# Hierarchical CUC/CNC management model

Amin Abdul
Molex Canada

May 2020

# Overview

Problem statement: It seems that neither IEEE standards nor “IEC/IEEE 60802 D1.1 TSN Profile for Industrial Automation” is prescribing any reference management model which deals with the hierarchical CUC/CNC scenario.

Goals:

- Abstract system models for configuration and monitoring in hierarchical CUC/CNC scenario.
- Identify requirements, gaps and potential remedies.
- Application examples.

Previous/Parallel work:

[4], [6], [7], [8], [9], [10], [12], [13], [14], [15], [16].

- Disclaimer – All models in this presentation may be known to the working groups.

# Summary (v1)

## Step 1 (Framework):

- Streamline configuration and monitoring planes.
- Based on generic and reusable system models.
- Model driven networking (try to avoid additional protocol).
- Identify gaps such as, missing abstractions and data models.

## Step 2 (Primary Application):

- Holistic view of system configuration and monitoring planes.

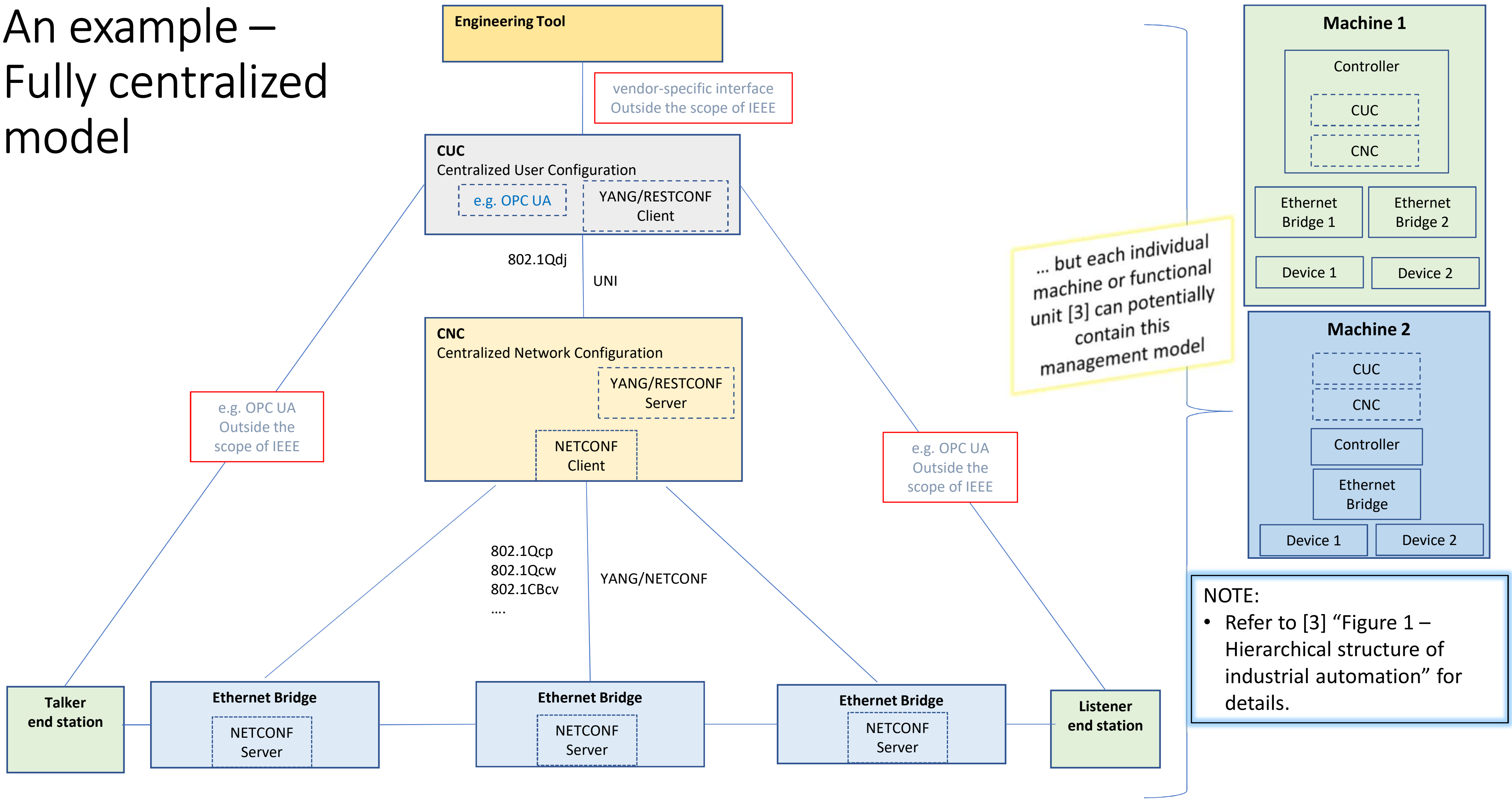
# Summary (v2)

- Incorporated P60802 working group inputs and suggestions.
- TSN Adapter domain [12] and its relation to inter-TSN domains communication.

