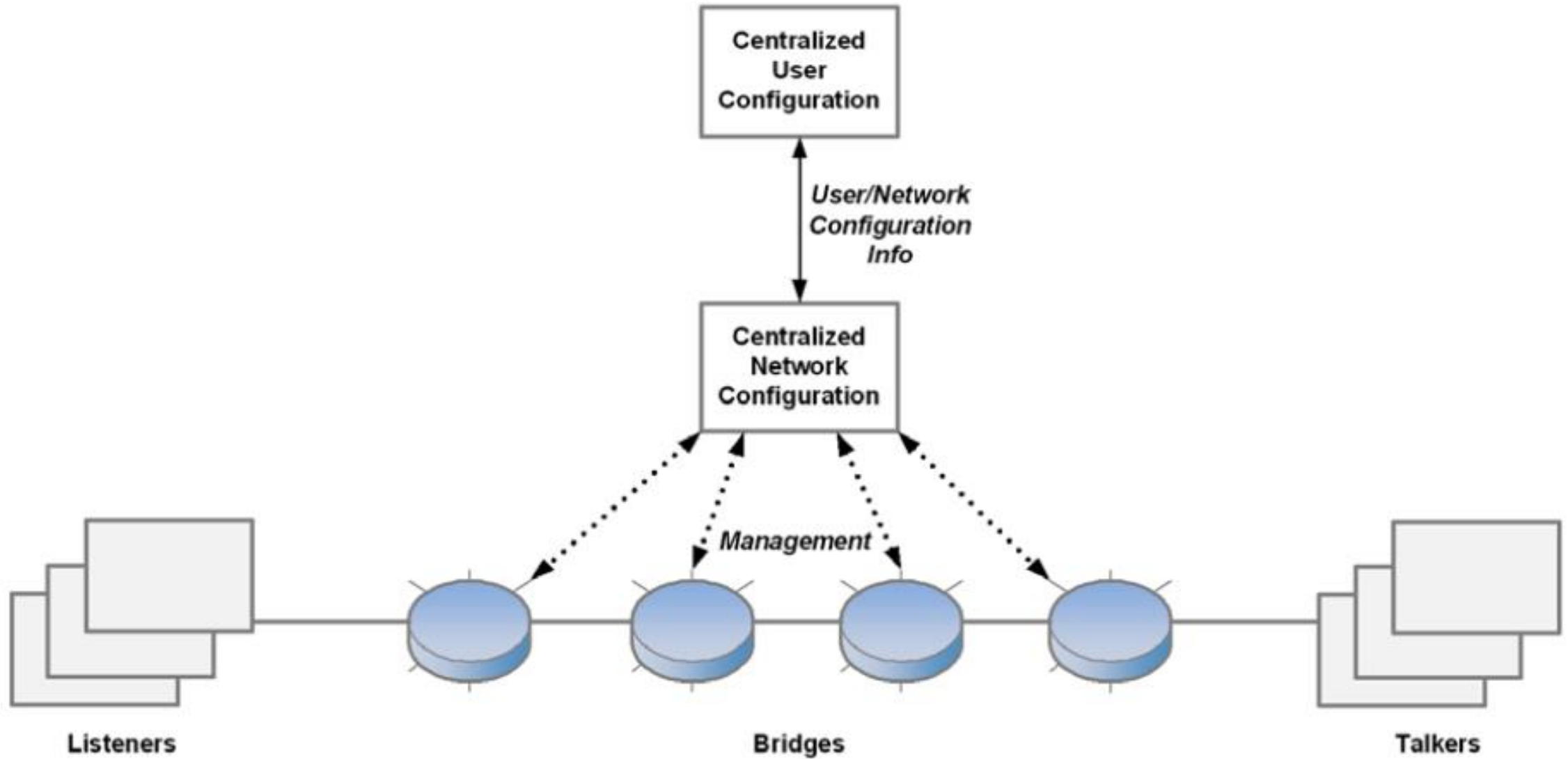


PTCC - TSN Endstation Centralized Configuration Interface for OPC UA PubSub Systems

Astrit Ademaj

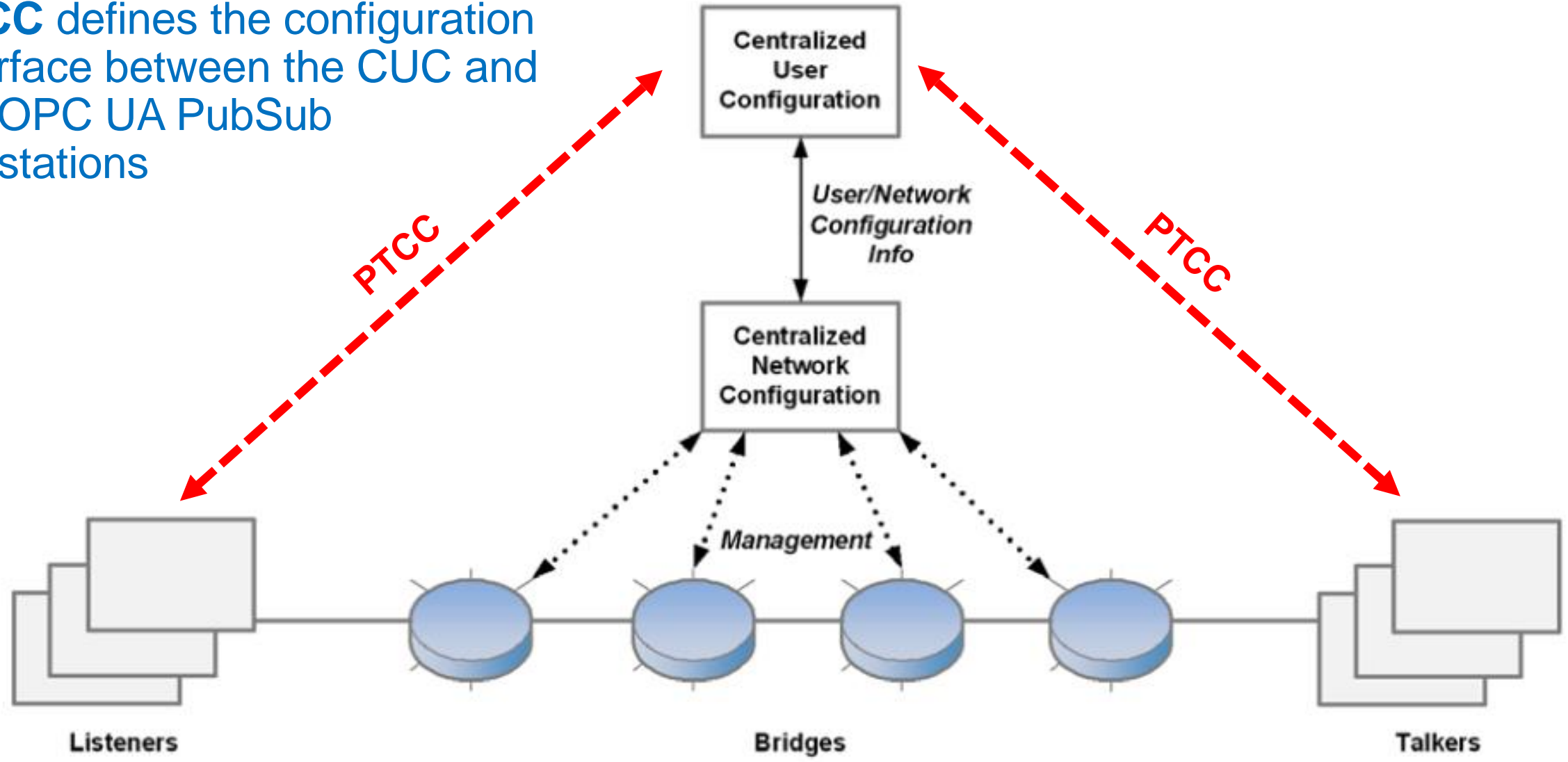
astrit.ademaj@ttech-industrial.com

TSN Fully Centralized Configuration Model



PubSub TSN Centralized Configuration (PTCC)

