IEEE 802.1 TSN
MSRP, LRP, RAP, and YANG

Feng Chen Norman Finn Rodney Cummings

Disclaimer

This presentation on MSRP, LRP, RAP, and YANG includes descriptions of work in progress in IEEE 802.1. It expresses the current intentions of certain contributors to, and editors of, these documents. The eventual standards output from IEEE 802.1 (if any!) may differ substantially from these intentions.

RAP

Feng Chen, Siemens AG

RAP: Resource Allocation Protocol Overview

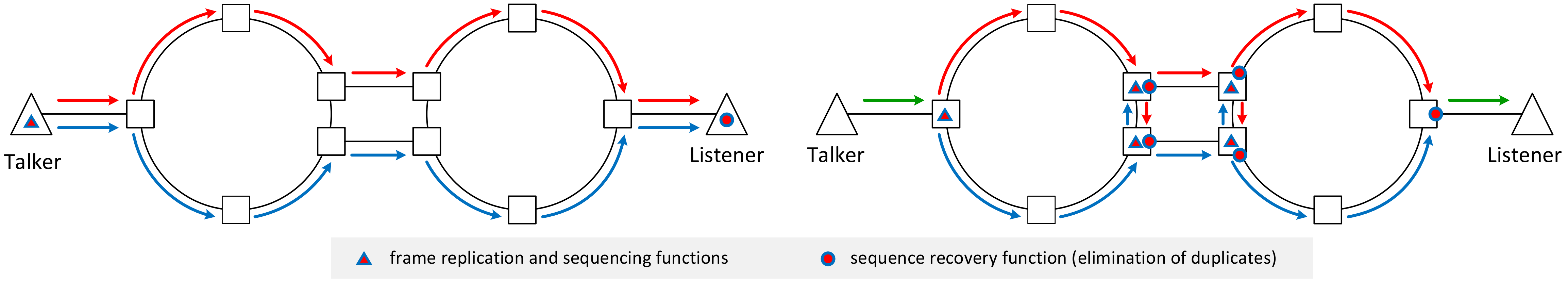
- A recently approved IEEE 802.1 TSN project (IEEE P802.1Qdd webpage)
- A peer-to-peer signaling protocol built upon P802.1CS Link-local Registration Protocol
- Providing dynamic reservation services for streams that desire to use the TSN QoS functions to achieve bouded-latency, zero congestion loss and optionally high reliability
- Preserving the principle of 802.1Q MSRP and offering expanded capabilities to support:
  - a larger number of streams than MSRP benifiting from the use of LRP
  - 802.1 queuing/transmission functions suitable for peer-to-peer reservation
  - stream reservation over redundant paths (e.g. 802.1CB FRER)
  - better proxying capability
- Providing a facility for backwards compatibility and interoperability with MSRP
- Not limited to use in bridged networks and having potential for use as a generic reservation protocol in a DetNet network

RAP: A SRclass based Reservation Approach

- A Stream Reservation Class (SRclass)
  - representing a QoS delivery class within a reservation domain that provides bouded latency and bandwidth guarantees for the streams using that class
  - characterized by {priority (corresp. a queue), bandwidth, measurem. interval, max frame size, trans. selection algorithm (shaper)} - all configurable by management
  - reporting always the worst-case latency calculated by taking account of the max. reservable bandwidth, independently of the currently reserved streams
- RAP expands the use of SRclass with TSN queuing/shaping functions
  - enabling use of various shapers (MSRP in AVB supporting only Credit-based Shaper)
  - taking advantage of the “topology-independent per-hop latency calculation“ capability of the TSN shapers, such as 802.1Qch Cyclic Queuing and Forwarding (CQF) and P802.1Qcr Asynchronous Traffic Shaping (ATS)

RAP: Reservation over Redundant Paths

- RAP supports dynamic reservation and configuration of the streams that require additionally redundant transmission over multiple paths e.g. using IEEE 802.1CB FRER
    - making reservation for each member stream along its own path
    - specifying rules for propagating RAP attributes at frame splitting and merging points
    - autoconfiguring the redundancy functions, e.g. FRER sequence recovery function



End-to-End Redundancy with 802.1 CB FRER in End-stations

Ladder Redundancy with 802.1 CB FRER in Bridges

RAP for DetNet

Norman Finn, Huawei

The Logical / Physical problem

- Es war einmal*, there were Routers and Bridges. They were separate boxes, with distinct functions.
- Today, the logical topology revealed by traceroute often has nothing whatsoever to do with the physical topology of boxes and wires. One box may be 500 virtual routers, one logical router may be spread out over a continent; any combination of logical vs. physical is possible.
- The **logical topology** is vital to — in fact, it defines — the **connectivity** of the network.
- But, any QoS **resource reservation** protocol requires knowledge of the real boxes and wires – the **physical topology**.