Step 3 (Other Applications):

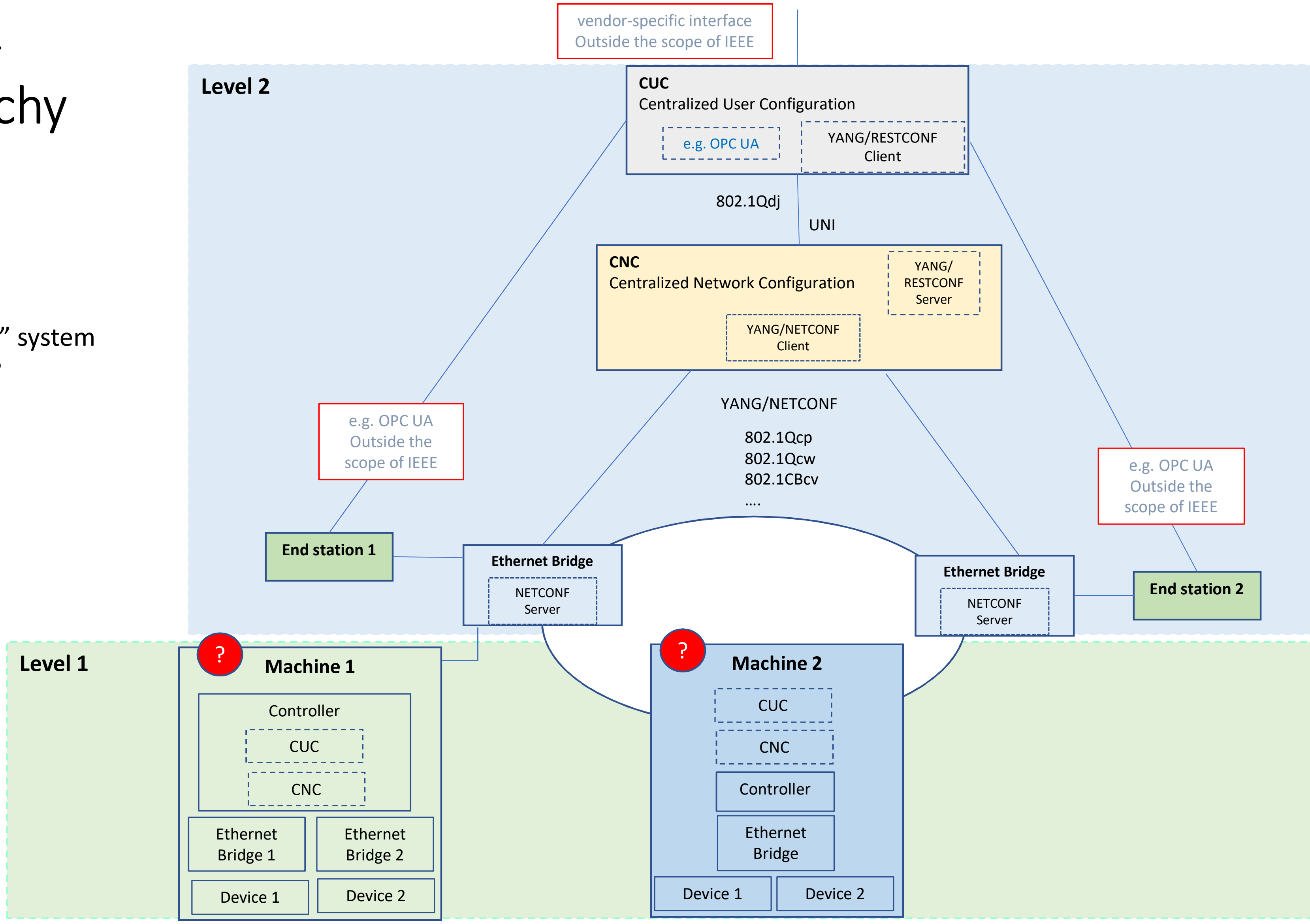
- How the end-station and the bridged-end-station system concepts can help us to deals with the concerns like:
  - TSN domain vs gPTP domain.
  - Composing inter-TSN domains communication.
  - TSN domains and configuration models.

# An example – Fully centralized model



# An example – 2-level hierarchy

Q. How to model "Level 1" system with respect to "Level 2"?



# Generalized view – N-level hierarchy

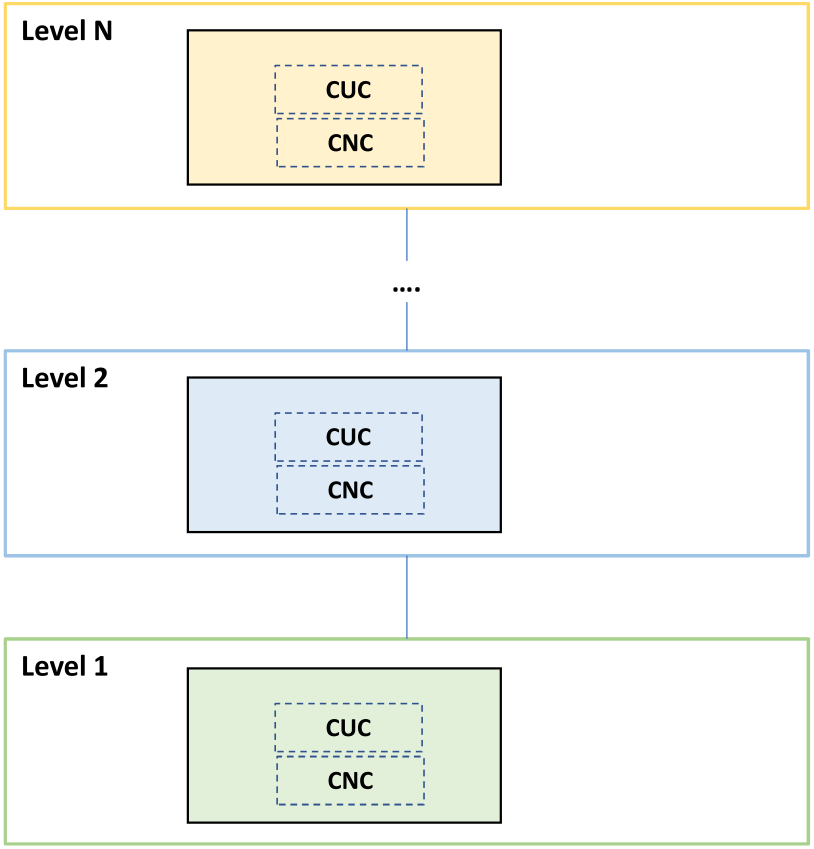
Problem:

- CUC/CNC
  - scope,
  - network view,
  - capabilities,
  - policies may varies between levels….