PTCC defines the configuration interface between the CUC and the OPC UA PubSub endstations



- **Specification Work within the UA TSN WG since March 2018**
- **Draft Stage**

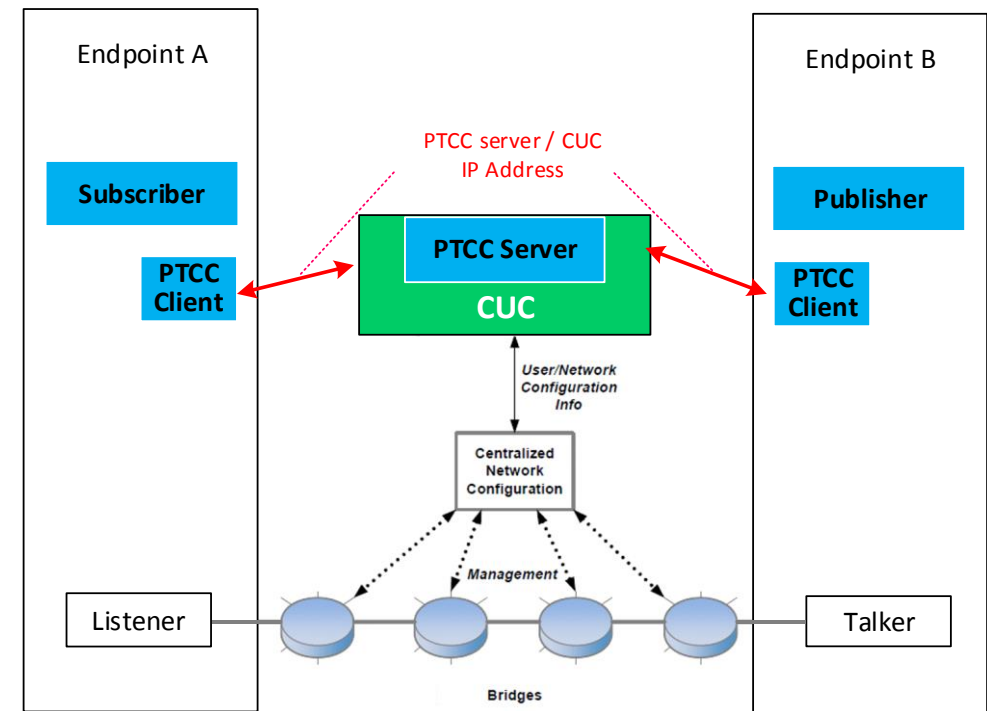
PTCC handles the stream configuration for TSN/Ethernet networks

PTCC is part of the CUC implementation for PubSub TSN based systems using the TSN centralized configuration model

PTCC configuration interface solution with a client-server type of operation

PTCC-client on the PubSub endstations.

PTCC-server can be implemented in one device in the network. PTCC-server is the part of the CUC - communication interface to Publishers and Subscribers



PTCC draft specification covers:

Concept/principle of operation

Different operation modes

- Request mode
- Push mode

StreamObject State machine

- PTCC-client Publisher
- PTCC-client Subscriber
- PTCC-server

Frame Mapping

- Extended UADP frame format
- Ongoing work on usage of the OPC UA client server communication for the PTCC

PTCC specification defines some requirements for the CNC operation

PTCC draft specification does not cover the Interface between the CUC and the CNC

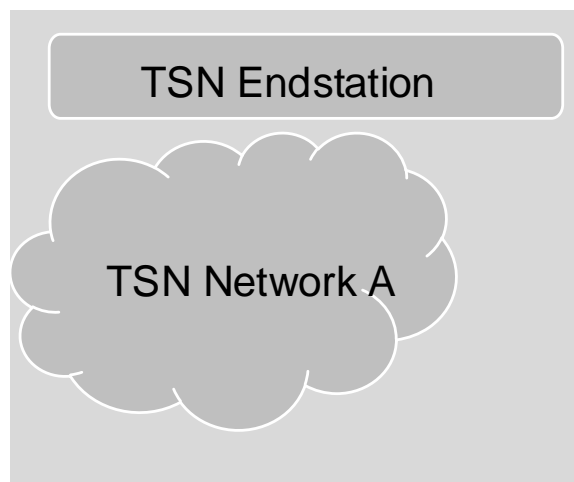
This is covered (going to be covered) by the IEEE TSN WG and is out of the scope of the PTCC specification

- StreamName is essential for the establishing the connections in the ad-hoc operation

Application

StreamName,
LatencyReq,...

- StreamName is application specific (and configured by the application configuration mechanism)



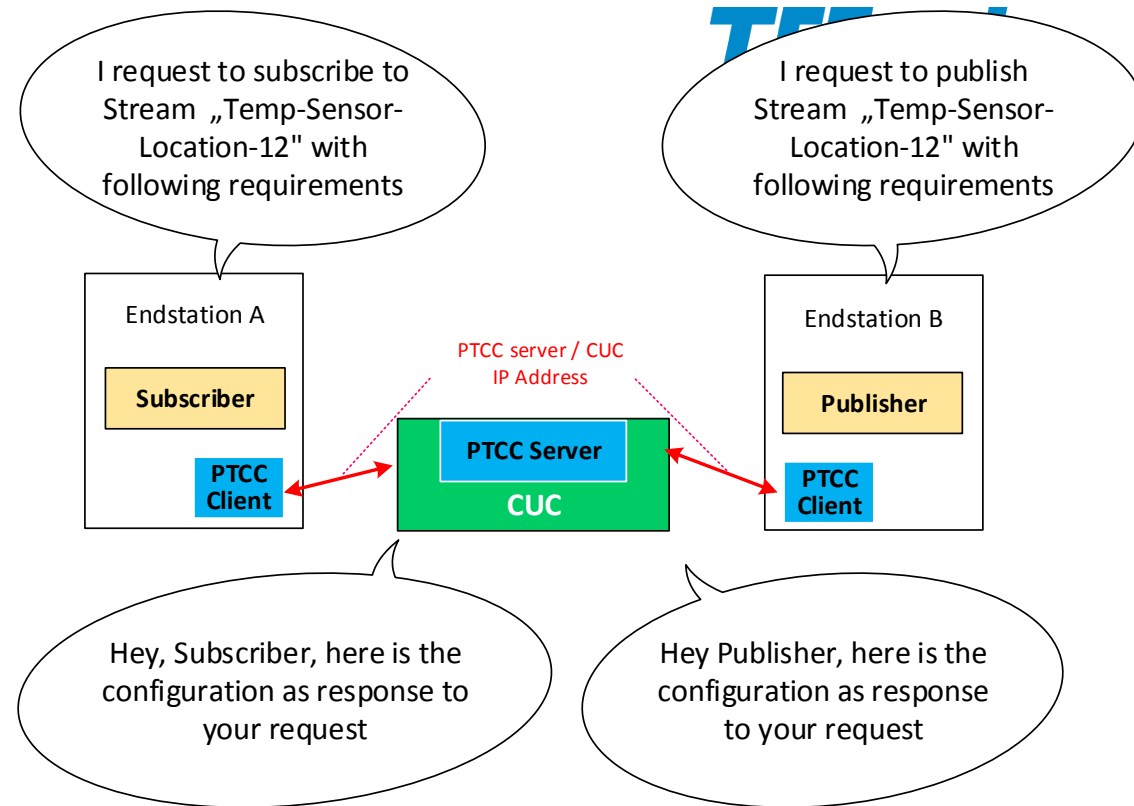
StreamID,
DMAC, VID,...
MaxLatency,...