* Once upon a time,

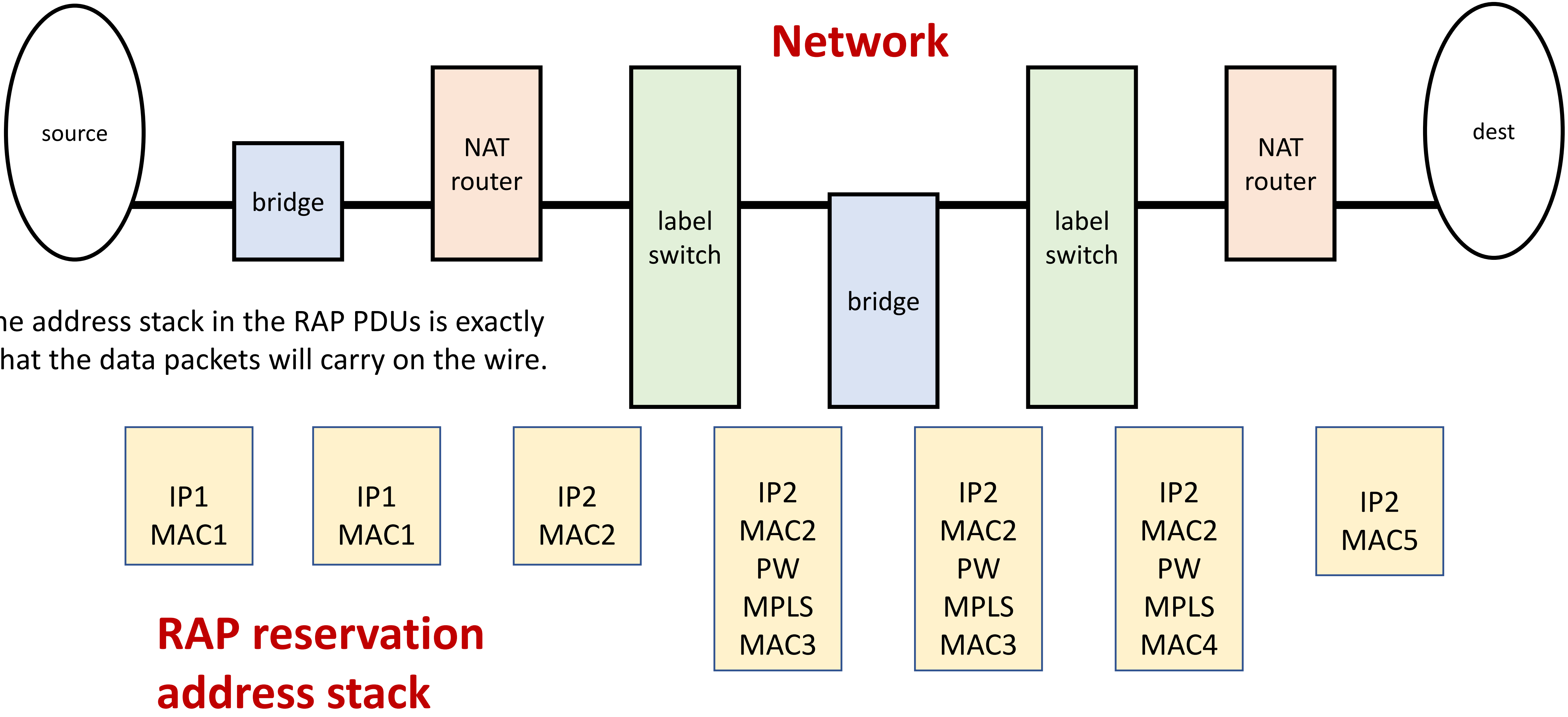
IEEE 802.1Qdd Resource Allocation Protocol

- RAP will **adhere rigidly** to the **standard layering** model and the **logical** network topology to define the path from Talker (sender) to Listener (receiver).
- But, RAP will also **follow the physical boxes and wires**, while passing registration information and reserving resources.

How?

- RAP follows the physical wires. The source and destination of RAP PDUs are boxes, not logical bridge/router functions.
- On each hop, a RAP reservation carries the data packet's entire address stack, from the wire up to the application, for that hop.
- The address stack can be extended, trimmed, or altered from hop to hop.