Aim:

- N-level system hierarchy is aiming for consistency and 1st level of interoperability (refer to [3] – Figure 3) in configuration and monitoring planes.
- Boundaries are logical in nature.
- Separation of concerns at each level.

Q. How to model “Level K-1” system with respect to “Level K”?



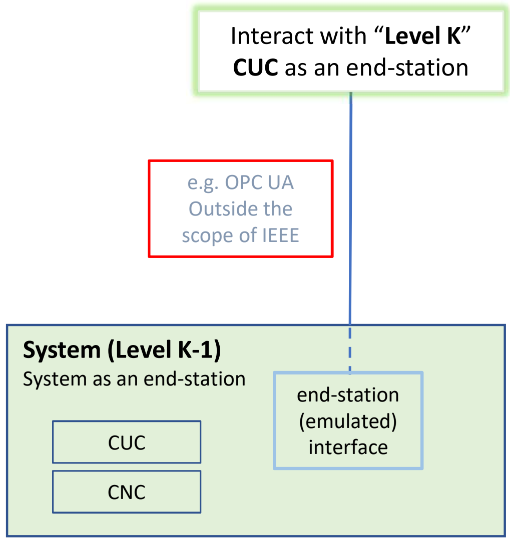
# How to model “Level K-1” system with respect to “Level K”?

- Depending on the system personality resides at “Level K-1”. System can be model as
  - An end-station [8] or
  - A bridged-end-station.

# System as an end-station (1)

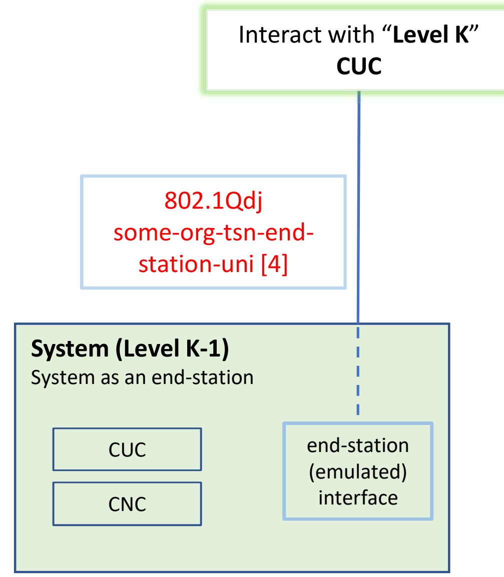
- System acts as a single monolithic entity.
- Options:
  - Model as an end-station with respect to “Level K” CUC (option 1 and 2).
  - Model as “Level K-1” CUC with respect to “Level K” CNC (option 3).

**Option 1:**




**Constraint:** System at “Level K-1” and CUC at “Level K” can interoperate only if they agree on the common user-to-user interface.

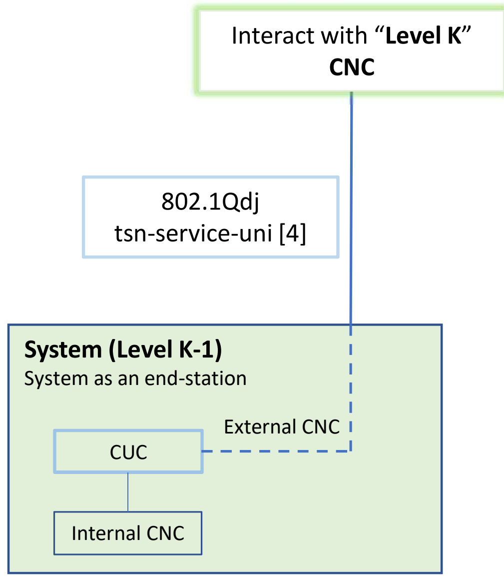
**Option 2:**




**Constraint:** System at “Level K-1” and CUC at “Level K” can interoperate only if they have common understanding of some-org-tsn-end-station-uni.

# System as an end-station (2)

**Option 3:**




- "Level K-1" CUC differentiate between internal and external CNC.

**Assumptions**:

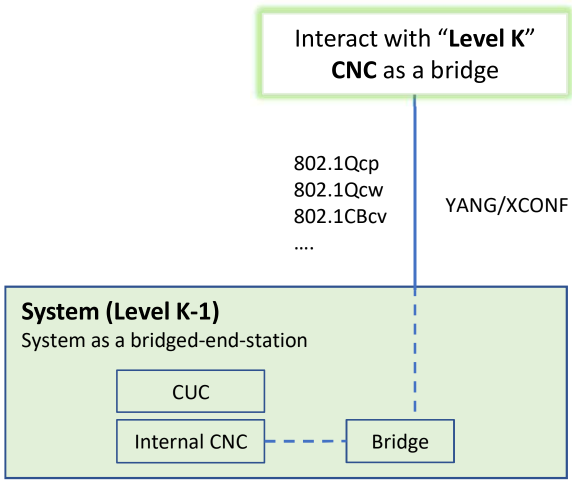
- CNC at "Level K" is capable of accepting requests from multiple CUC and resolve aspects like resource conflicts accordingly.
- 802.1Qdj working group will take care of YANG model - tsn-service-uni.

# System as a bridged-end-station (1)

XCONF → NETCONF/RESTCONF

- System acts as a single monolithic entity with respect to internal details but acts as a Bridge for pass-through traffic.
- Options for end-station entity:
  - List of "system as an end-station" plus ieee802-dot1q-tsn-end-station-uni [4].
- Options for Bridge entity:
  - System expose Bridge(s) directly via YANG/XCONF (option 1 and 2)
  - Constrained based exposure (option 3 and 4).