- StreamID is generated by the network
- StreamName is used at the PTCC-server to link the requests from a publisher and subscribers related to a specific stream
- Under the assumption that one CUC is responsible for one TSN domain*, it must be ensured that the StreamNames are unique within one TSN domain, whereas TSN domain identifies the endstations that are managed by one CUC

*TSN domain is not fully defined yet

PTCC (Client) operational modes

- **Request Mode**
 - Configuration requests are initiated by the PTCC-Clients



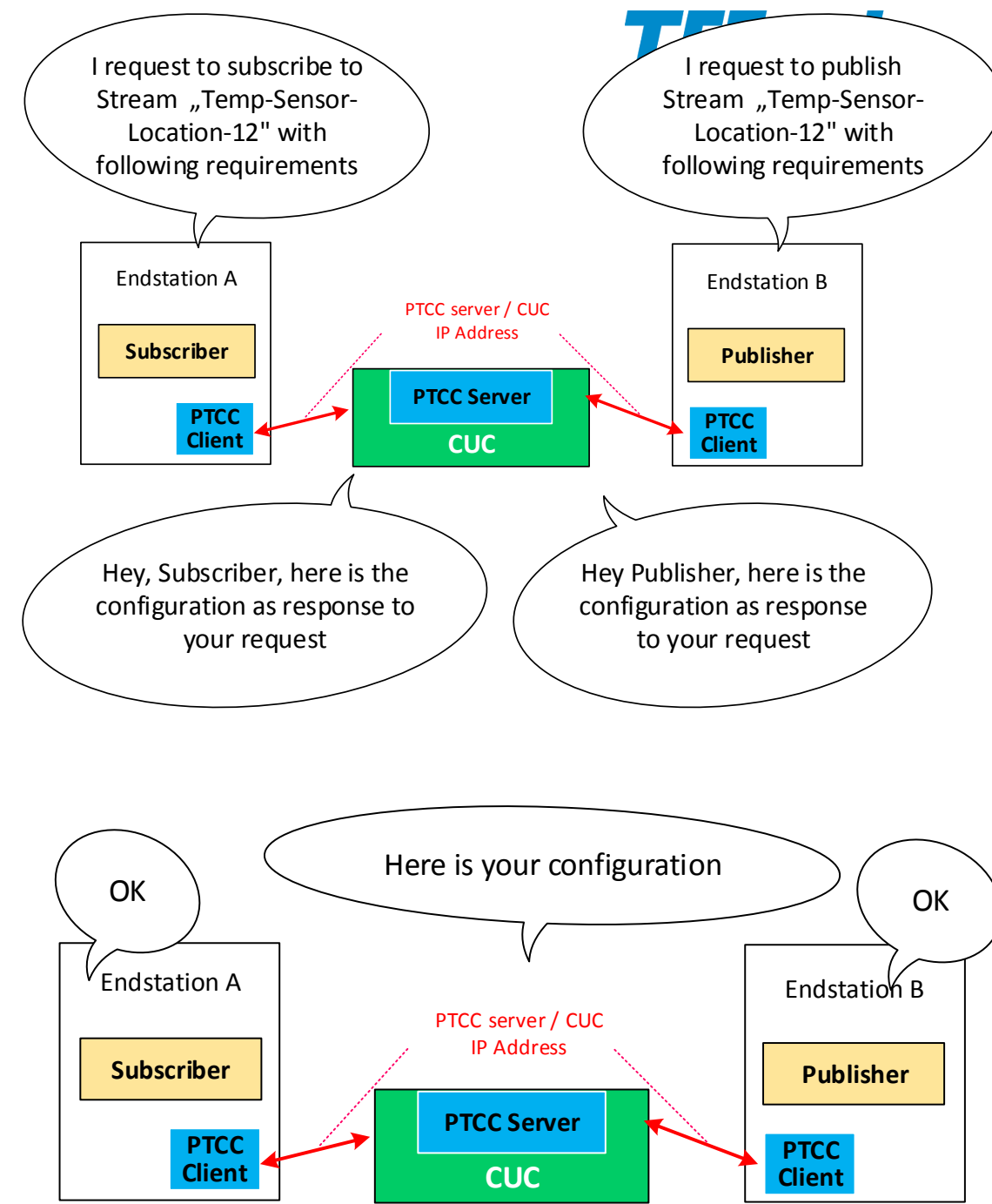
PTCC (Client) operational modes

- **Request Mode**

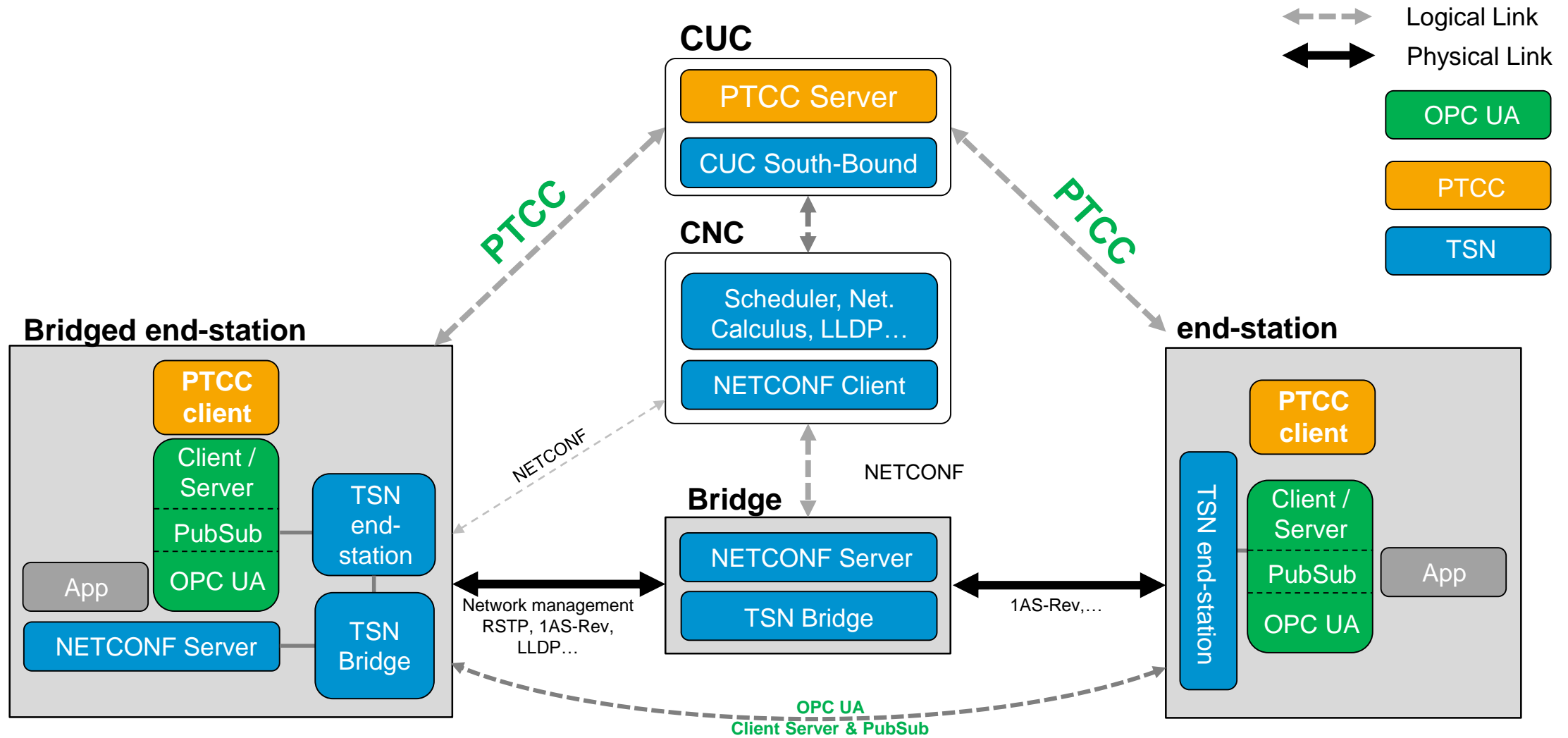
- Configuration requests are initiated by the PTCC-Clients

