SCReAM (RFC8298) experiments and future

IETF112 - RMCAT

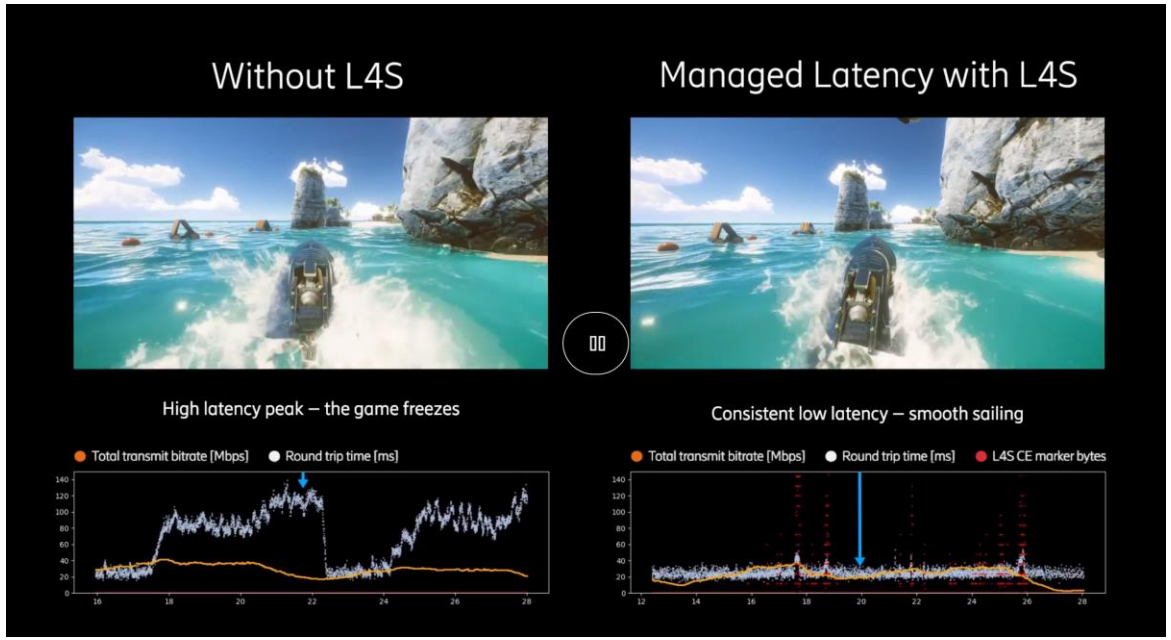
Ingemar Johansson
Ericsson AB
ingemar.s.johansson@ericsson.com

Where SCReAM is (or isn't) used

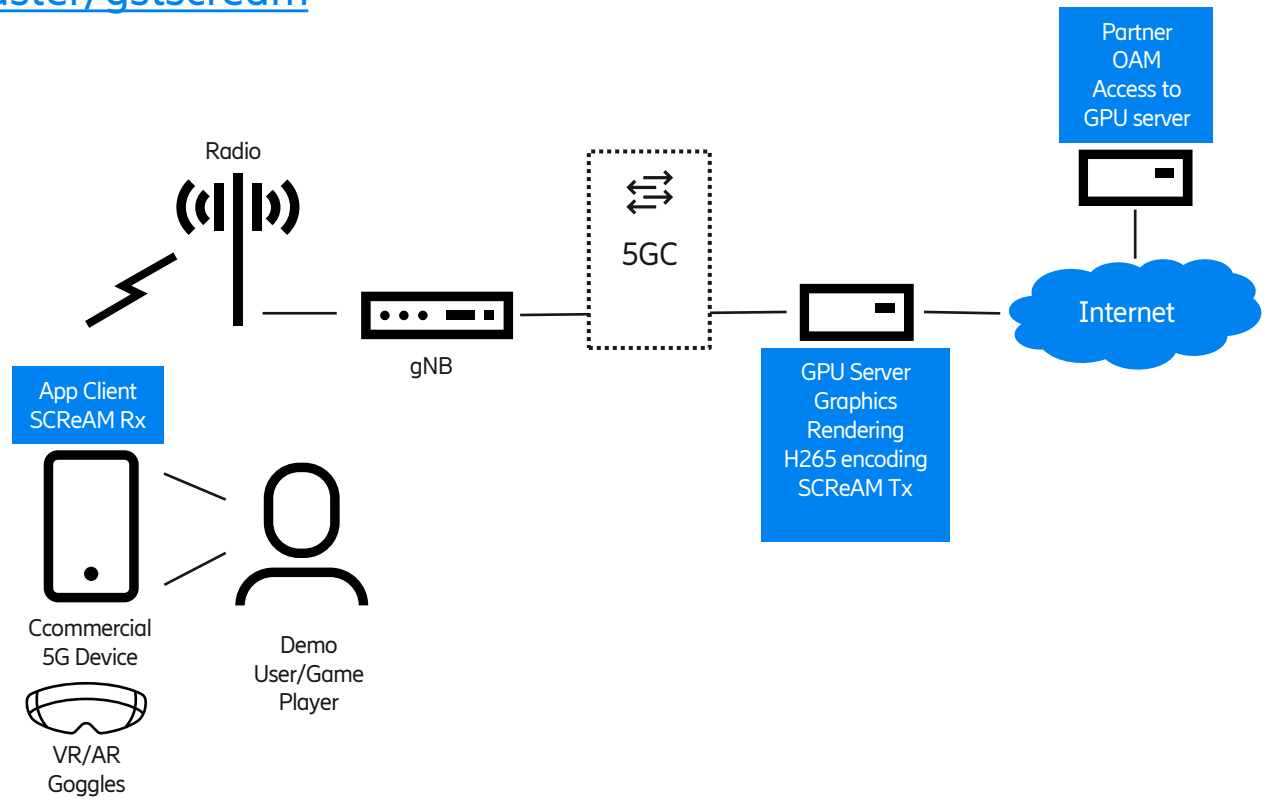
- WebRTC – was implemented for OpenWebRTC but never got traction
  → Never picked up for WebRTC
- Cloud rendered gaming experiments
- Remote driving
- 5G benchmark activities