RAP example: TSN + DetNet

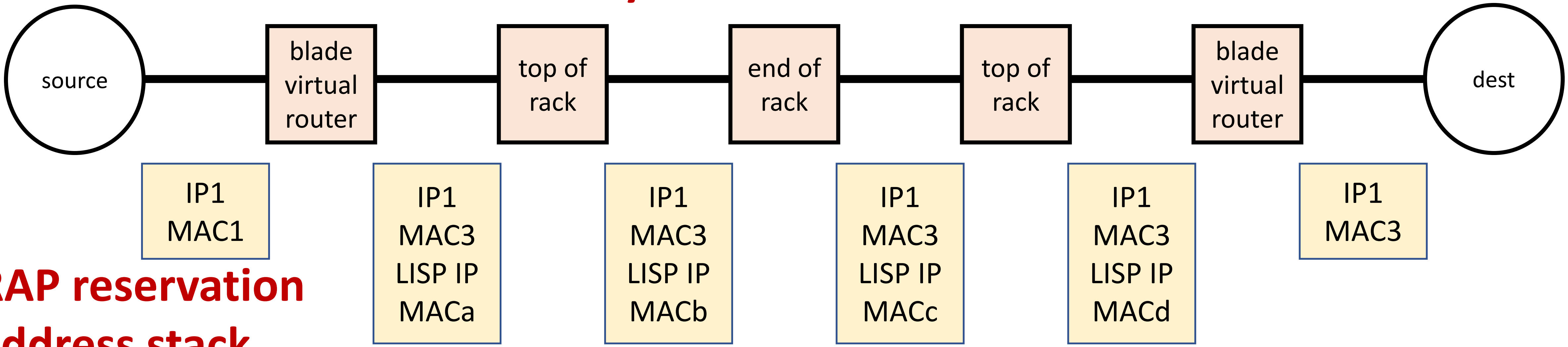


The address stack in the RAP PDUs is exactly what the data packets will carry on the wire.

RAP example: Data center

The address stack in the RAP PDUs is exactly what the data packets will carry on the wire.

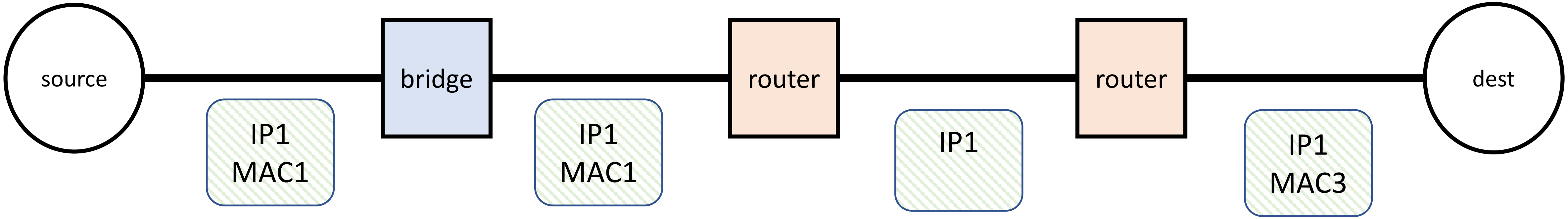
Physical Network



RAP reservation address stack

(first blade makes all routing/bridging decisions)

Note that intermediate logical addresses (MAC2) are irrelevant to RAP; they are part of the logical/physical simulation