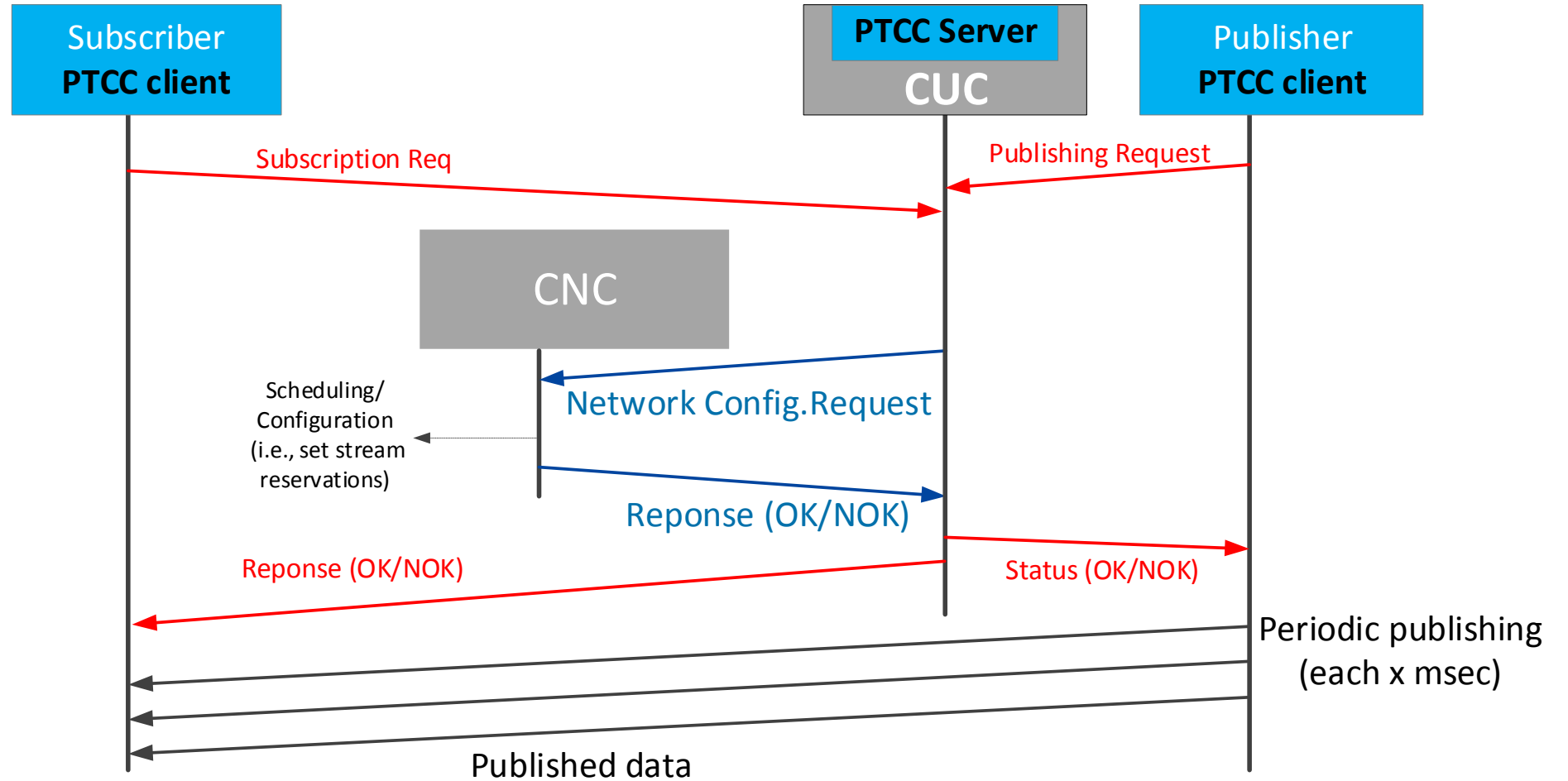
- **Push Mode**

- Configuration requests are initiated either by a proxy-device (i.e., controller, or an engineering tool within a configuration management device/PC). In this mode a proxy-device can send the request to the server on behalf of the PTCC client. The request sent by the proxy-device on behalf of a target device to the PTCC-servers called a proxy-request
- Content of the request is the same as in the “request mode”

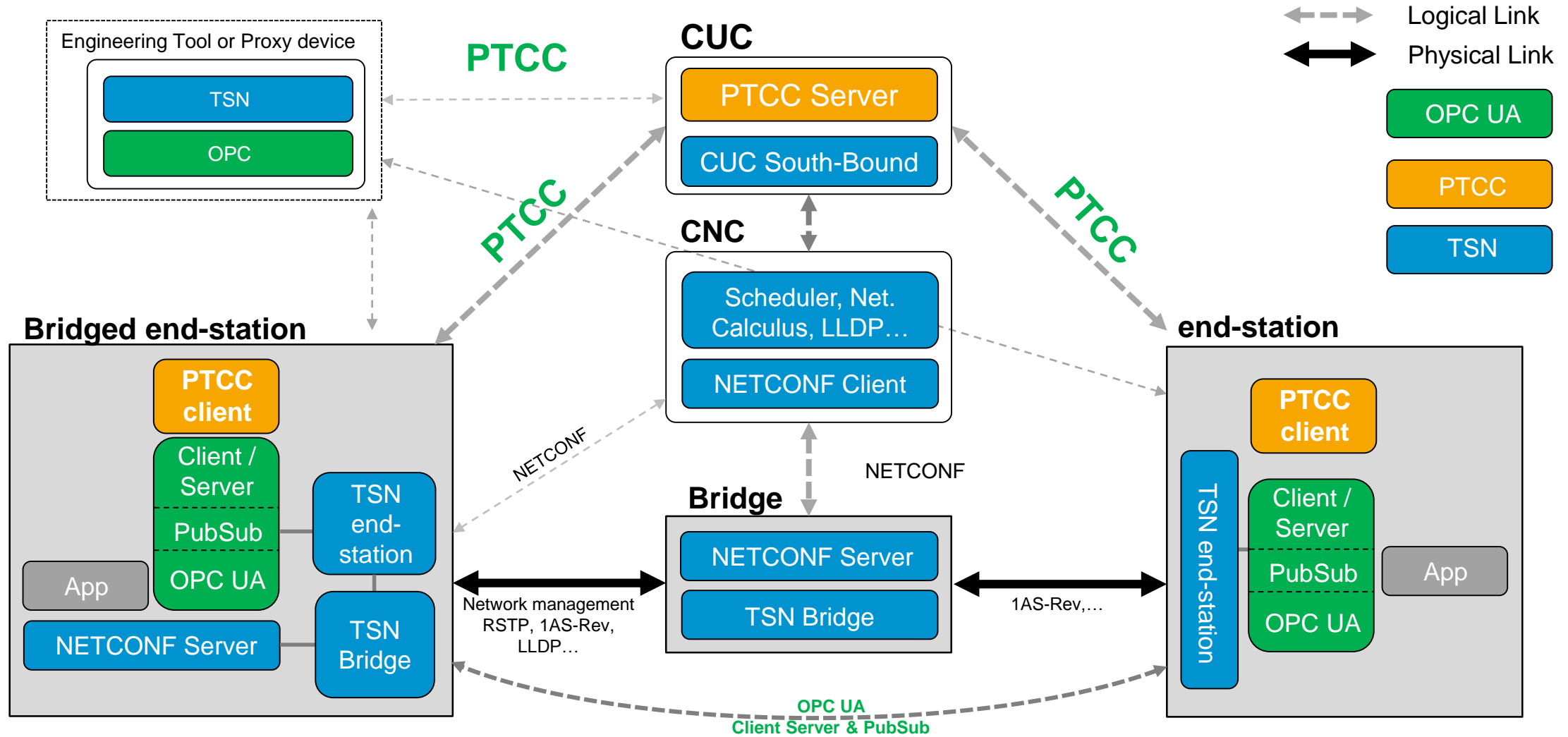


Involved actors in the "Request Mode"

