TSN Configuration Interaction

Lihao Chen
Huawei Technologies

HUAWEI

Purpose

- Summarize existing considerations
  - How to configure inter-TSN-domain communications?
  - How do multiple CNCs work together?
  - How do multiple TSN configuration models coexist?
- Propose an early-stage outline of a non-vendor-specific solution idea
  - Specify a new Protocol that uses LRP to carry information between TSN configuration entities (e.g., CNC entities).
- Non-purpose
  - Discuss what would be the definition of a TSN domain.
  - Discuss what enhancements are needed for the User-Network Interface.
  - Discuss how a TSN configuration entity specifically configures a bridge.

Current TSN configuration models

- The distributed model doesn't need a CNC.

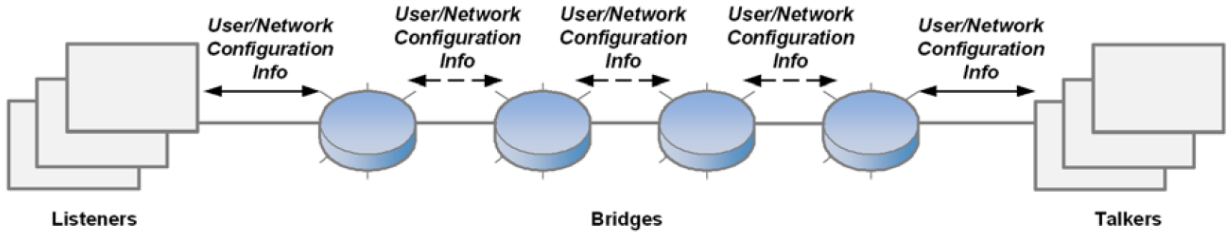


Figure 46-1 — Fully Distributed Model

- The hybrid model and centralized model work with one CNC.