Cloud rendered gaming

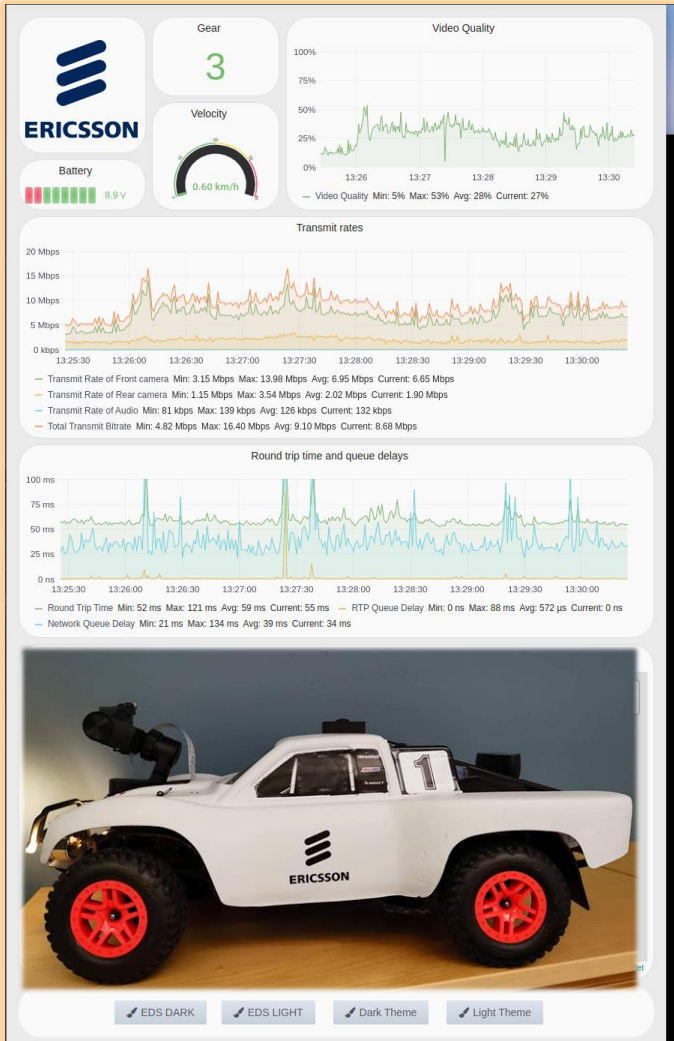
- Game is rendered in (edge) cloud
- Video is encoded and streamed to terminal
- SCReAM Gstreamer plugin from https://github.com/EricssonResearch/scream/tree/master/gstscream