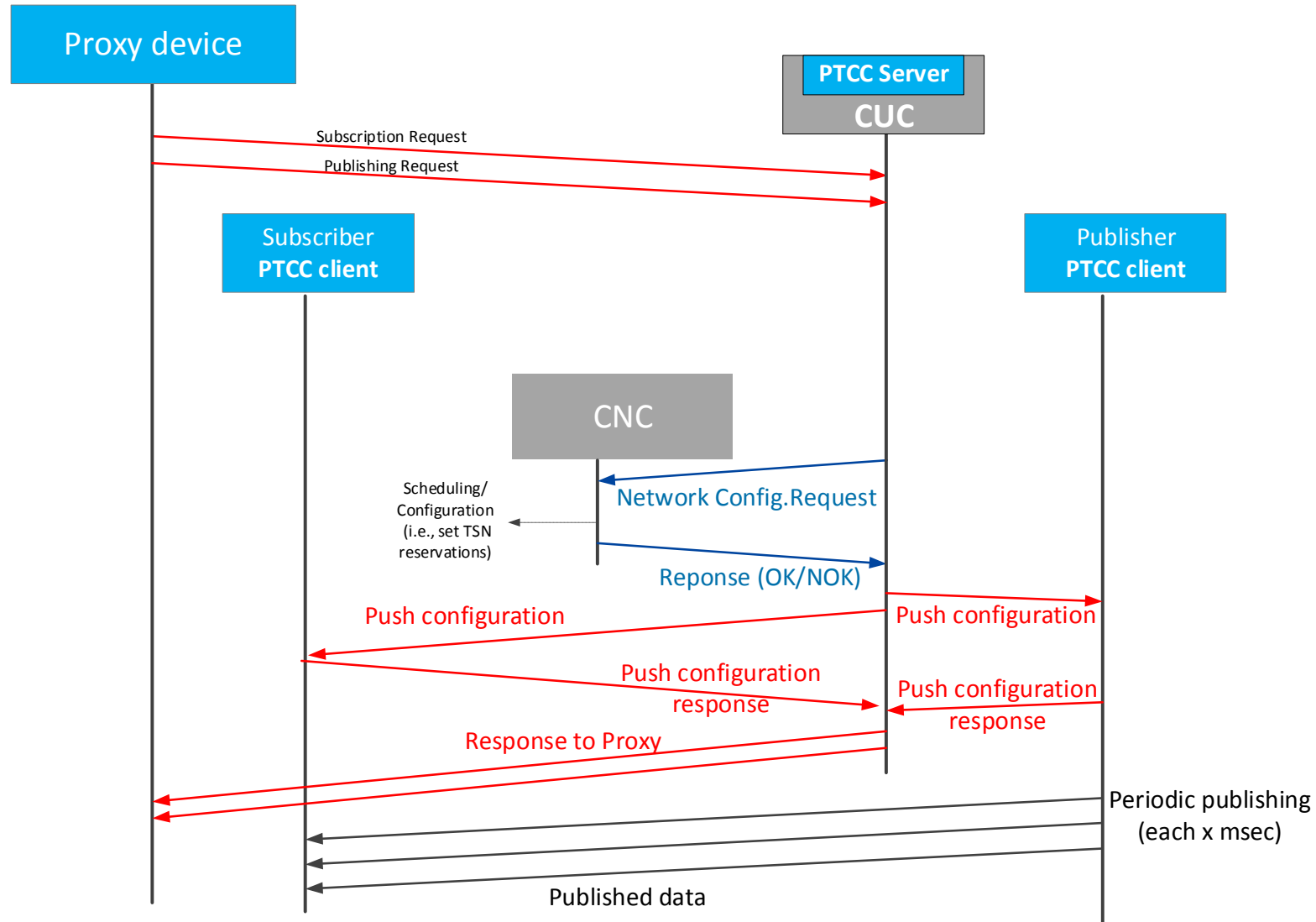




Involved actors in the "Push Mode"