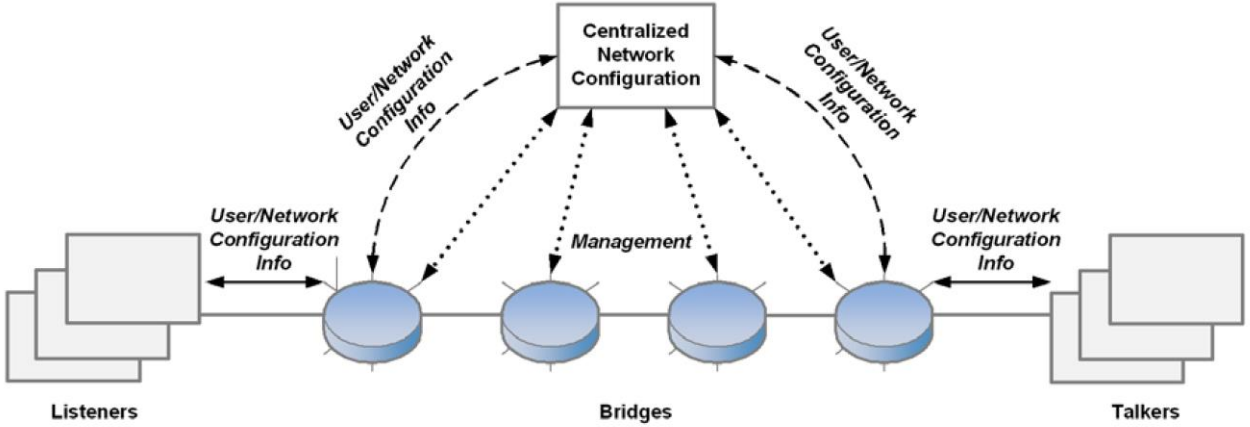


Figure 46-2 — Centralized Network / Distributed User Model

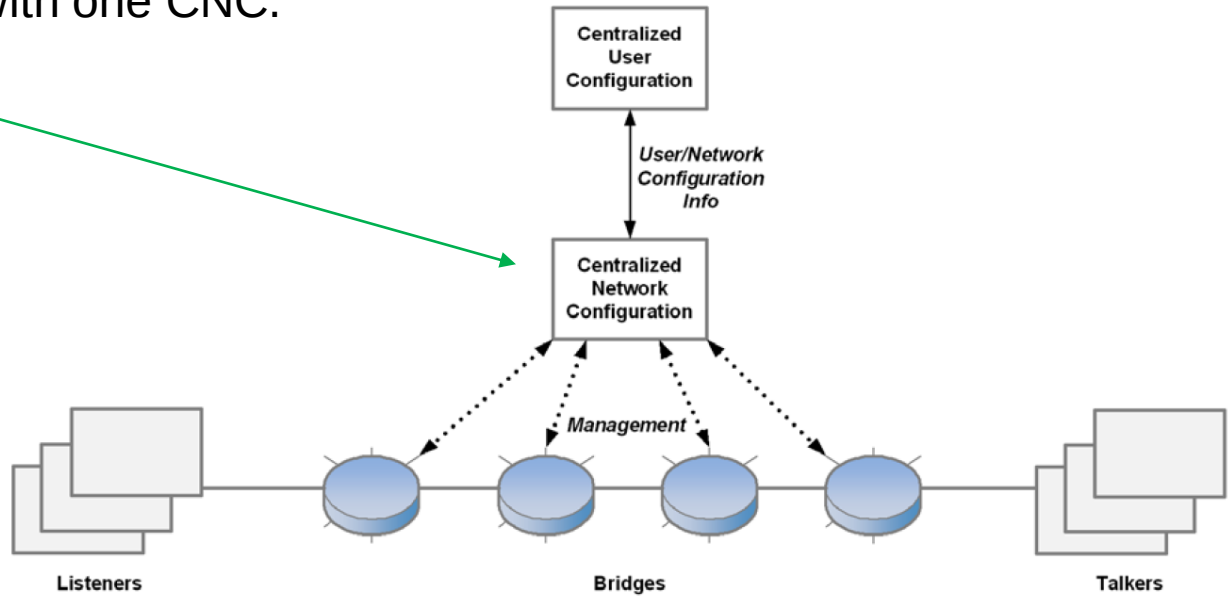


Figure 46-3 — Fully Centralized Model

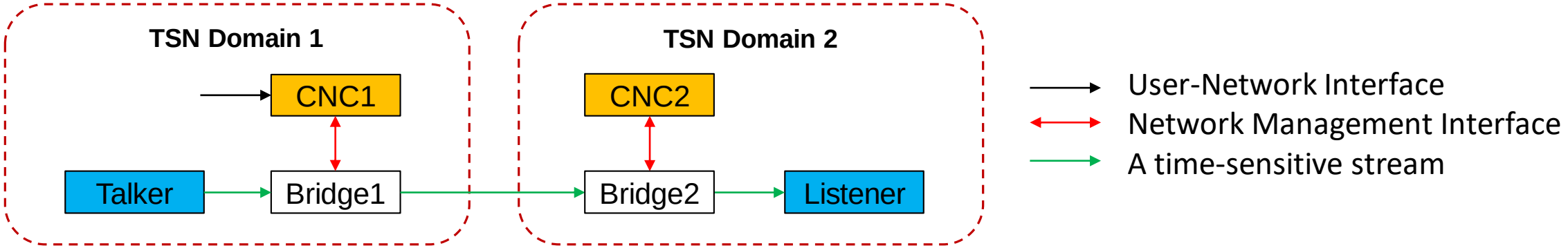
References of existing considerations

- http://www.ieee802.org/1/files/public/docs2017/new-Hantel-AnIndustrialApproachToTSN-0317-v02.pdf
- http://www.ieee802.org/1/files/public/docs2018/60802-Hantel-TSN-Interdomain-Communications-0718.pdf
- http://www.ieee802.org/1/files/public/docs2018/60802-stanica-convergence-coexistence-0718-v03.pptx
- http://www.ieee802.org/1/files/public/docs2018/60802-Steindl-InterTSNDomain-0818-v02.pdf
- http://www.ieee802.org/1/files/public/docs2018/60802-Steindl-ConfigurationModelAlignment-0918-v02.pdf
- http://www.ieee802.org/1/files/public/docs2019/60802-chen-TSN-management-0119-v00.pdf
- http://www.ieee802.org/1/files/public/docs2019/60802-Steindl-ExampleSelections-0119-v02.pdf

- 3 questions were raised in these materials: (See Page 5-7)
  - How to configure interdomain communications?
  - How do multiple CNCs work together?
  - How do multiple TSN configuration models coexist?

