Maintenance Item #387

IEEE 802.1AS Backwards Compatibility Issue between 2020 and 2011 Editions

Christian Aaen (ADI)
Woojung Huh (Microchip)
Alon Regev (Keysight)
Richie Pearn (NXP)
Stijn Geysen (Luminex)
Jeremy Rover (Intel Corporation)
Ganesh Venkatesan (Intel Corporation), 802.1AS

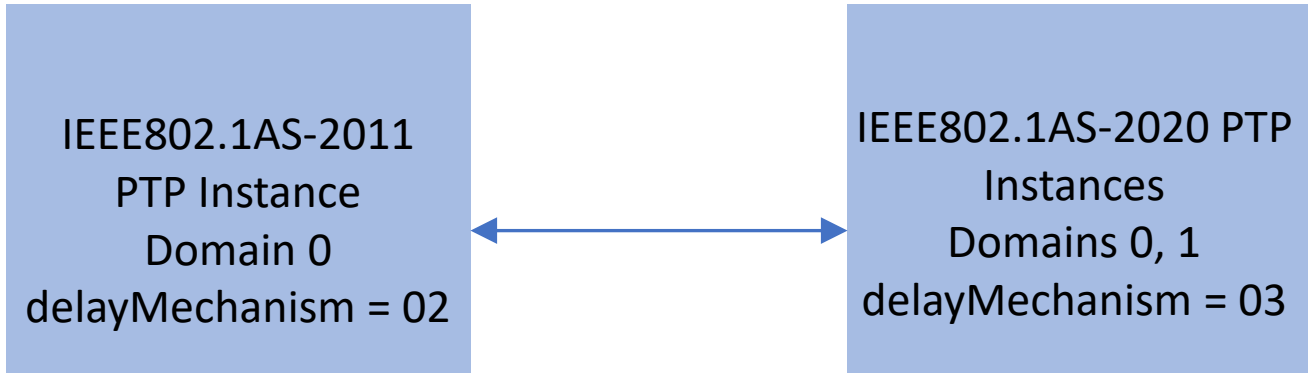
IEEE802.1 Maintenance TG

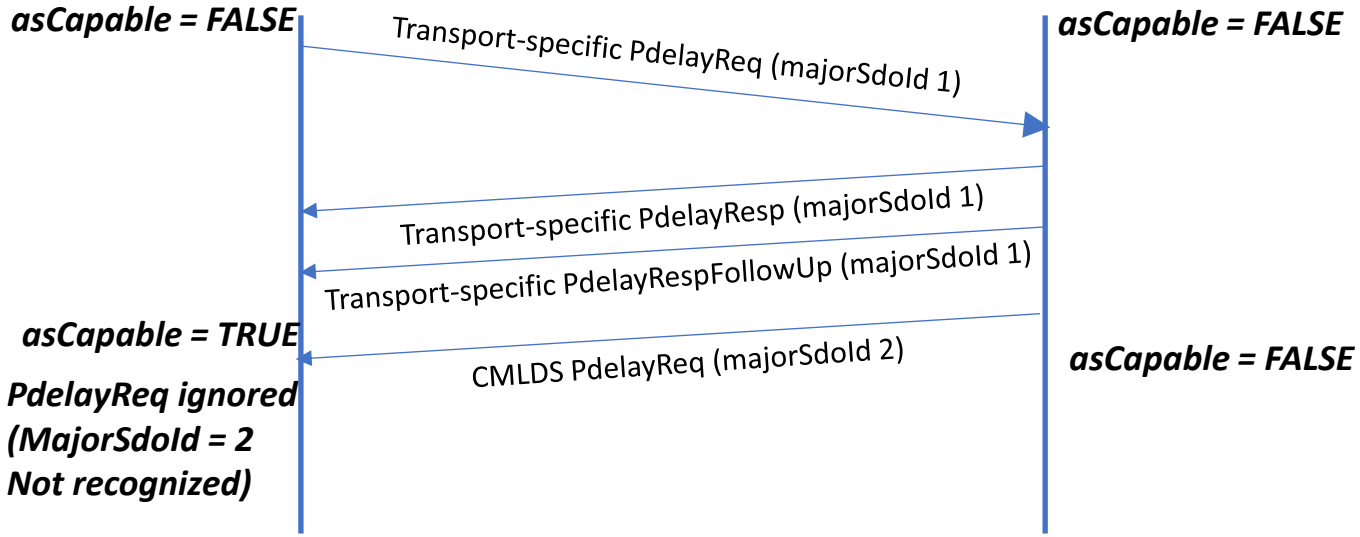
2025.05.19

Goals of this discussion

- Present the problem we encountered when a IEEE802.1AS-2011 implementation is peered with a IEEE802.1AS-2020 implementation
- Get feedback from IEEE802.1AS experts -- specifically clarity on Backward Compatibility, PortDS.delayMechanism, PTP Port versus Link Port
- Discuss potential clarifications in IEEE802.1AS – i.e., add NOTEs where appropriate

What did we try?