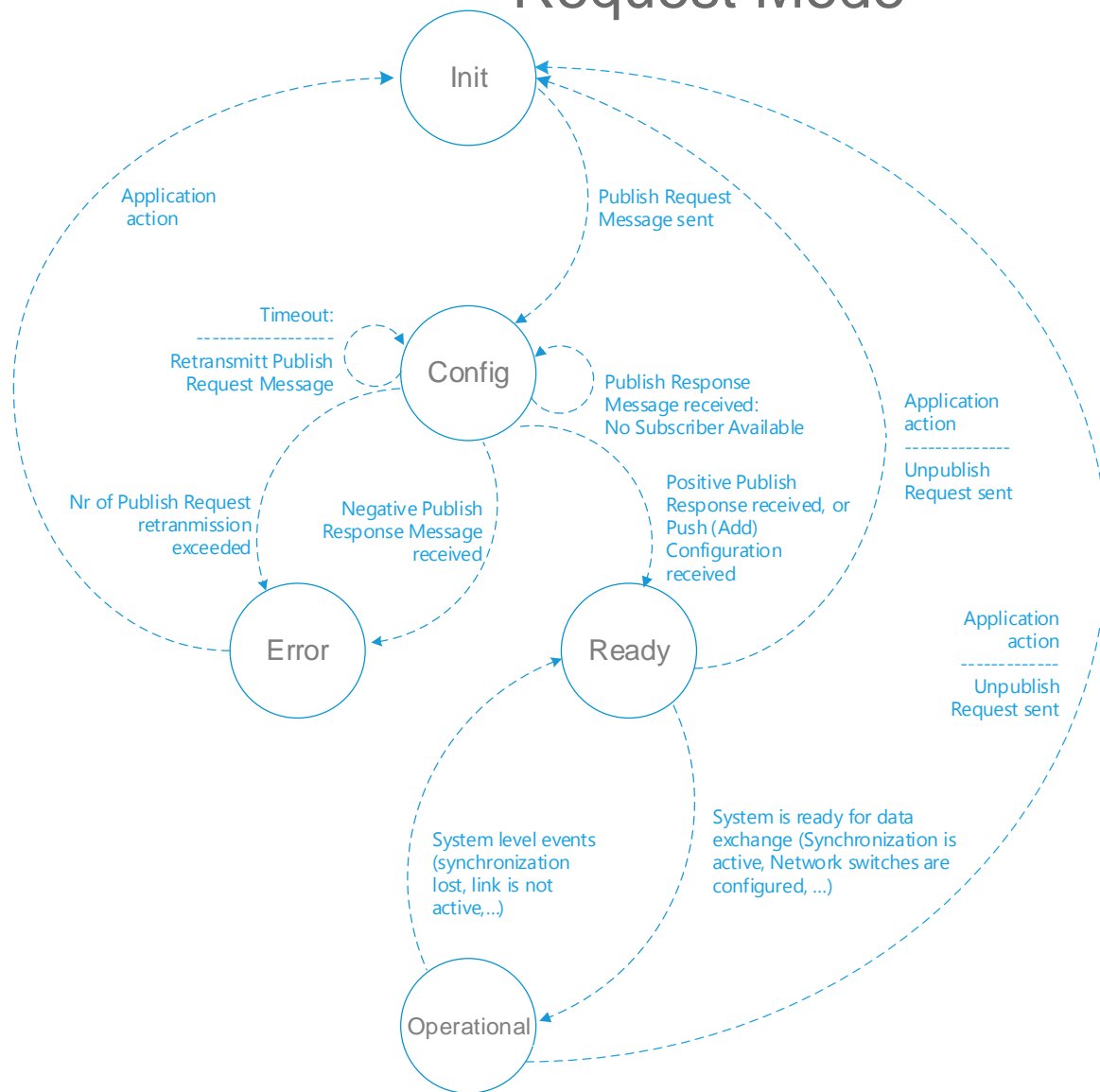
Push Mode



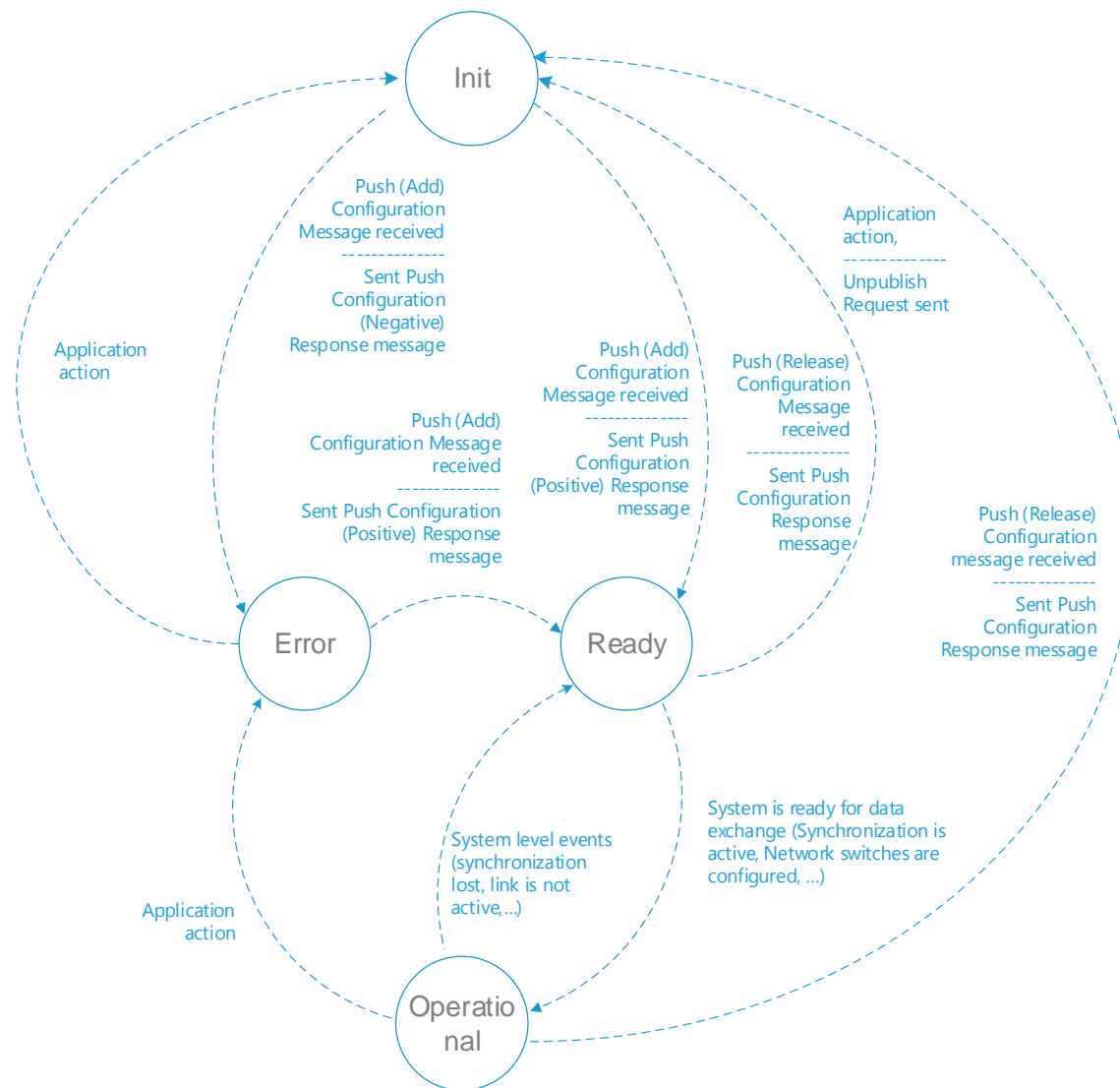
- Push Mode support is mandatory in the PTCC-Client
- In case that a Publisher sends a “Unpublish” request, the PTCC-server will send Push Request to Subscriber to “Unsubscribe” that stream.
- In case that all existing Subscribers send “Unsubscribe” request, the PTCC-server will send Push Request to Publisher to “Unpublish” that stream.

- Stream in the PTCC contains the
 - state of the stream configuration
 - configuration parameters
- State machine for a StreamObject
 - PTCC-client Publisher
 - PTCC-client Subscriber
 - PTCC-Server
- Aligning the state machine with the distributed stream reservation model such that towards the OPC UA PubSub interface - there is a single state machine (some renaming are outstanding)