How to configure interdomain communications?

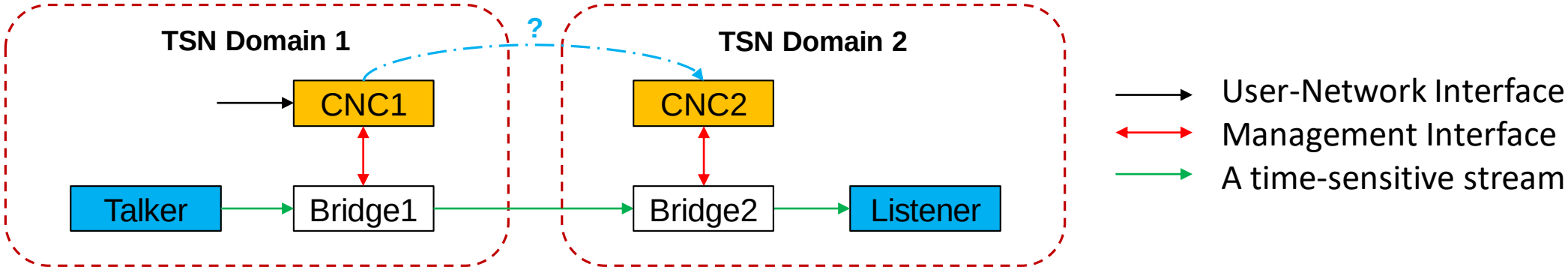
- A method for configuration of time-sensitive streams over multiple TSN Domains is needed.
- An example: (suppose the hybrid or centralized configuration model is used)
  - The requirement of establishing a new time-sensitive stream between the Talker and the Listener is sent to CNC1 by whatever User-Network Interfaces.
  - Network management Interfaces exist between CNC1 and Bridge1, CNC2 and Bridge2.
  - Question: How to properly configure bridges, especially Bridge 2?



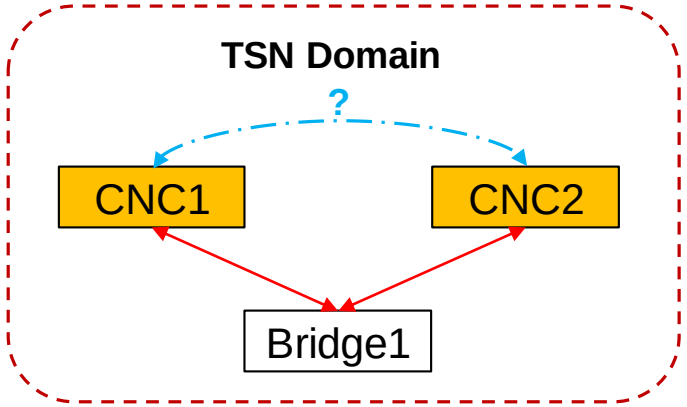
See page 13 and 14 for a early-stage solution idea.

How do multiple CNCs work together?

- CNC1 may need to tell something to CNC2. How?



- What if there are multiple CNCs within one domain?
  - How to ensure that only one CNC is responsible for configuration this TSN Domain at one time?
  - How to ensure that each CNC is able to configure the whole TSN domain in case of failover?



See page 15 for a early-stage solution idea.

How do multiple TSN configuration models coexist?

- The reason we need them to coexist mainly comes from the need for interdomain communication.