1. What is the definition of Backward Compatibility?
   Our Expectation: Pair any IEEE802.1AS-2020 implementation with any IEEE802.1AS-2011 – expect IEEE802.1AS-2020 to step down to IEEE802.1AS-2011 'mode' of operation (transmit Pdelay Request messages with MajorSdoId set to 1; set delayMechanism to 02)
2. Is this a valid test?
3. Or should the test setup configure the IEEE802.1AS-2020 implementation with delayMechanism set to 02 and use only domain 0?

General Problem

- The issue can also be seen if two IEEE802.1AS-2020 implementations supporting domain 0 are peered and one has the delayMechanism set to 02 (P2P) and the other has the delayMechanism set to 03 (COMMON_P2P)

Who is responsible for setting delayMechanism?

- Management entity?
  - The PTP Instance reads the value of delayMechanism and acts accordingly
- Management entity and the PTP entity
  - The PTP Instance reads the value of delayMechanism and acts accordingly
  - The PTP Instance updates the value of delayMechanism based on operating conditions (specifically, peered with a PTP Instance that is a IEEE802.1AS-2011 implementation)

PTP Port and Link Port

- portDS is associated with a PTP Port and has the member delayMechanism (set to 02 for transport-specific peer delay or to 03 for CMLDS peer delay)
- commonServicesPortDS is associated with a PTP Port
  - Explicitly stated in 1588-2019 Cl. 8.2.27, as providing management access to the common services port (CMLDS port, i.e., the Link Port)
  - Uses the delayMechanism value from portDS
- Not clear if delayMechanism can be written only by the management entity (and read by the PTP Instance) or if it can also be written by the PTP Instance

What would help?

1. Define what is meant by Backward Compatibility
2. Add a NOTE-2 to Cl. 11.2.17.1 on how the IEEE802.1AS-2020 PTP Instance behaves if it detects a peer that is IEEE802.1AS-2011
   a) There is normative text (above NOTE-3 in Cl. 11.2.17.1 on how the IEEE802.1AS-2020 PTP Instance responds to IEEE802.1AS-2011 transport-specific PdelayRequest messages)
   b) Extend the text above to describe expected behavior when a IEEE802.1AS-2011 peer is detected
3. Neither 802.1AS-2011, 802.1AS-2020, nor 802.1AS-2020-Rev are clear on what fields must be verified on receive and what fields must only be set during transmit (except for reserved fields where the spec is clear that they are set to 0 on transmit and ignored on receive).
   1. e,g., Control field: IEEE802.1AS-2020 Implementation should ignore the control field in messages transmitted by a IEEE802.1AS-2011 peer; IEEE802.1AS-2020 implementations set the control field to 0 in all ptp messages. A IEEE802.1AS-2011 implementation receiving these messages should ignore (and not check it to be 0 in Sync, 2 in Follow_Up and 5 in other PTP messages) – address this with a new comment.

Potential Fix to compatibilty

- If a PTP Port that invokes the CMLDS receives a Pdelay_Req message with majorSdoId value of 0x1, minorSdoId value of 0x00, and domainNumber value of 0, the PTP Port shall respond with PTP Instance specific peer delay messages (i.e., the Pdelay_Resp and Pdelay_Resp_Follow_Up corresponding to this Pdelay_Req) using the instance-specific peer-to-peer delay mechanism. These instance-specific messages have majorSdoId value of 0x1, minorSdoId value of 0x00, and domainNumber value of 0

  The above requirement provides: (this is NOTE 4 in .1AS-2020-Rev)
- a) Backward compatibility with time-aware systems that comply with the 2011 version of this standard provided that delayMechanism is set to 02 (P2P, See Cl. <tbd>), and
- b) Compatibility with time-aware systems that implement only one domain and invoke the MDPdelayReq and MDPdelayResp state machines on domain 0, provided that delayMechanism is set to 02 (P2P, See Cl. <tbd>)
- ~~NOTE 4—~~In general, a port can receive:

Questions?