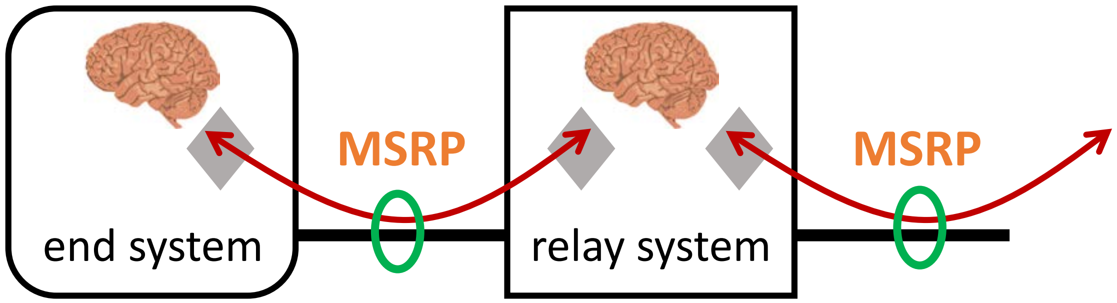


Logical Network and virtual addresses

Proxying

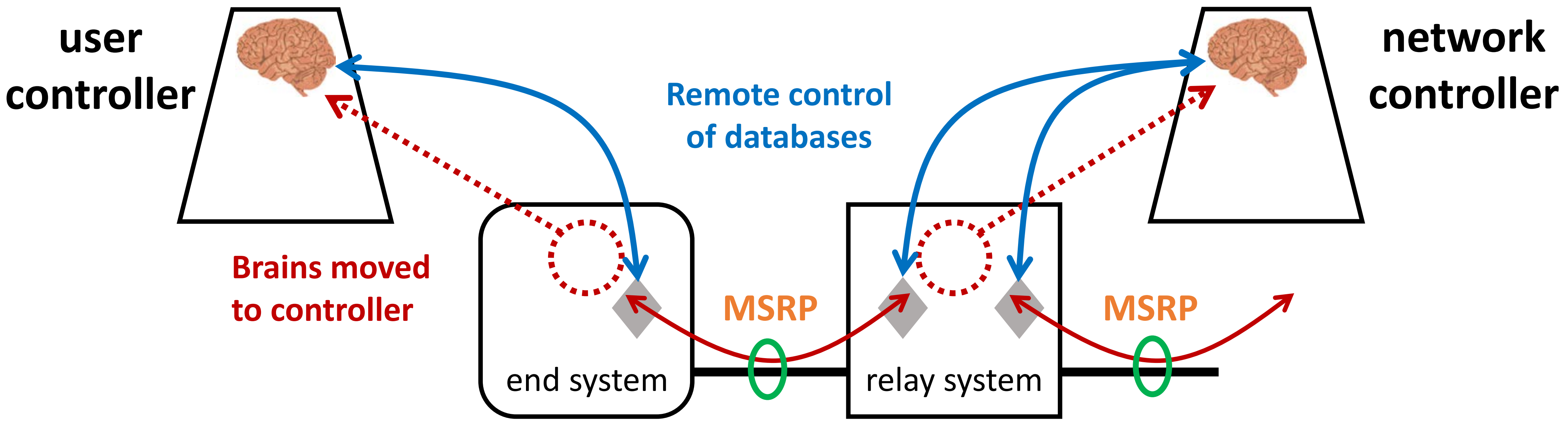
Norman Finn, Huawei

IEEE Std 802.1Q-2018 clause 35
MSRP: Multiple Stream Reservation Protocol



- Hop-by-hop resource reservation protocol.
- MSRPDUs are Ethernet frames, and use the same links as the TSN streams.
- Every MSRP attribute is tied to one particular target link.

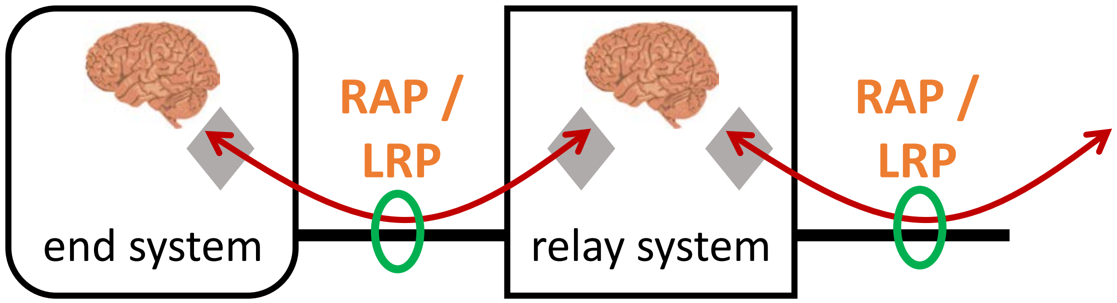
SRP + IEEE Std 802.1Qcc-2018



- MSRP **STILL** information follows the data path.
- Every MSRP attribute is **STILL** tied to one particular target link.

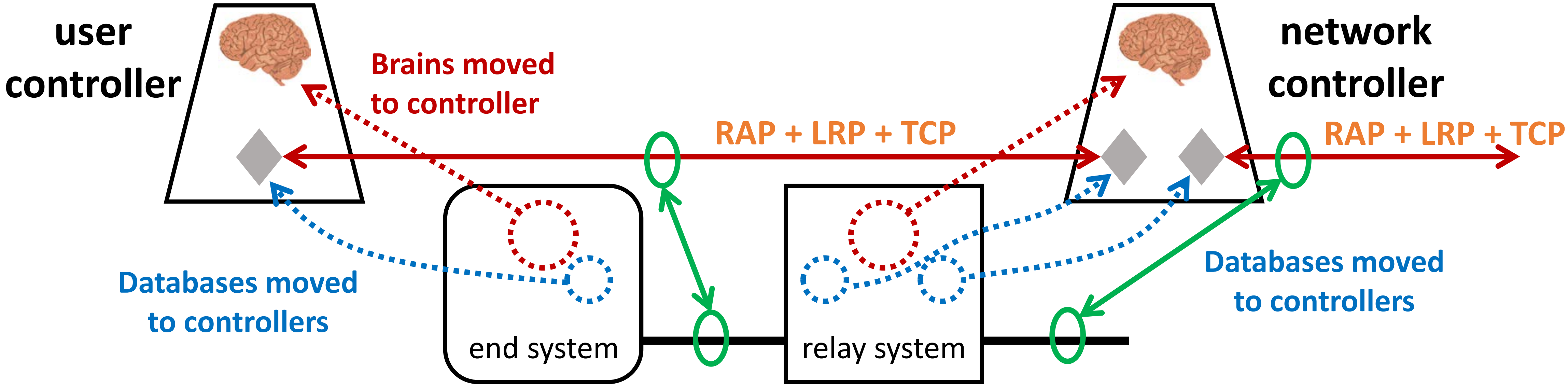
P802.1CS Link-local Registration Protocol (LRP)
P802.1Qdd Resource Allocation Protocol (RAP)