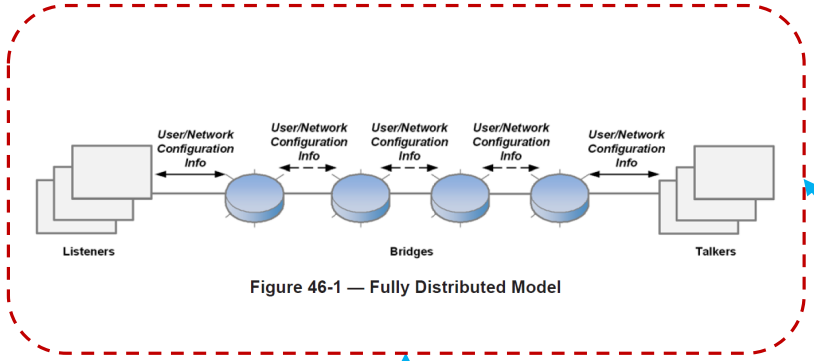


Figure 46-1 — Fully Distributed Model

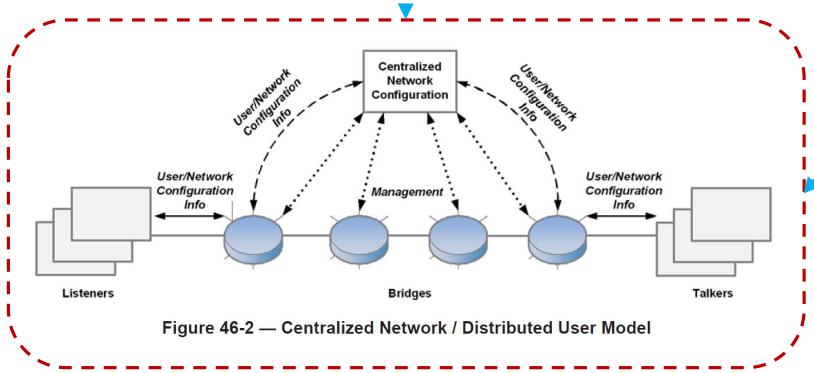


Figure 46-2 — Centralized Network / Distributed User Model

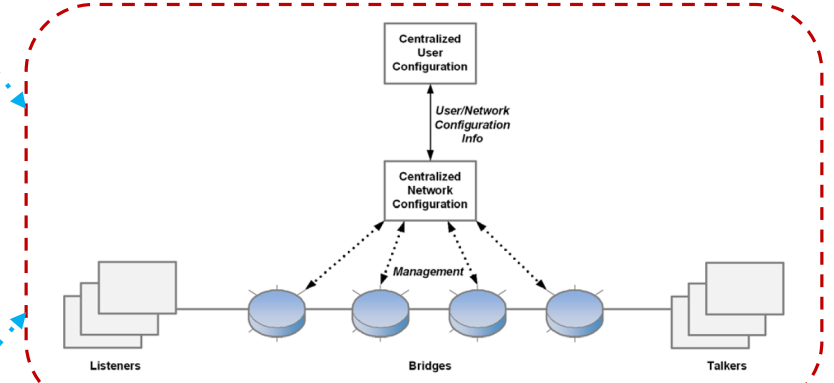


Figure 46-3 — Fully Centralized Model

See page 16 for a early-stage solution idea.