**Option 1 (single Bridge):**

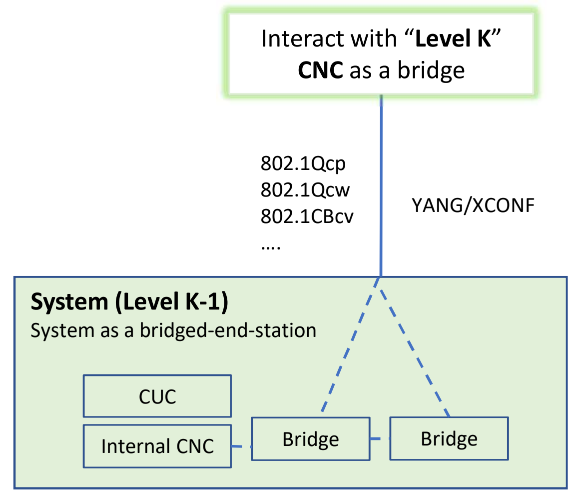



- Superimposed end-station models from the "system as an end-station" + ieee802-dot1q-tsn-end-station-uni to get the complete picture.

**Constraints:**

- Direct access to Bridge management interface may not be feasible.
- Bridge personality is control via internal and external CNC which may trigger such as resource ownership issues, conflicting configuration requirements …
- External CNC may invalidate certain assumption(s) made internally.

**Option 2 (Multiple Bridges):**




- Same as Option 1 (Single Bridge)

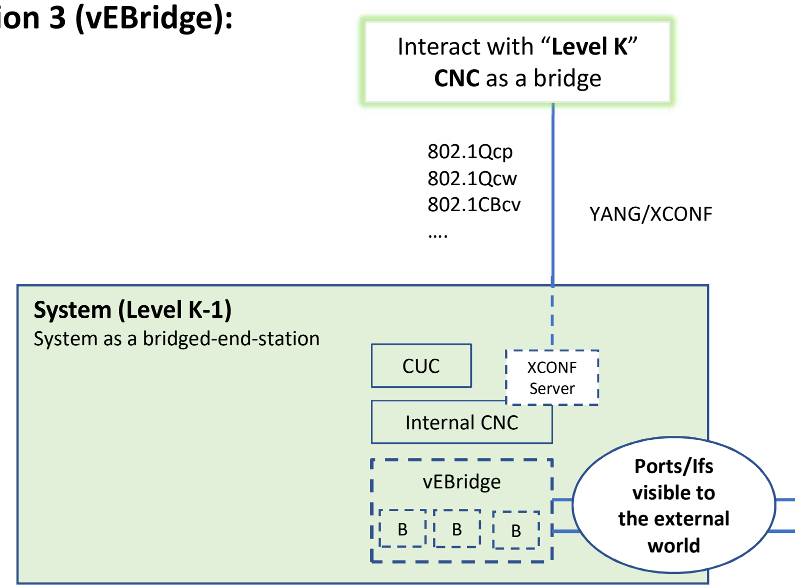
**Additional constraint:**

- Invalidate bridged-end-station abstraction due to the exposure of multiple internal bridges.

# System as a bridged-end-station (2)

vEBridge → virtual External Bridge

## Option 3 (vEBridge):




- Superimposed end-station models from the "system as an end-station" + ieee802-dot1q-tsn-end-station-uni to get the complete picture.

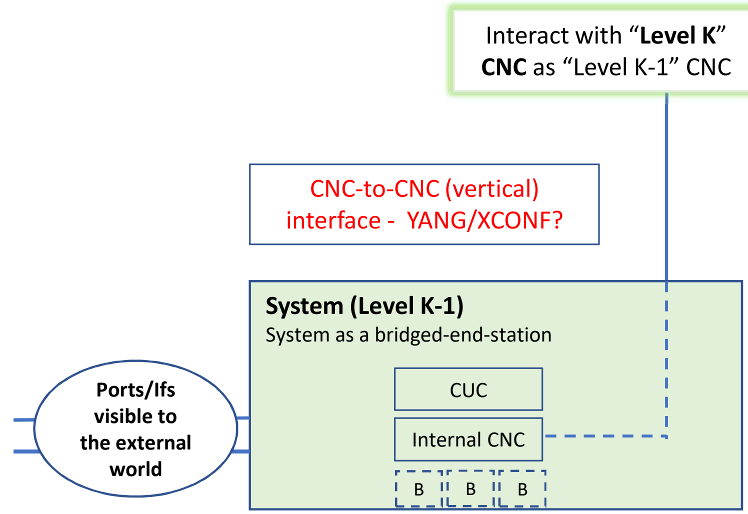
**Assumption:**

- Controlled Bridge(s) visibility via the concept of vEBridge.
- Internal CNC should implement the concept of vEBridge to avoid ownership issues.

**Constraints:**

- System internally manage the composition of vEBridge.
- Concept of vEBridge?
- YANG model for vEBridge?

## Option 4 (CNC-to-CNC):




- Superimposed end-station models from the "system as an end-station" + ieee802-dot1q-tsn-end-station-uni to get the complete picture.

**Assumption:**

- Internal CNC control the visibility of internal infrastructure via YANG model.