Native systems



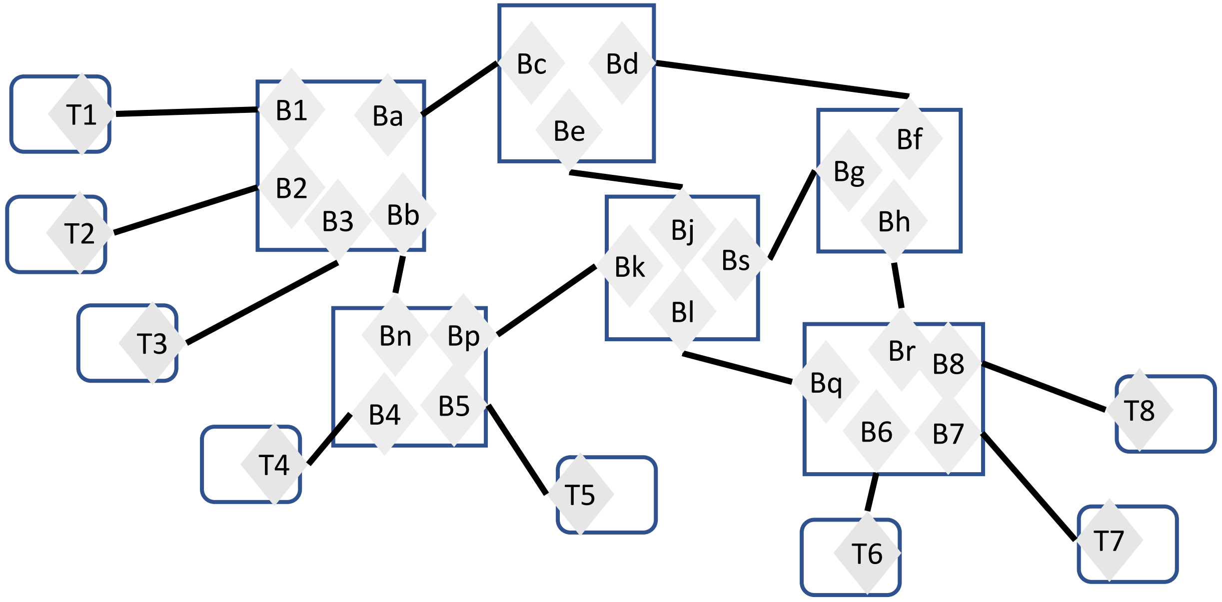
- RAP/LRP information **STILL** follows the data path.
- RAP/LRP PDUs use a link-local transport protocol
- Every RAP attribute is **STILL** tied to one particular target link.

LRP + RAP Proxy/Slave systems



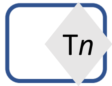
- LRP/RAP information **no longer** follows the data path.
- LRP/RAP PDUs flow over a **TCP** connection.
- But, every RAP attribute is **STILL** tied to one particular target link.

Example 1: Peer-to-peer

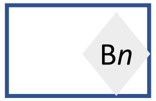


- Eight Talkers. Six Bridges.
- Running MSRP or RAP/LRP peer-to-peer.
- (Listeners not shown.)

Key:

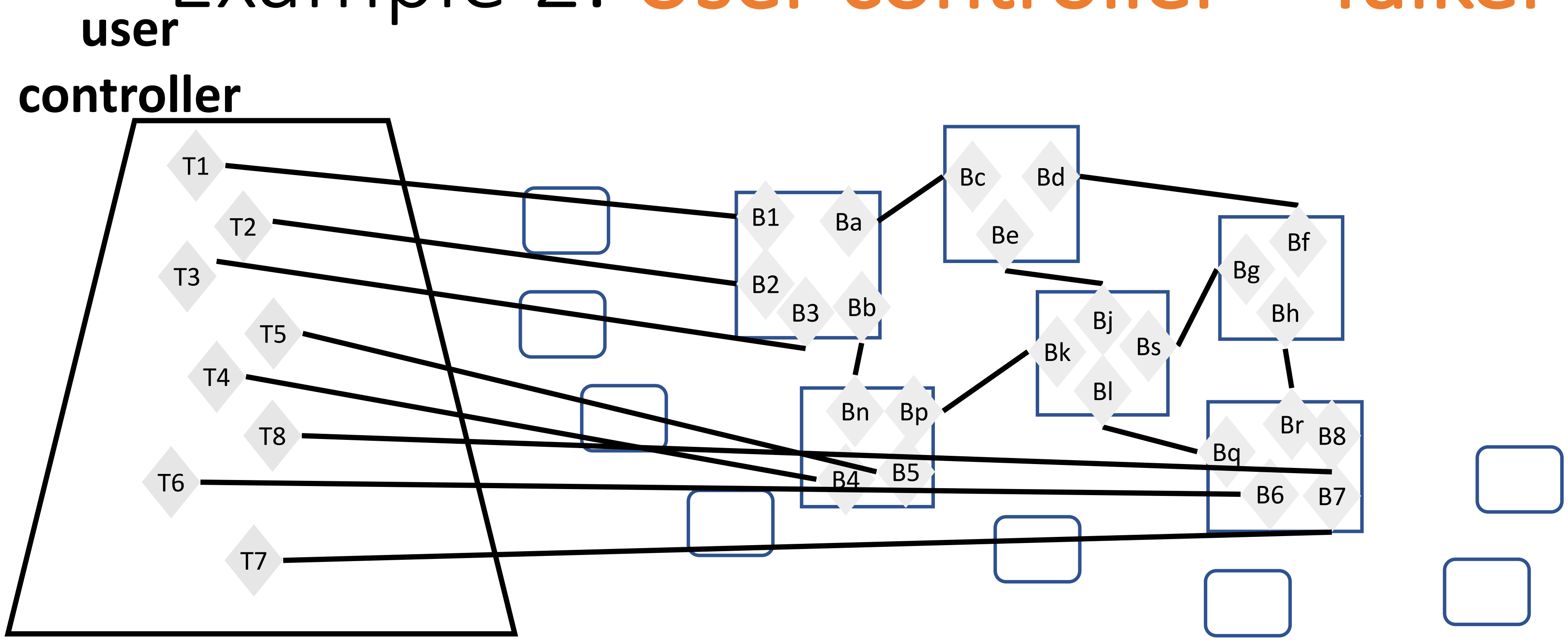


Applicant + Registrar database living in a Talker



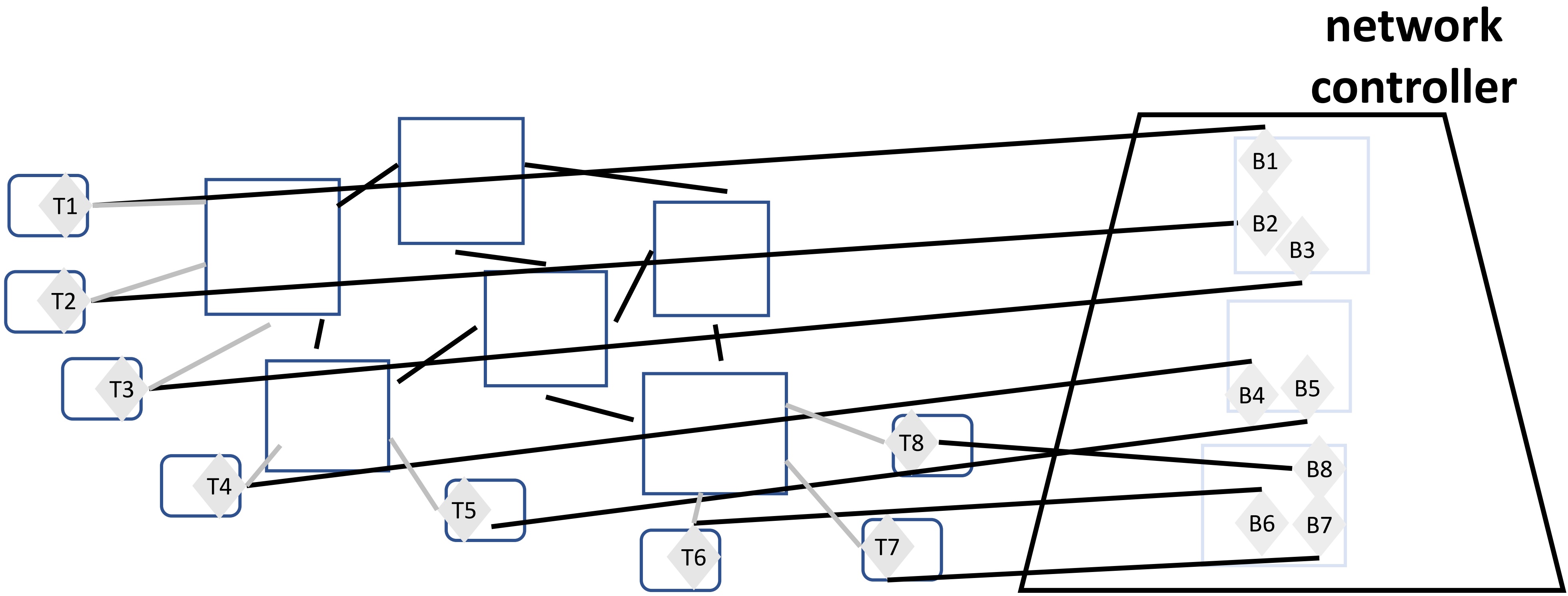
Applicant + Registrar database living in a Bridge

