RAP Extensions for the Hybrid Configuration Model

Lukas Osswald, Steffen Lindner, Lukas Wüsteney and Michael Menth – published on ETFA 2021

http://kn.inf.uni-tuebingen.de
✓ Three configuration models
  ▪ Fully centralized
    – Central (logical) entities
      • Central User Configuration (CUC)
      • Central Network Configuration (CNC)
    – Admission control is based on a global view
      • Per-flow scheduled traffic for ultra low latency
  ▪ Fully distributed
    – Distributed signaling
      • Resource Reservation Protocol (RAP)
        (This work is based on Draft 0.4)
    – Admission control is based on local information
      • Traffic Shaping, limited scheduling
Hybrid Configuration Model

► Signaling
  ▪ End stations: signal stream properties using a distributed signaling protocol (e.g., RAP)
  ▪ Edge bridges: ensure that requests are directly forwarded to the CNC
  ▪ CNC: takes admission control decision and notifies end stations

► Advantage
  ▪ End stations using distributed signaling have access to globally optimized per-flow scheduled streams

► Research Question: Is RAP ready for the hybrid configuration model? What is missing?
Resource Reservation Protocol (RAP)

- Dynamic, distributed signaling protocol for future TSN (IEEE P802.1Qdd D0.5)
- Overcomes limitations of the Stream Reservation Protocol
  - More streams, more TSN mechanisms (shapers, support for path redundancy, …)

Link-Local Registration Protocol (LRP) (IEEE 802.1CS)

- Protocol for persistently distributing data through the network

Distributed model: Replicate “stream requests” of talkers and listeners along streams path

Figure: Reservation process with RAP in fully distributed model (TAA = Talker Announce Attribute, LAA = Listener Attach Attribute)
Resource Reservation Protocol (RAP)

- Dynamic, distributed signaling protocol for future TSN (IEEE P802.1Qdd D0.4)
- Overcomes limitations of the Stream Reservation Protocol
  - More streams, more TSN mechanisms (shapers, support for path redundancy, …)

Link-Local Registration Protocol (LRP) (IEEE 802.1CS)

- Protocol for persistently distributing data through the network

Hybrid model: Transport “stream request” of talker and listener to CNC using LRP Proxy Mode

- Edge bridges via LLDP
  - Application information
  - Address information of proxy (IP + Port)
- End stations connect via TCP
- End stations and Proxy exchange RAP attributes

Figure: RAP + LRP protocol stack
Resource Reservation Protocol (RAP)

- Dynamic, distributed signaling protocol for future TSN (IEEE P802.1Qdd D0.4)
- Overcomes limitations of the Stream Reservation Protocol
  - More streams, more TSN mechanisms (shapers, support for path redundancy, …)

Link-Local Registration Protocol (LRP) (IEEE 802.1CS)
- Protocol for persistently distributing data through the network

Hybrid model: Transport “stream request” of talker and listener to CNC using LRP Proxy Mode

RAP Proxy System
- What does it do?
- Where is it located in hybrid model?
- Is RAP’s data model complete?
Where to place the RAP Proxy?

- Model as in standardization
  - RAP-CNC would additionally have to
    - Speak RAP ("user-specific protocol")
    - Keep track of new stream requests
    - Trigger resource reservation procedure
  → Tasks of a CUC in fully centralized model!

- We propose a RAP-CUC
  - Handles LRP+RAP signaling
  - Extract information relevant for admission control
  - Manage life-cycle of streams
  - Handle resource reservation procedure with CNC

- Advantage
  - CNC has the same responsibility in all models
    - CNC does not handle any user-specific protocol
▲ Protocol Connector (PC)
  ▪ Connects to end stations and performs user-specific signaling

▲ Stream Management (SM)
  ▪ Manages the life cycle of a stream
  ▪ Generic function

▲ CNC Connector (CNCC)
  ▪ Requests resources from a specific CNC implementation
  ▪ Webhook Handler
    – Callback for notifying finished computations

▲ Modular design
  ▪ Allows support for additional user-specific protocols or CNCs

---

**Figure:** Architecture of RAP-specific CUC entity
Data Model Comparison

- RAP must include data defined by the TSN User/Network Interface (UNI) (IEEE Std 802.1Qcc)
  - We compare the data required for reservation of
    - Per-flow scheduled streams using time-aware end stations
    - With the MSRP Traffic Specification of RAP

- User → Network (stream properties, QoS requirements)
  - Missing data can be included in MSRP Traffic Specification

- Network → User (reservation status, configuration data)
  - RAP lacks an option to transport configuration data to the end stations
    - E.g., transmission start of a stream to the Talkers
  - Missing data can be attached to Listener Attach Attribute

### Part of UNI (U→N) for per-flow scheduled streams

<table>
<thead>
<tr>
<th>Stream-id</th>
<th>Talker's MAC address</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Data-frame-params: priority, VLAN ID, Dest. MAC</td>
</tr>
<tr>
<td></td>
<td>TrafficSpecification: interval <em>(resolved in D0.5)</em></td>
</tr>
<tr>
<td></td>
<td>TrafficSpecification: max frames per interval</td>
</tr>
<tr>
<td></td>
<td>TrafficSpecification: max frame size</td>
</tr>
<tr>
<td></td>
<td>Time-aware: earliest-transmit offset</td>
</tr>
<tr>
<td></td>
<td>Time-aware: latest-transmit offset</td>
</tr>
<tr>
<td></td>
<td>User-to-network-requirements: maximum latency</td>
</tr>
<tr>
<td></td>
<td>Listener's MAC address</td>
</tr>
<tr>
<td></td>
<td>User-to-network-requirements: maximum latency</td>
</tr>
</tbody>
</table>
We analyzed RAP for its applicability in the hybrid configuration model
- We redefined the hybrid model by adding a CUC including the RAP Proxy
- We propose an architecture for a CUC entity
- We analyzed the data model of RAP and proposed extensions to support per-flow scheduling in the hybrid model

A partial implementation of the RAP-CUC is published on GitHub
- [https://github.com/uni-tue-kn/rap-cuc](https://github.com/uni-tue-kn/rap-cuc)
  - It can be used for developing prototypes with other user-specific protocols and other CNC entities
Thank you for your attention. Questions?

RAP Extensions for the Hybrid Configuration Model
Lukas Osswald, Steffen Lindner, Lukas Wüsteney and Michael Menth – published on ETFA 2021

http://kn.inf.uni-tuebingen.de