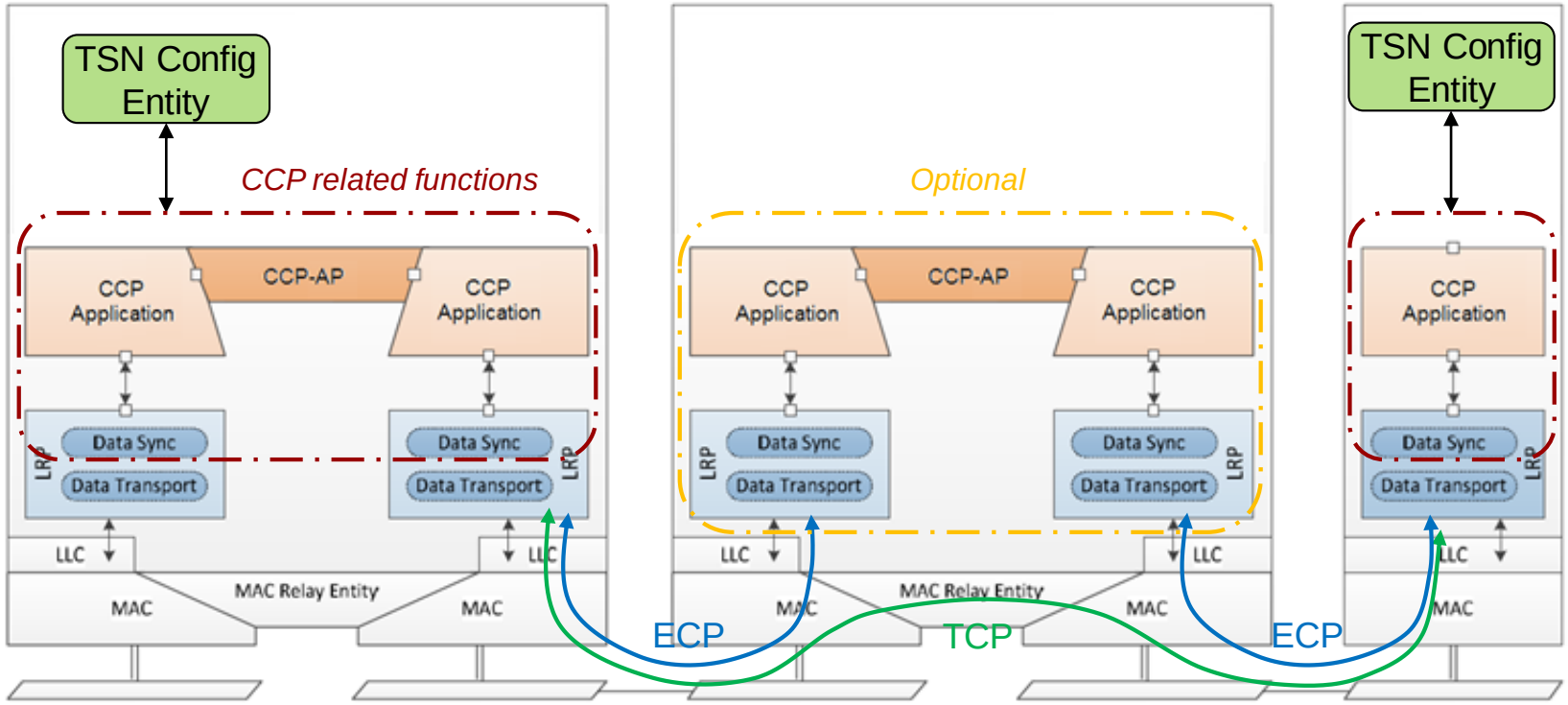
Theoretically, almost everything can be manually configured.

However, this is definitely not what we want.

CCP proposal

CCP, either Config-entity to Config-entity Protocol or CNC to CNC Protocol, is just a tentative name from the author.

- CCP could be used to carry information between TSN configuration entities (e.g., CNC controllers).
- Need to specify protocols, procedures, and managed objects for this CCP that uses LRP.



A bridge with TSN Config Entity embedded | A bridge without TSN Config Entity | A TSN Config end-station

LRP is designed to facilitate the creation of **LRP application** protocols (e.g., RAP, and potentially CCP) that distribute information through all or part of a network.

CCP defines a per-port CCP Application component and a per-bridge CCP Attribute Propagation (CCP-AP) component.

This figure is taken from http://www.ieee802.org/1/files/public/docs2017/tsn-chen-RAP-whitepaper-1117-v02.pdf and has been modified.

CCP - mode of operation

CCP+ indicates optional functions of CCP. Whether CCP+ is need or not is TBD (supposing that CCP is needed).

- To transmit information, Config-entities / CNCs need to 'know' each other.
  - Where are you?
  - Which TSN domain do you belong to?
  - Are you currently in charge?
  - Etc.