Example 2: User controller = Talker Proxy



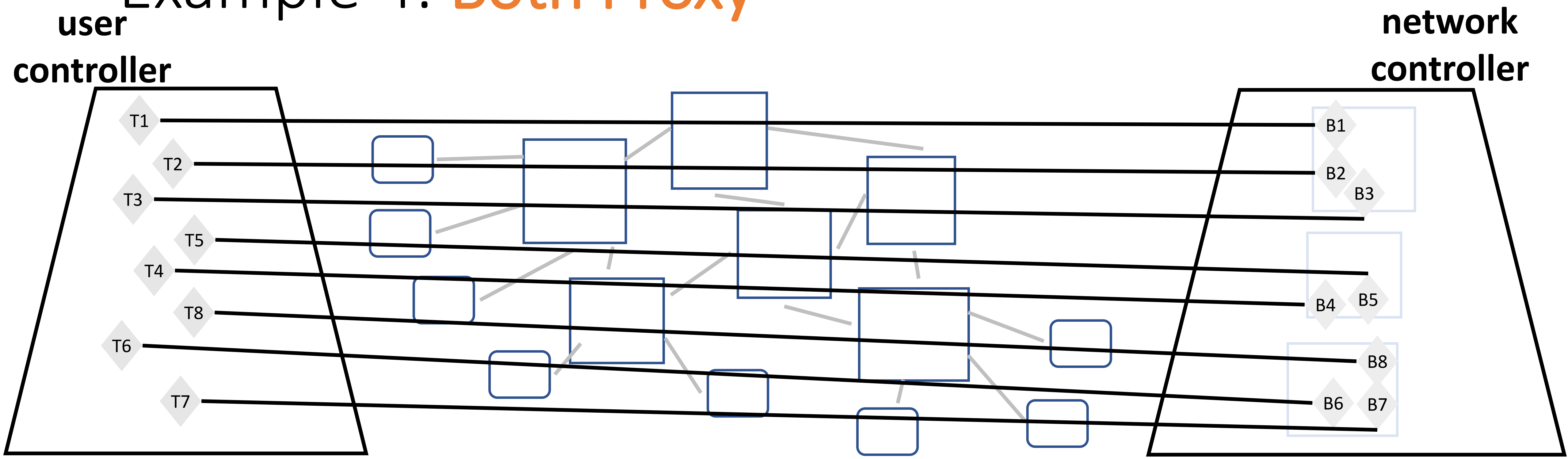
- **User controller** pretends to be **8 Talkers** using RAP/LRP or 802.1Qcc
- **Bridges don't care** whether connected to controller or individual Talkers – it's the same **RAP** protocol.

Example 3: Network controller = Edge Bridge Proxy



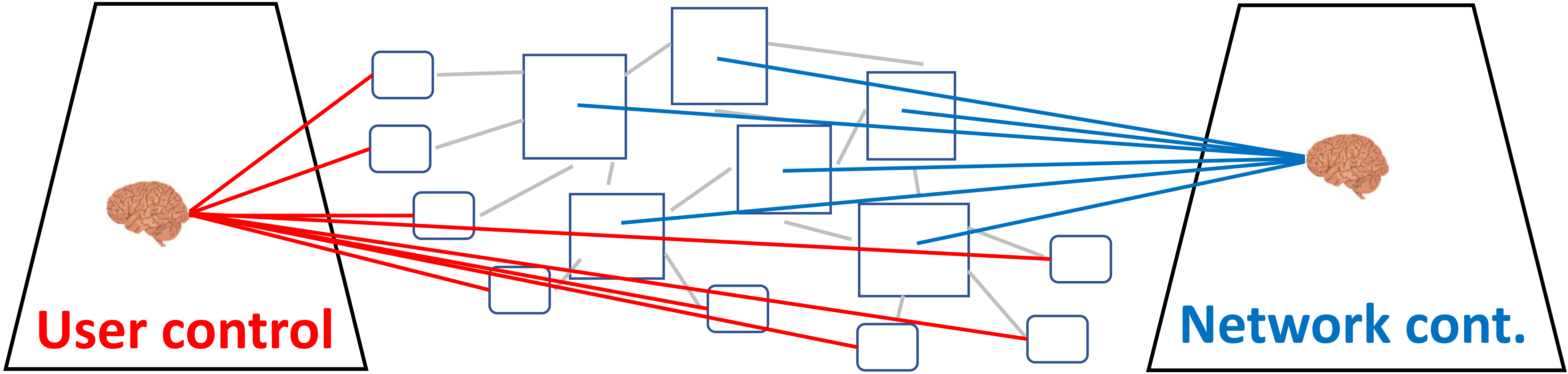
- **Network controller** pretends to be **6 Bridges** using RAP/LRP or 802.1Qcc
- **Talkers don't care** whether connected to controller or individual Bridges – it's the same **RAP** protocol.