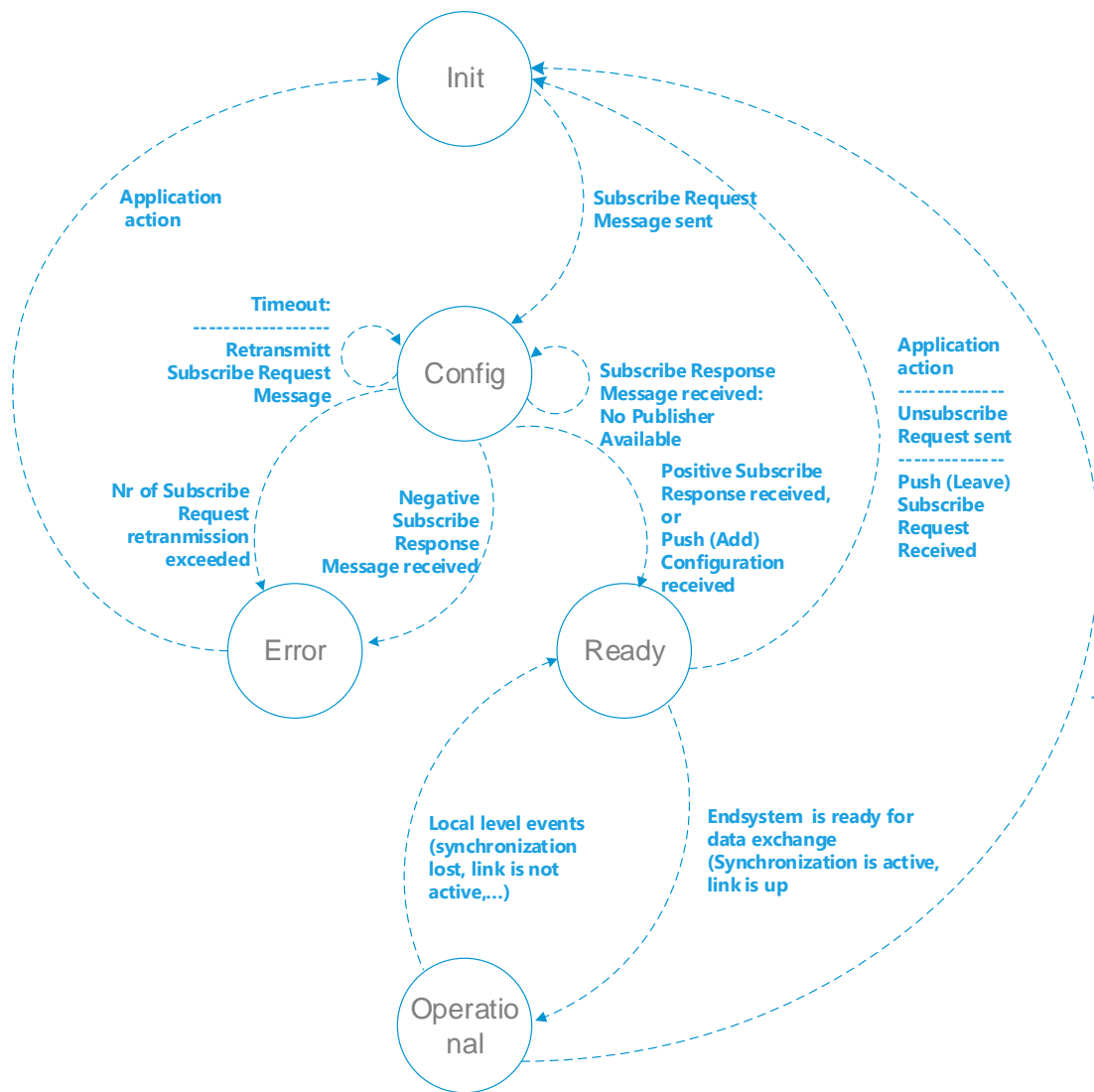
Request Mode



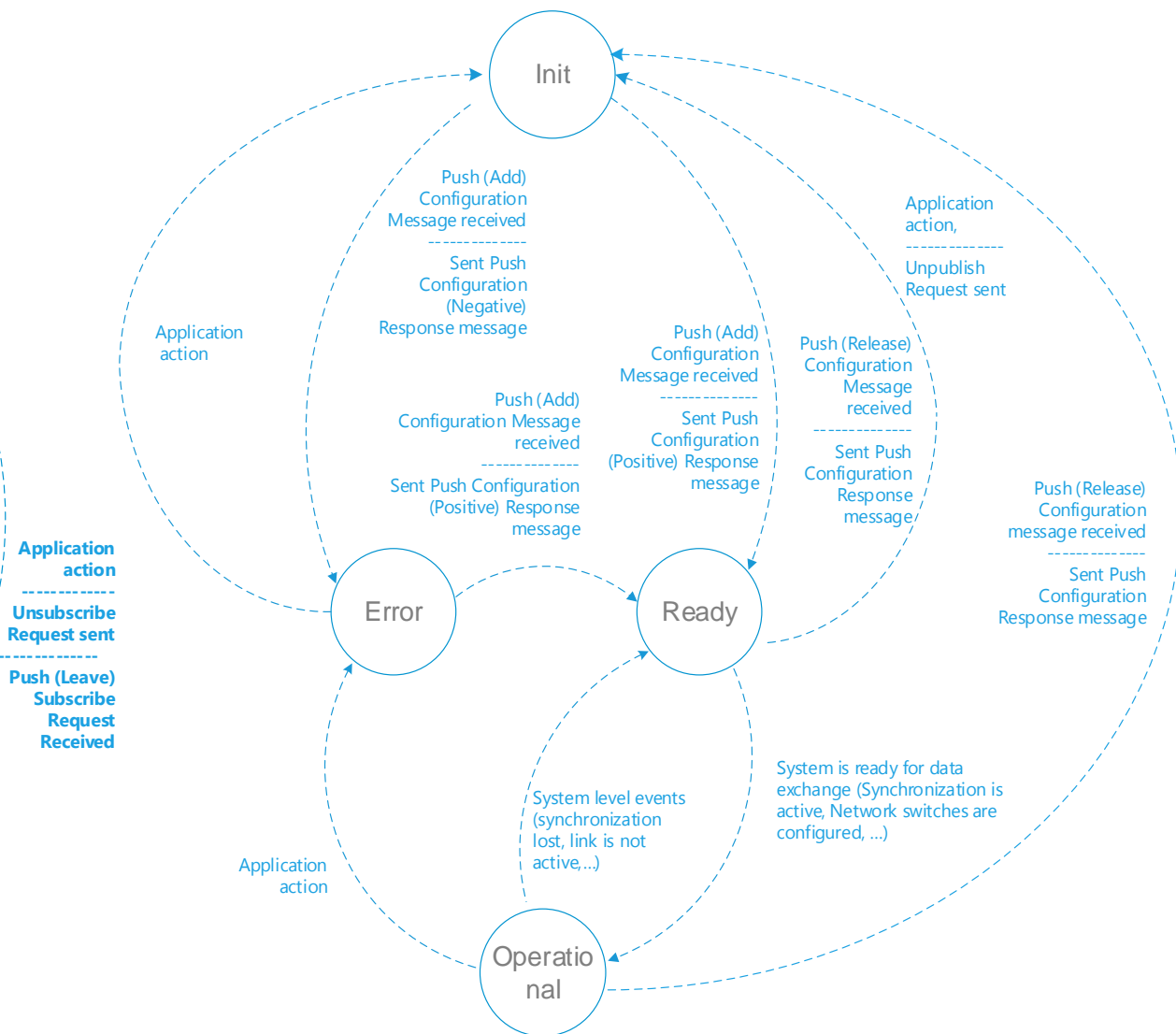
Push Mode



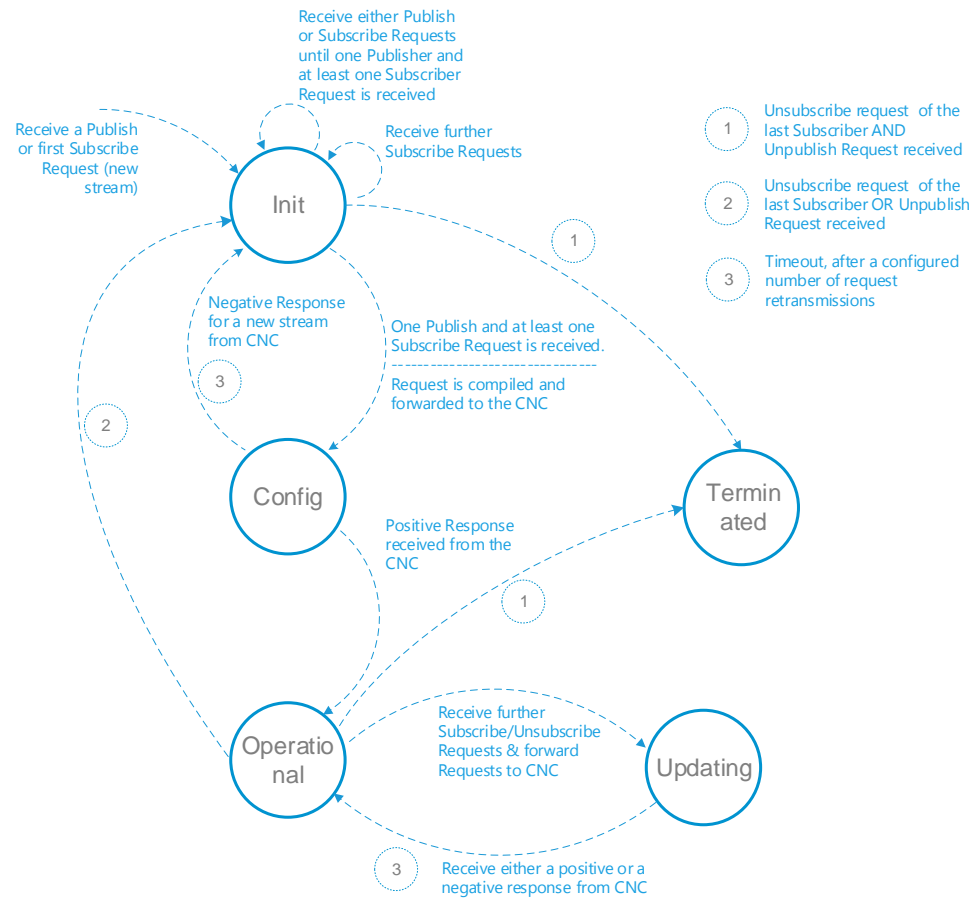
Request Mode



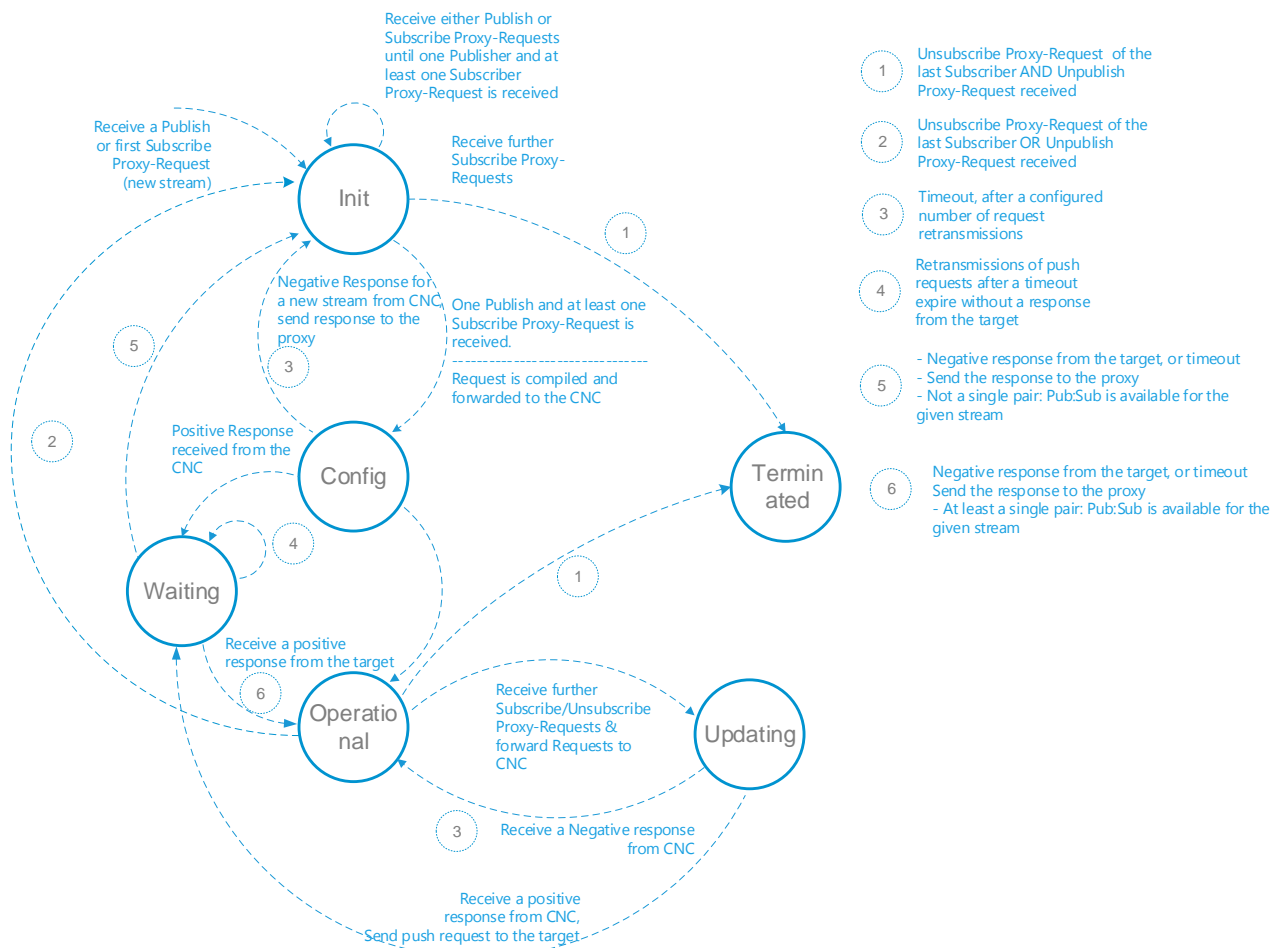
Push Mode



PTCC-server: Request Mode



PTCC-server: Push Mode



- PTCC clients send keep alive messages to the PTCC server
- This is used for two purposes:
 - To release the resources in case endstations fail “silently”
 - In case of a failure of a device that executes the PTCC-server functionality, the PTCC-server shall use the Keep-Alive messages to restore the configuration state

- The CNC need to have at least one Publish and one Subscribe Request in order to process/generate the stream configuration for a new stream (a stream which is not yet configured in the network)
- the CNC can process single requests coming from endstations for an existing stream (already configured in the system), i.e.,
 - Subscribe Request (from an additional subscriber)
 - Unsubscribe Request (from an existing subscriber)
 - Unpublish Request (from the publisher)
- CNC can generate the configuration for the scheduled and non-scheduled streams
 - During stream configuration generation the CNC can handle the constraints coming from the CUC (PTCC-server) by considering:
 - Latency constraints from a Subscriber (listener/receiver)
 - Deadline constraints from a Subscriber (listener/receiver)
 - Send offset constraints from a Publisher (talker/sender)
 - Redundancy requirements

- A positive configuration response will be sent from the CNC towards CUC (PTCC-server) including endstation stream configuration parameters (DMAC, VLAN ID, ...) for the endstations, after the CNC has successfully
 - generated the configuration
 - distributed the configuration to the switches
 - received the confirmation from the switches that the configuration update is accepted