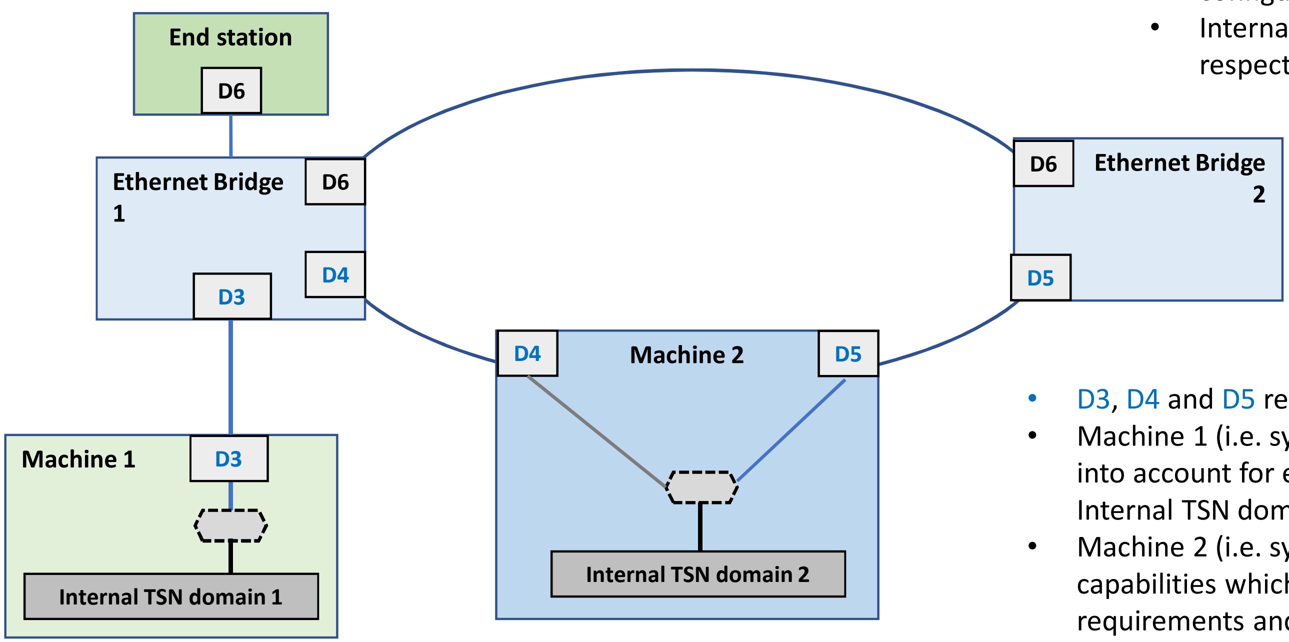
**Constraints:**

- CNC-to-CNC (vertical) interface [6] – YANG model/XCONF but also satisfy the above assumption.

Q. There is any existing IEEE standard which address the concept of vEBridge.
Q. There is any on-going work on defining an vertical interface between "Level K CNC" and "Level K-1 CNC" with controlled visibility of internal resources.

# TSN Adapter Domain

- TSN Adapter domain [12] is point-to-point in nature (logically).
- TSN Adapter domain basic purpose is to facilitate inter-TSN domains communication.
- It provides a tool to encapsulation an internal TSN domain(s) and concisely presents the interworking capabilities.



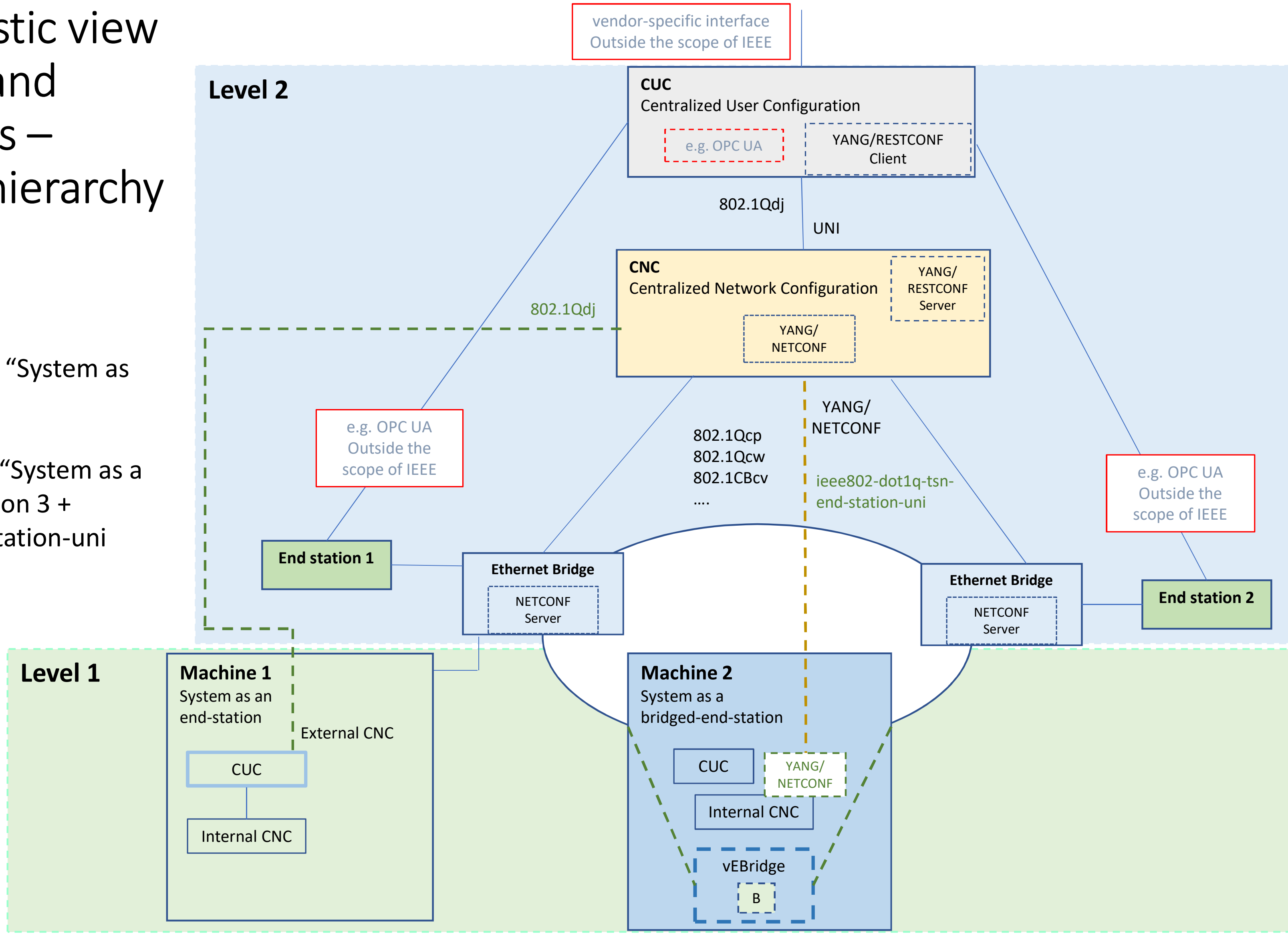

- Enhanced system composability (i.e. separation of concern per domain).
- It is possible to assign multiple TSN Adapter domains on a single physical point-to-point link as long as this link satisfy the individual TSN Adapter domain requirement(s).
- TSN Adapter domain re-emphasize the role of policy driven configuration [16].
- Internal (private) TSN domain(s) are black boxes [10] [15] with respect to the outside world.