Example 4: Both Proxy



- **User controller** proxies Talkers, **network controller** proxies for Bridges.
- Controllers can still the **RAP/LRP** over a single TCP connection, and still **don't care** whether the other end is a controller or an individual.

Underlying assumption: Controllers have management channels to their devices



- Some means of effecting the decisions made as a result of the resource reservations is required.
- This is, presumably, based on **YANG** or **MIBs**.

YANG-based Solution

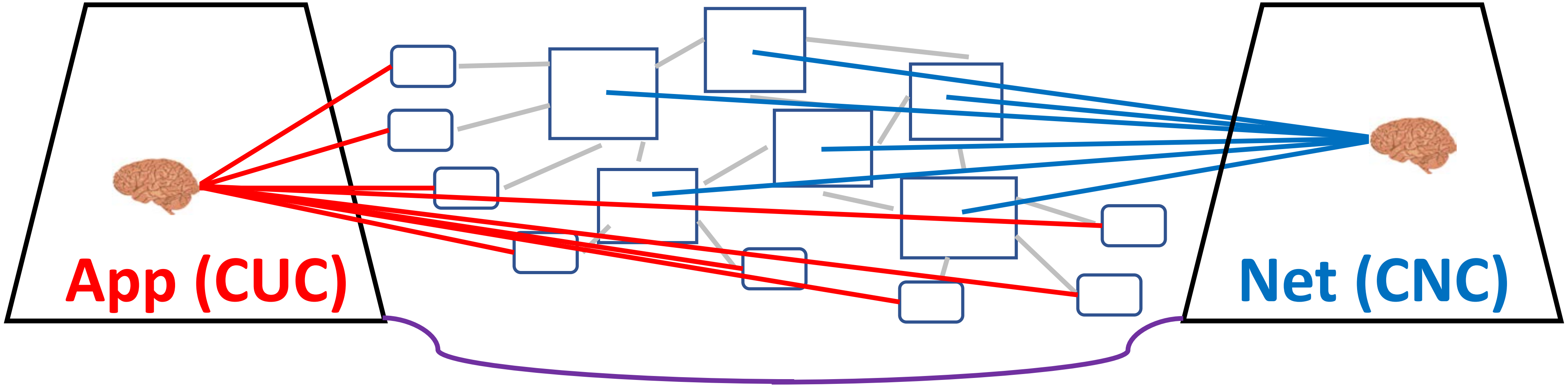
Rodney Cummings, National Instruments

Why? (Rationale)

- For the next 3 slides, let's define "end-system" as
  - An application product not sold as switch or router... not "net infrastructure"
  - Includes a product that uses switch chip/IP internally, but not sold that way
- Fact: Some end-systems will never invest in net infrastructure protocols
  - Not MSRP, not RAP, not PCEP, not NETCONF, not RESTCONF, not CoMI...
- Fact: Many end-systems don't know TSN/DetNet info on their own
  - Example: End I/O devices don't know the network-wide control loop cycle time, since that is tied to physical / application requirements, so they learn TSpec from an application controller (CUC)
- Fact: Protocols for application (user) controller are well-established

What? (Proposal)

- For app protocol, TSN/DetNet does **nothing**
  - Existing protocols can integrate 802.1Qcc-2018 / detnet-flow-info as a guideline
- Net protocol is **management**, so TSN/DetNet keep doing YANG



- TSN/DetNet UNI: Avoids proxies; Best option is YANG
  - App to Net: Latency and loss requirements; TSpec; End-system packet info
  - Net to App: Status of flow reservation; Configuration for NIC (if any)

How? (Next Steps)

- UNI is best done as a cooperative project in TSN/DetNet
  - Two sets of YANG would reduce momentum in application industries
- Step 1: Decide whether to do a UNI in YANG
- Step 2: Decide where to standardize UNI cooperatively (TSN or DetNet)
- Step 3: Decide terminology to use in the YANG
  - Different terminology is fine in each standards organization
  - Different terminology is not a valid reason to have two sets of YANG
    - Software developers in applications (use cases) don't care
- Step 4: Use 802.1Qcc-2018 / detnet-flow-info as the basis
- Step 5: Add other features (e.g. topology exchange)

Thank you