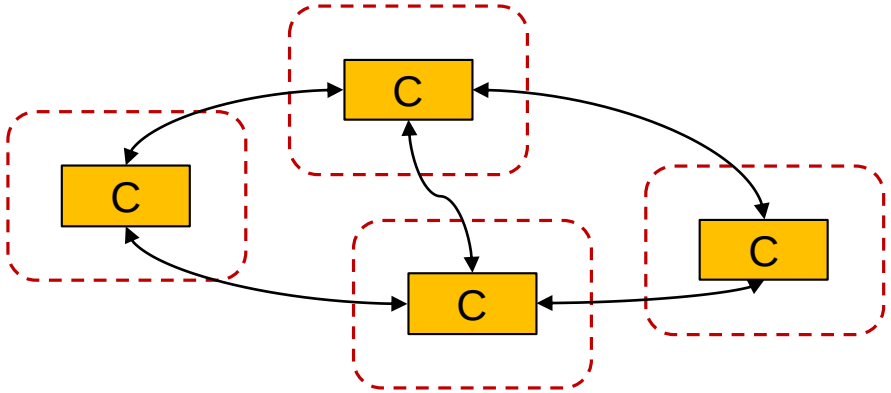
  } New attributes are needed.

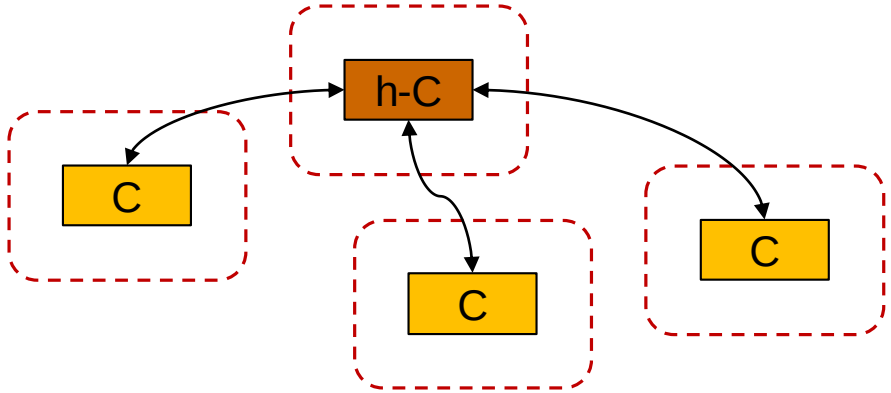
- Three options:
  - Manual configuration, whenever a new Controller / Config-entity is added into the network.
  - Auto-detection by a CCP+ function, using ECP to propagate specific information (CCP attributes) hop by hop in the network. Need to design this CCP+ function.
  - Auto-detection by other mechanisms, e.g., LLDP with proper enhancements.

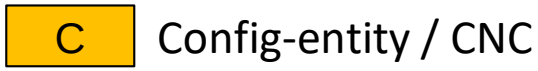
CCP - mode of operation

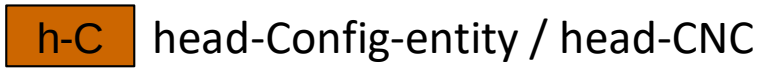
CCP, either Config-entity to Config-entity Protocol or CNC to CNC Protocol, is just a tentative name from the author.

- CCP may have two modes for transmitting interdomain information:
  - Peer-to-Peer
  - Hierarchy