- CNC operates in incremental configuration mode, i.e., new requests do not change the configuration parameters of streams and devices that are not related to that request
- CNC responses to the CUC for each received request. If bulk request handling is supported by the CNC, this might be added in a future (ongoing discussions)

- CNC is idempotent, i.e., receiving multiple identical request (with the same StreamId) for shall produce the same configuration results as if the request is received at the first time
 - There are cases when the configuration result of a request is the same but the response to the CUC (PTCC-server) may be different, e.g., sending Unpublish Request multiple time, results in a stream being deleted from the network (reservations are released), but the response of the first request will be “stream deleted” the response of the seconds request will be “not found”. The result of both responses is that the stream is not configured/reserved in the network.

- Specifies the message based Interface
 - Publish Request
 - Publish Response
 - Subscribe Request
 - Subscribe Response
 - Un-publish Request
 - Un-publish Response
 - Unsubscribe Requests
 - Unsubscribe Response
 - Push Configuration to Publisher
 - Push Unpublish message
 - Push Configuration to Subscriber
 - Push Unsubscribe message
 - Push Configuration Publisher Response
 - Push Configuration Subscriber Response
 - Keep Alive Publisher
 - Keep Alive Subscriber
- Specifies the structure of the data exchanged between the PTCC Clients and PTCC-server
- Specifies the StreamObject state machine for PTCC-client Publisher, PTCC-client Subscriber and PTCC-server
- Requirements on the CNC operation
- Request Mode (ad-hoc networks)
- Push Mode (engineered networks)

Looking forward to use this as input for upcoming project (802.1Qdj) with respect to the CNC operation