- D3, D4 and D5 represents TSN Adapter domains.
- Machine 1 (i.e. system as an end-station) advertise D3 capabilities which take into account for example, the application requirements and the constraints of Internal TSN domain 1.
- Machine 2 (i.e. system as an bridged-end-station) advertise D4 & D5 capabilities which take into account for example, the application requirements and the constraints of Internal TSN domain 2.
- It should be possible in a system to merge TSN Adapter domains. For example, D4, D5 and D6 are mergeable, if their requirements and capabilities are compatible.

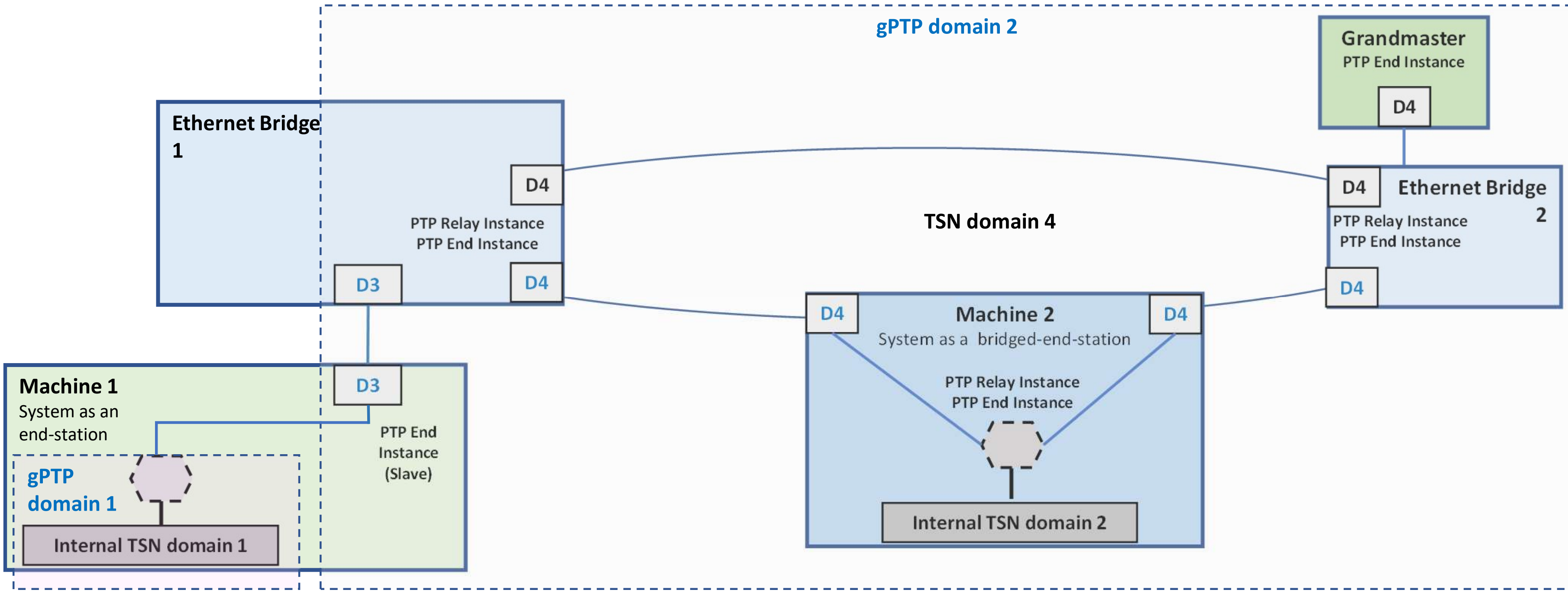
# Application examples

# An example: Holistic view of configuration and monitoring planes – Updated 2-level hierarchy

- Machine 1 is modelled as “System as an end-station” option 3.
- Machine 2 is modeled as “System as a bridged-end-station” option 3 + ieee802-dot1q-tsn-end-station-uni