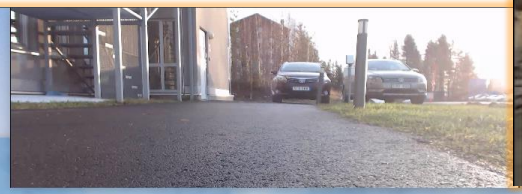


Ericsson and DT demo 5G low latency feature



Remote driving





Traxxas 1:10 RC-Car

NVIDIA Jetson Nano

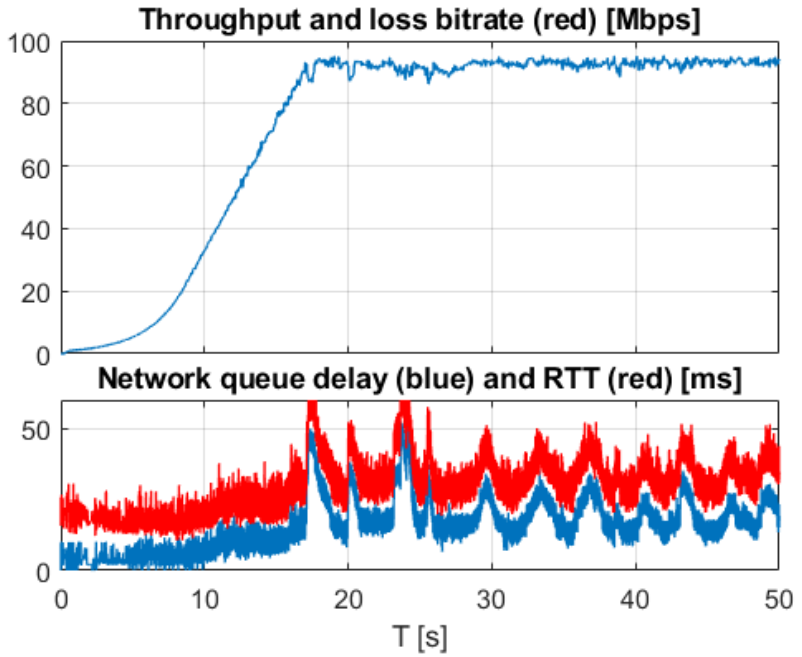
Two cameras

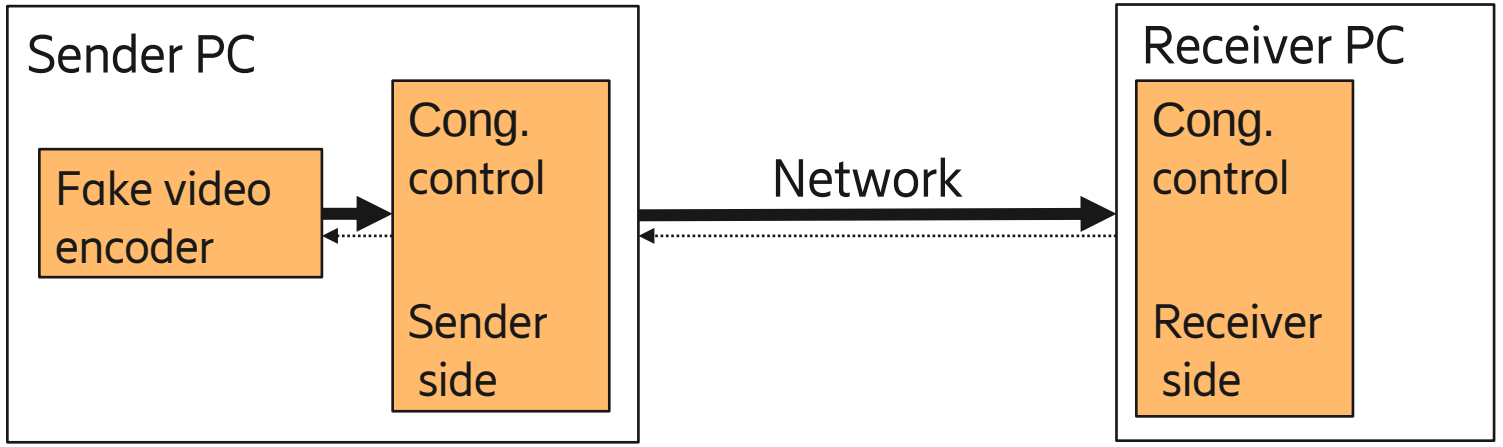
- Front camera prioritized

Max 40Mbps

Benchmarking
SCReAM BW Test tool

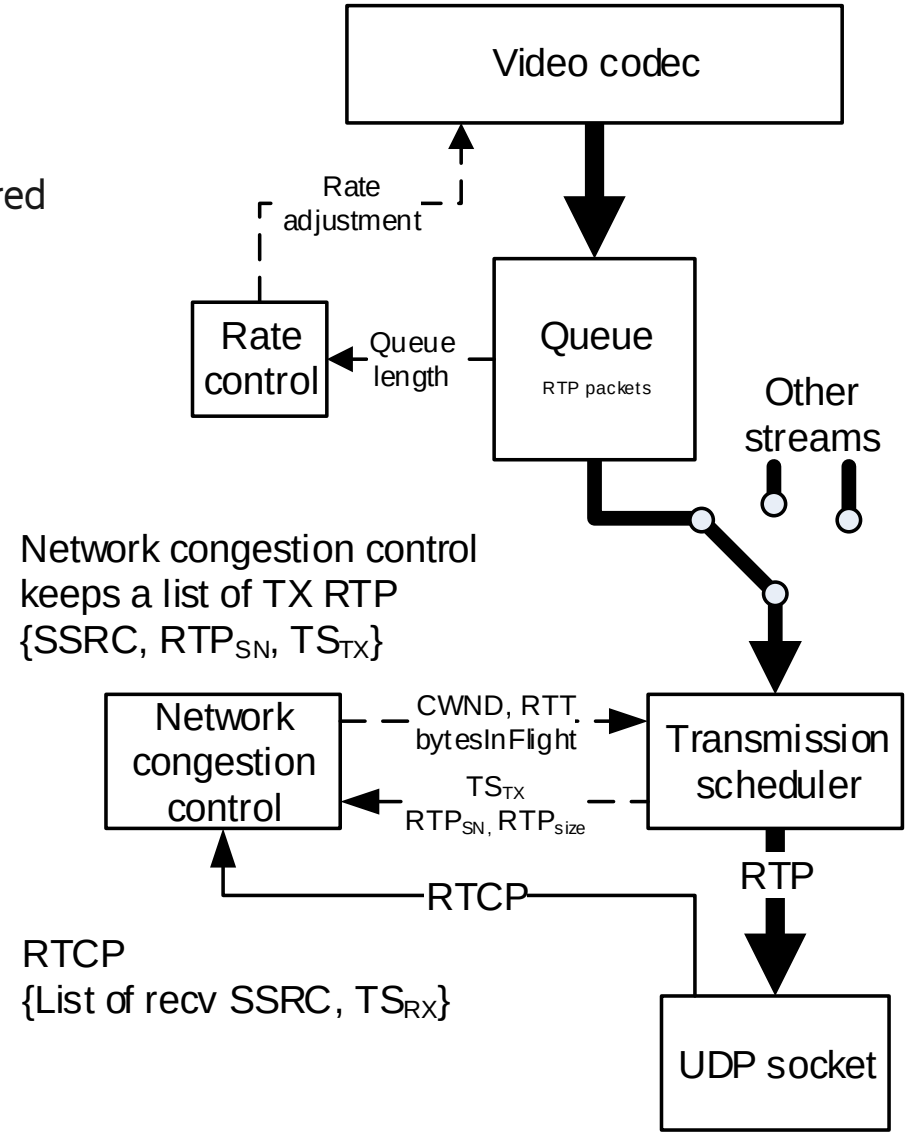
- A network performance test tool that emulates a video coder
  - Fixed rate or…
  - Rate adaptive : Adjusts bitrate to available network throughput, working range 10kbps-500+Mbps
  - I frames and variable frame sizes can be modeled
- Measures : RTT, estimated queue delay, throughput, packet loss, CE marks….
- Source code : https://github.com/EricssonResearch/scream





Findings

- Window based CC is probably good!
  - Radio resource configuration, handover can cause pauses in transmission
  - RTP packets put on hold in sender → can be discarded → Force IDR can be triggered already on sender side → faster recovery
- Feedback rate (~ 1 per 16 RTP packets) is probably overkill
  - But .. more focus on having stable streaming than optimizing ACK overhead
- Video coder matters (80+ %) of the development work !
  - Rate control is not always optimal
    - Video encoder rate control can become confused by frequent updates
    - Sluggish rate control loop can be an issue
  - I-frames are problematic in congested situations
    - Transmit I-frames with care
    - Gradual Decoder Refresh (GDR)
    - Compress I-frames harder



Future

- L4S in running code but not in RFC8298 → future RFC8298bis
- Possibly improved L4S implementation


Questions/Comments?
65.5268037536038N, 22.79877262159975E