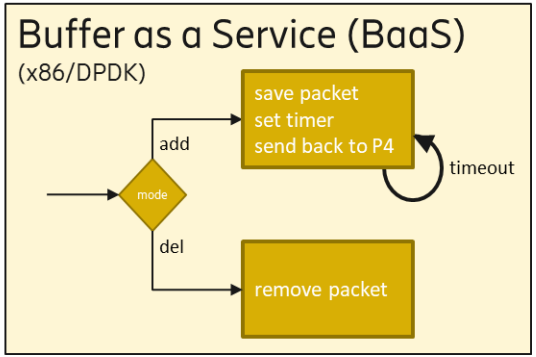
Implementing 5G with P4

- Select the suitable hardware elements for the Network Functions (NFs)
- We envision a “rich-DC” scenario with
  - programmable switch as Top-of-Rack (ToR)
  - x86 or ARM blades
  - with SmartNIC support
  - and (at least some) additional GPUs
- Optimal allocation
  - CPUs could run anything with limited performance
  - offload towards
    - GPUs: better suited for signal processing
    - SmartNICs: cache and crypto functions
    - SmartSwitch: cache and stateless nw functions



Missing feature #1: buffering and timers

- Our use case: the RLC loop between gNB and UE
  - RLC AM goal: make sure that the given packet is forwarded to the UE
  - a hybrid solution was needed
  - part of the pipeline (buffer-as-a-service, BaaS) runs external to the P4 domain
  - very simple task:
    - receive a new packet and set a timer
    - if receive a del signal, delete the packet and the timer
    - if timer reaches zero resend the packet and reset the timer
  - huge diff in performance
    - BaaS performance = 5.5 Mpps per core
    - SmartSwitch (RLC main) ~ 5000 Mpps
- But it's not just RLC: TCP, QUIC also use buffering and retransmission timers

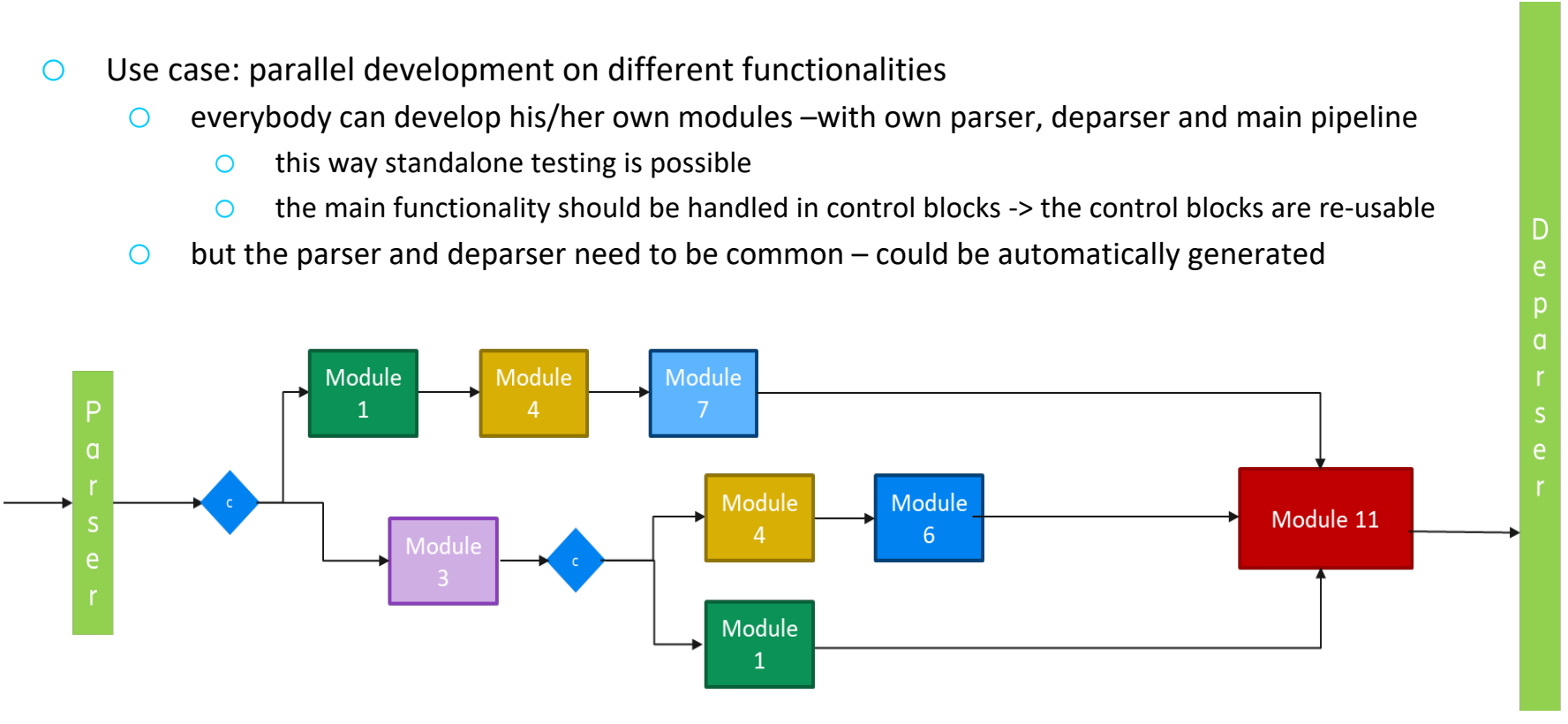


Missing feature #2: clock-based events

- Our use case: scheduler
  - the scheduler collects user packets into a buffer (see gap #1)
  - it waits for the right slot where the user's device listens to the channel
  - then it sends the packets with high accuracy timing
  - so what would it require?
    - Buffering as in gap #1
    - Time sync (e.g. IEEE 1588 – PTP)
    - Internal clock based events
- Other possible use cases for this function: traffic shaping, keepalive messages

(Partly) missing feature #3: modularization

- Use case: parallel development on different functionalities
  - everybody can develop his/her own modules –with own parser, deparser and main pipeline
    - this way standalone testing is possible
    - the main functionality should be handled in control blocks -> the control blocks are re-usable
  - but the parser and deparser need to be common – could be automatically generated



Some further (low prio) missing items

- Multiple access to the same table, counter or register (e.g. routing & encapsulation)
- Conditional header field write in deparser
- Better use of header stacks (to support multi-level encapsulation)

Thank You

gergely.pongracz@ericsson.com