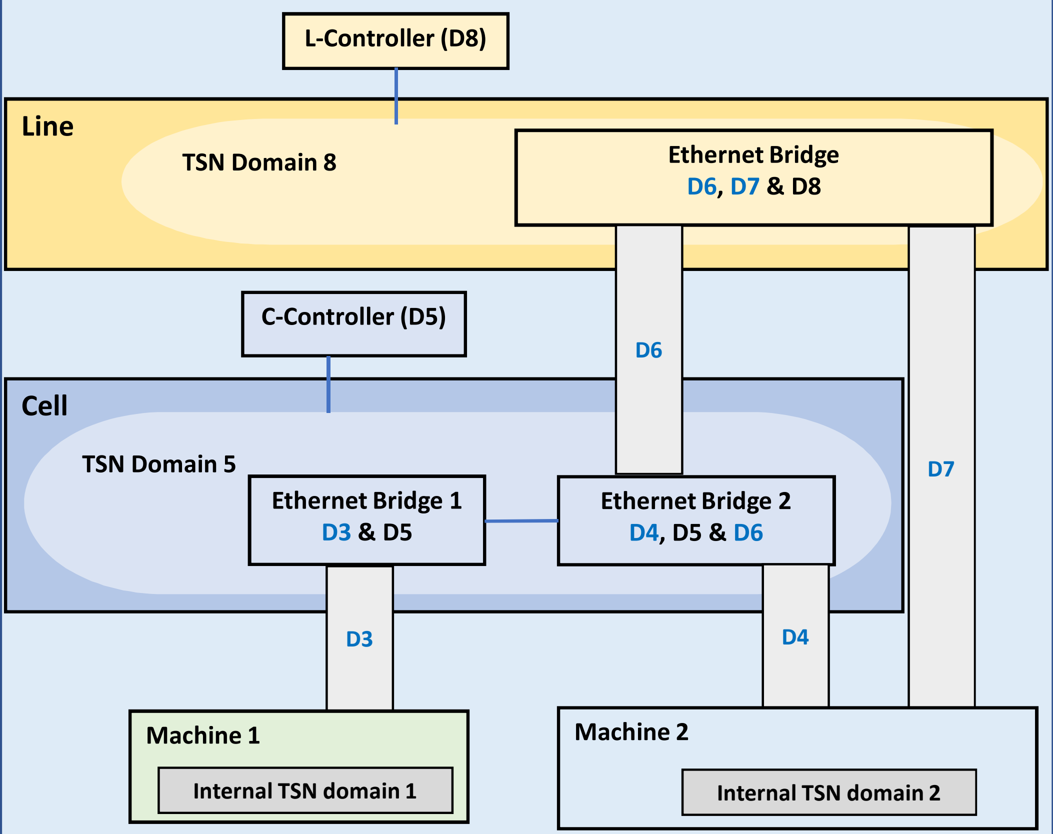


# An example – TSN domain vs gPTP domain

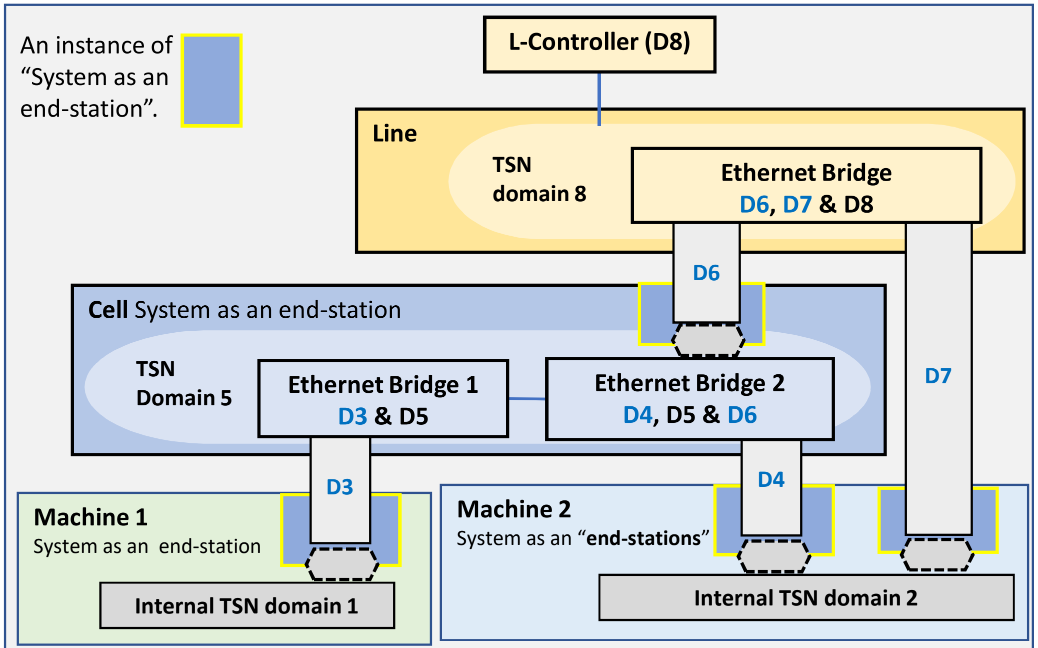



- The extent of a gPTP domain is either greater than or equal to a TSN domain [13].
- TSN domain 2, 3 and 4 are using the timing information from gPTP domain 2.
- Externally observable behavior of “system as an end-station” or “system as bridged-end-station” running gPTP is the same as any other compliant end-station or bridged-end-station.
- Bridges will need a gPTP End Instance besides PTP Relay Instance to support features e.g. 802.1Qbv.
- Machine 2 internally relay the timing information from gPTP domain 2 to TSN domain 2 (black box).
- TSN Adapter domain (D3) is used to highlight a mismatch in requirements/capabilities which necessitate the use of dedicated gPTP domain, e.g. Machine 1 internal timing functionality is non-modifiable except external interface.