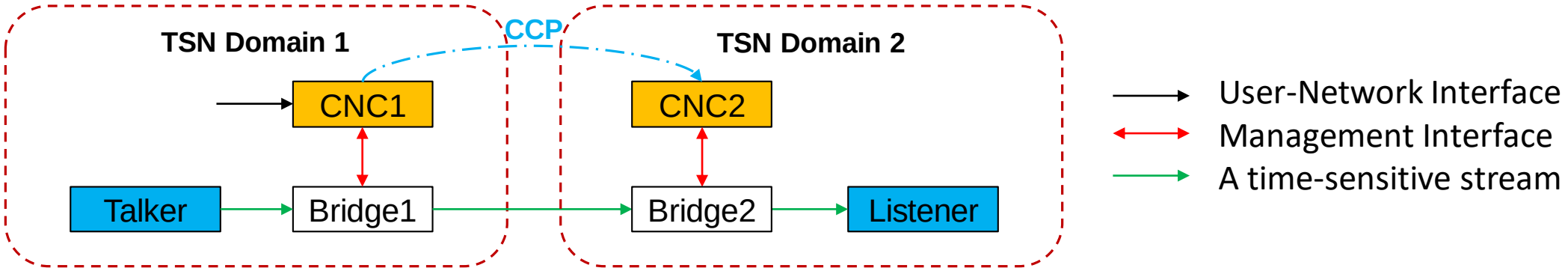




Now let’s get back to the 3 questions and see how CCP can help.

How to configure interdomain communications?

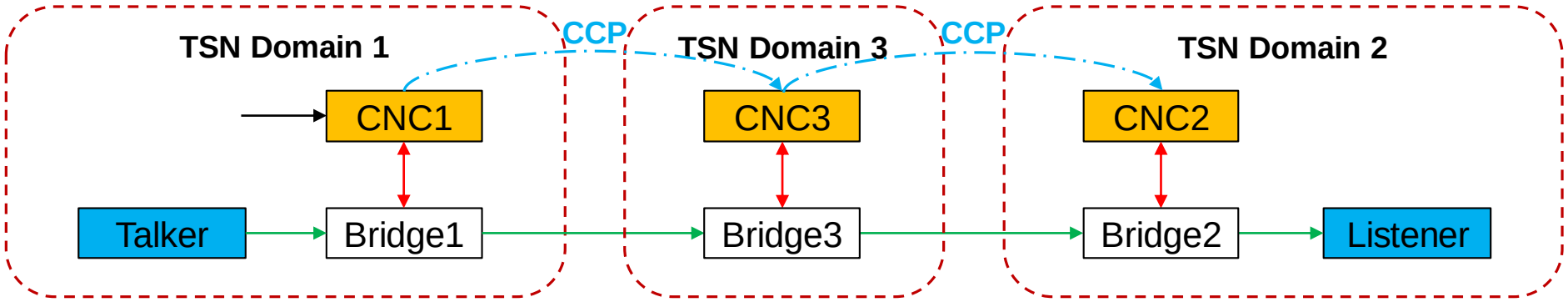
- More specifically, how to properly configure Bridge 2?
- Intuitive Answer:
  - As CNC2 is responsible to configure Bridge2, CNC1 could use CCP to tell CNC2 the **information** about the new TSN stream to be established.
  - It is the author's perspective that this **information** can include TSN UNI data, e.g. end station capabilities and requirements (some has already been specified in Qcc and some will be specified in future 802.1 projects), and should not include any data from non-802.1-standardized data models.



How to configure interdomain communications?

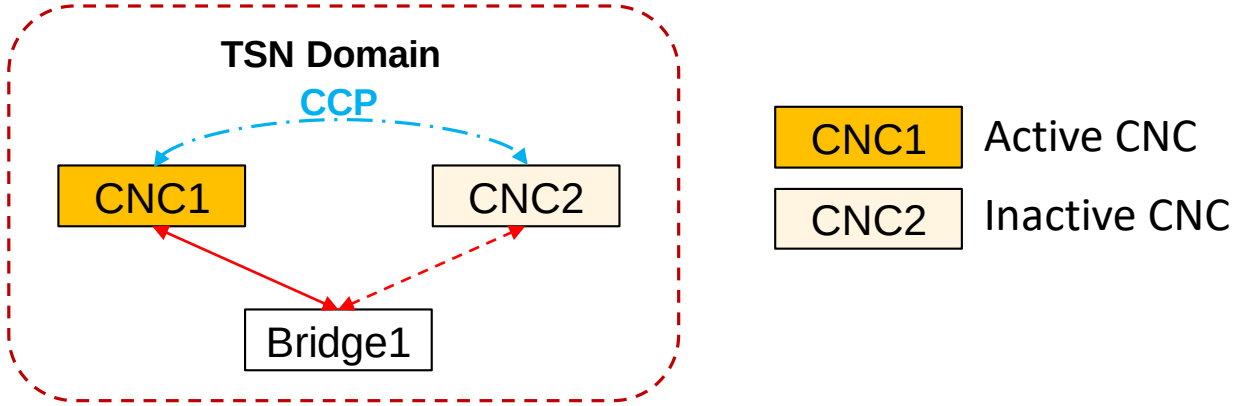
- What's more, how does CNC1 know that the **information** should be sent to CNC2?

  And what if there are more TSN Domains on the path from Talker to Listener?
- Intuitive Answer:
  - Peer-to-Peer mode: CNC1 does not know. Just send the **information** to its neighboring CNCs, and so on.
  - Hierarchy mode: CNC1 sends the **information** to the head-CNC, and let the head-CNC decide.



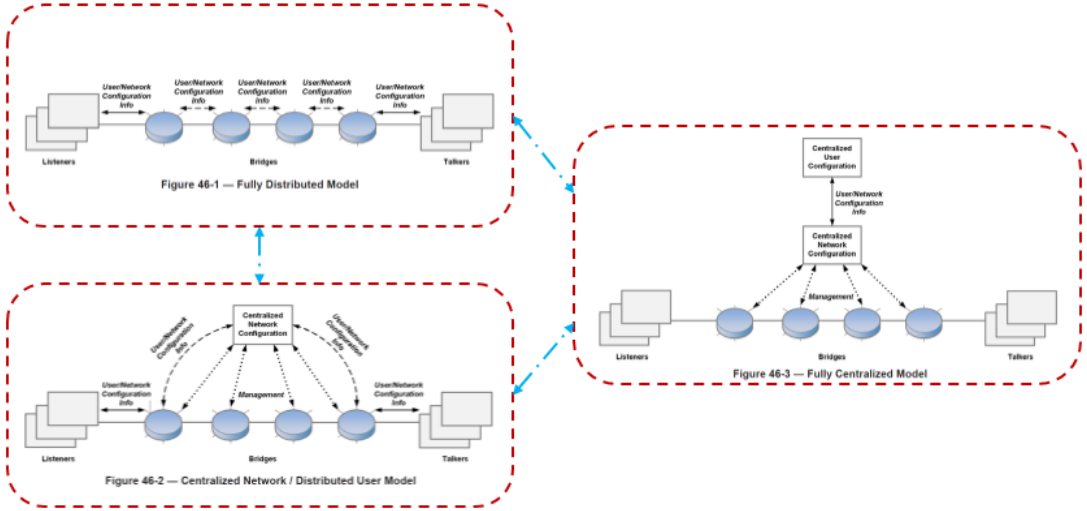
How do multiple CNCs work together?

- More specifically, how to ensure that only one CNC is responsible for configuration at one time while other CNCs are able to configure the whole TSN domain in case of failover
- Intuitive Answer:
  - Using CCP for **information** synchronization between CNCs in one TSN Domain. Again, it is the author's perspective that this **information** can include TSN UNI data, maybe some network management related data as well, and should not include any data from non-802.1-standardized data models.
  - Using CCP+ for active-CNC selection, declaration, and failover. A LLDP-based method may work as well.



How do multiple TSN configuration models coexist?

- For hybrid/centralized models, as there are CNCs, the issue is the same as in Page 13 and 14.
- Question: How does a TSN domain using distributed model cooperate with other TSN domains?
- The author does not have an ideal answer yet. Maybe, a **functional entity** is needed to gather all useful MSRP/RAP attributes from devices in this TSN domain which uses distributed model, then this **functional entity** can perform CCP just like a CNC do? (However, we might be changing a distributed model into a hybrid model.)



Next steps

- Does this CCP (or something similar) need to be standardized in TSN TG?
  - Need more discussion and feedback.
  - Related discussions, e.g., Mark Hantel's proposal on TSN-IA Domains and constructability, could also help us to better figure out this question.

    http://www.ieee802.org/1/files/public/docs2019/60802-Hantel-TSN-IA-Domains-Constructability-0519-v01.pdf
  - If yes, we can explore further details such as specific parameters and procedures of CCP.

Thank you