# An example – Composing inter-TSN domains communication

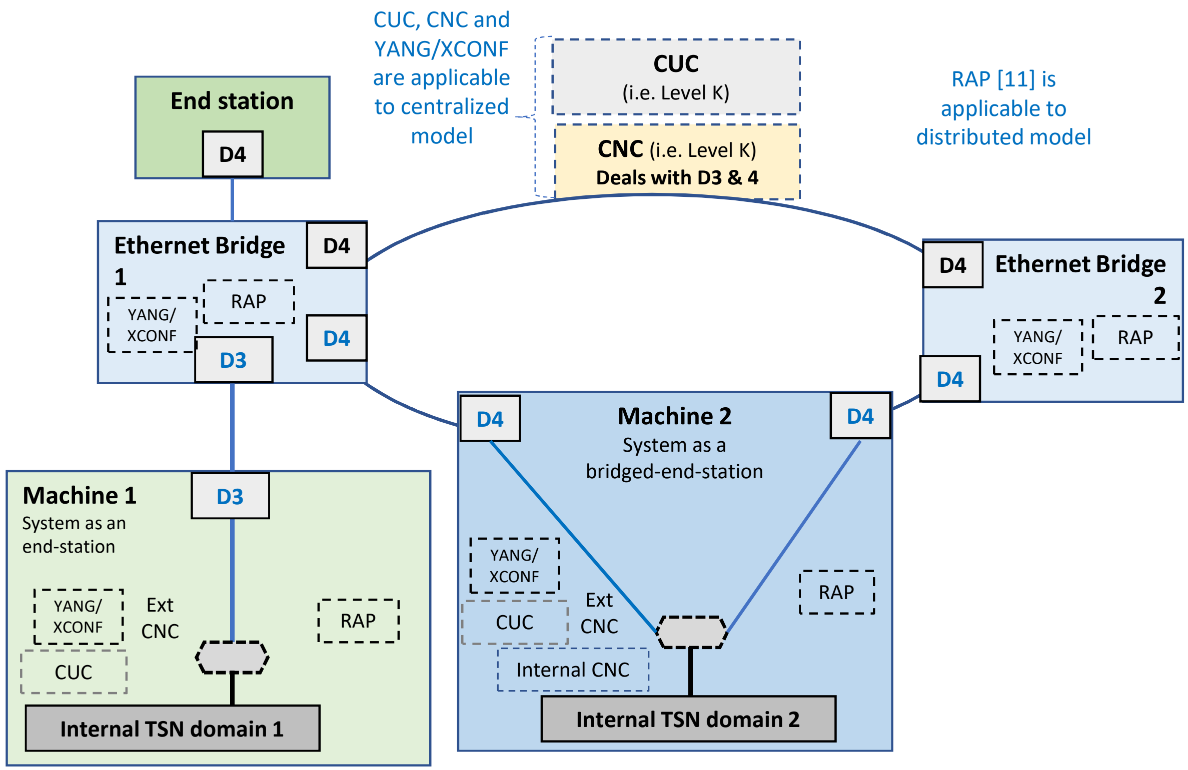



- All inter-TSN domains communication are via Adapter TSN domain(s).




- Machine 1, Machine 2 and Cell when acting as a "System as an end-station" advertise their system capabilities to higher level based on the application requirements and the constraints of Internal infrastructure.
- System will pass the first level of verification, if D3, D4, D6 and D7 satisfy the system requirements.
- Level violation (Line to Machine 2) is intentional to demonstrate different views.

# An example – TSN domains and configuration models




- Internal (private) TSN domain(s) and its configuration model(s) are black boxes [10] [15] with respect to the outside world.
- The diagram assumes that
  - requirements/capabilities advertise by Machine 2 are compatible with TSN domain 4.
  - CNC at Level K can deal with multiple TSN domains (i.e. D3 and D4).

- Centralized Model: If higher level network (i.e. Level K) is based on fully centralized model, then the YANG/XCONF can be used to communicate end-station/bridged-end-station personality to “Level K” CUC and CNC.

- Distributed Model: If higher level network (i.e. Level K) is based on fully distributed model (i.e. no CUC/CNC at level K), then the protocol like RAP [11] can be used (in principle as is) to communicate end-station/bridged-end-station personality to “Level K” entities (i.e. Ethernet Bridge, end-station).

Q. RAP – streams handling in the presence of multiple TSN domains (i.e. D3 and D4)?

# Conclusions

1. “System as an end-station” recommendation: Option 3.
2. “System as a bridged-end-station” recommendations: Option 4.
   - Bridge entity: CNC-to-CNC (vertical) interface.
   - End-station entity: “System as an end-station” option 3 or ieee802-dot1q-tsn-end-station-uni [4].

Gaps:

- Realization of an entity [8] [14] which presents whole system as an end-station or a bridged-end-station. Compatibility is required for externally observable behavior only.
- CNC-to-CNC (vertical) interface [6] and corresponding YANG data model with controlled visibility of internal resources.
- Concept of vEBridge and corresponding YANG data model.

Notes:

- P802.1Qdj [5] (limited info - work in progress).
- ieee802-dot1q-tsn-end-station-uni [4] (limited info - work in progress).
- P802.1Qdd [11] (work in progress).

Thank you

# References


1. IEEE802.1Qcc-2018
2. http://www.ieee802.org/1/files/private/60802-drafts/d1/60802-d1-1.pdf
3. http://www.ieee802.org/1/files/public/docs2018/60802-industrial-use-cases-0918-v13.pdf
4. http://www.ieee802.org/1/files/public/docs2019/new-kehrer-TSN-Configuration-Enhancements-0319-v01.pdf
5. https://1.ieee802.org/tsn/802-1qdj/
6. http://www.ieee802.org/1/files/public/docs2018/60802-stanica-convergence-coexistence-0718-v03.pptx
7. http://www.ieee802.org/1/files/public/docs2018/60802-woods-QccConfig-0918-v01.pdf
8. http://www.ieee802.org/1/files/public/docs2019/60802-Hantel-TSN-IA-Domains-Constructability-0519-v01.pdf
9. http://www.ieee802.org/1/files/public/docs2019/60802-chen-TSN-management-0119-v00.pdf
10. http://www.ieee802.org/1/files/public/docs2020/new-Steindl-TSN-inter-domain-communication-0120-v4.pdf
11. https://1.ieee802.org/tsn/802-1qdd/
12. http://www.ieee802.org/1/files/public/docs2018/60802-Steindl-ConfigurationModelAlignment-0918-v02.pdf
13. http://www.ieee802.org/1/files/public/docs2018/60802-Hantel-TSN-Interdomain-Communications-0718.pdf
14. http://www.ieee802.org/1/files/public/docs2019/new-chen-TSN-Configuration-Interaction-0719-v01.pdf
15. http://www.ieee802.org/1/files/public/docs2018/60802-harima-industrial-use-case-0518-v04.pdf
16. http://www.ieee802.org/1/files/public/docs2018/60802-Hantel-Zuponcic-Policy-Based-Configuration-0518-v01.pdf