

# ETHERNOVIA

TRANSFORMING HOW CARS OF THE FUTURE ARE BUILT

---

IEEE802.1ASDM – REDUNDANCY

NOVEMBER 2021

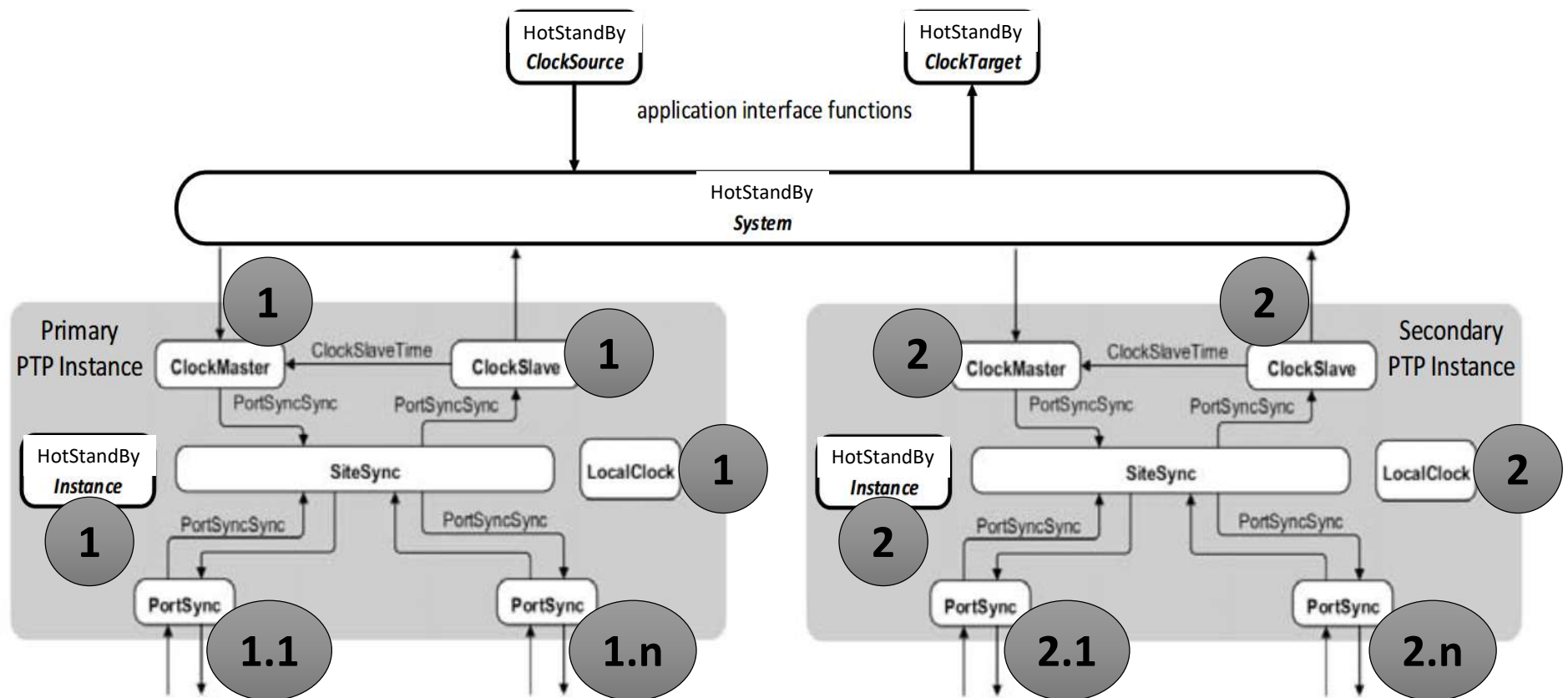
IEEE contribution

ETHERNOVIA

TRANSFORMING HOW CARS OF THE FUTURE ARE BUILT

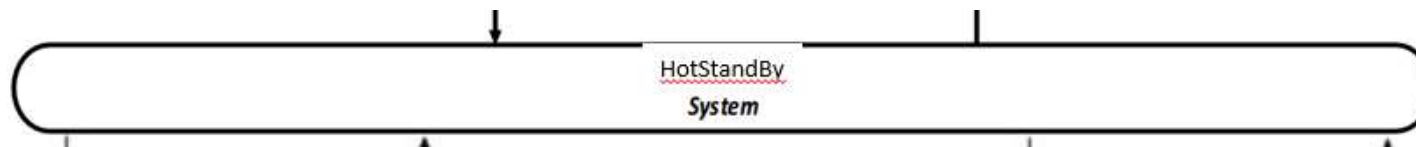


# Figure 17-1—Model of hot standby redundancy for time synchronization





# HotStandBy System States

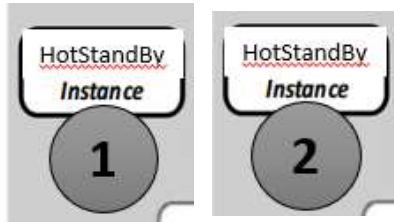


**17.10.1 HotStandbySystemState:** State of the HotStandbySystem for this hot standby timescale. The variable is an enumeration that takes one of the following values:

- a) INIT: Initialization after the HotStandbySystem powers on and is enabled. In this state, the system is waiting for both PTP Instances to synchronize.
- b) REDUNDANT: Both PTP Instances are synchronized according to the requirements of the respective application or TSN profile. Time synchronization is redundant.
- c) FAULT: One PTP Instance is synchronized, and the other PTP Instance is faulted (not synchronized). Time synchronization continues to meet the requirements of the respective application or TSN profile. Time synchronization is not redundant.
- d) HOLDOVER: Both PTP Instances are faulted (not synchronized). The HotStandbySystem is adjusting phase/frequency of its local time using the data stored in REDUNDANT or FAULT state, but that local time will eventually drift relative to other time-aware systems. During HOLDOVER state, time synchronization might not meet the requirements of the respective application or TSN profile.



# HotStandBy Instance(s) States



**17.5.1 hotStandbyInstanceState:** State of the PTP Instance with respect to requirements of the respective application or TSN profile. The variable is an enumeration that takes one of the following values:

NOT  
Synced!

a) NOT\_CAPABLE: For one or more external ports that is enabled for PTP, the neighbor is not exchanging the messages that are required for conformance to this standard.

Synced!

b) SYNCED: Time is synchronized to the requirements of this standard and the respective application or TSN profile.

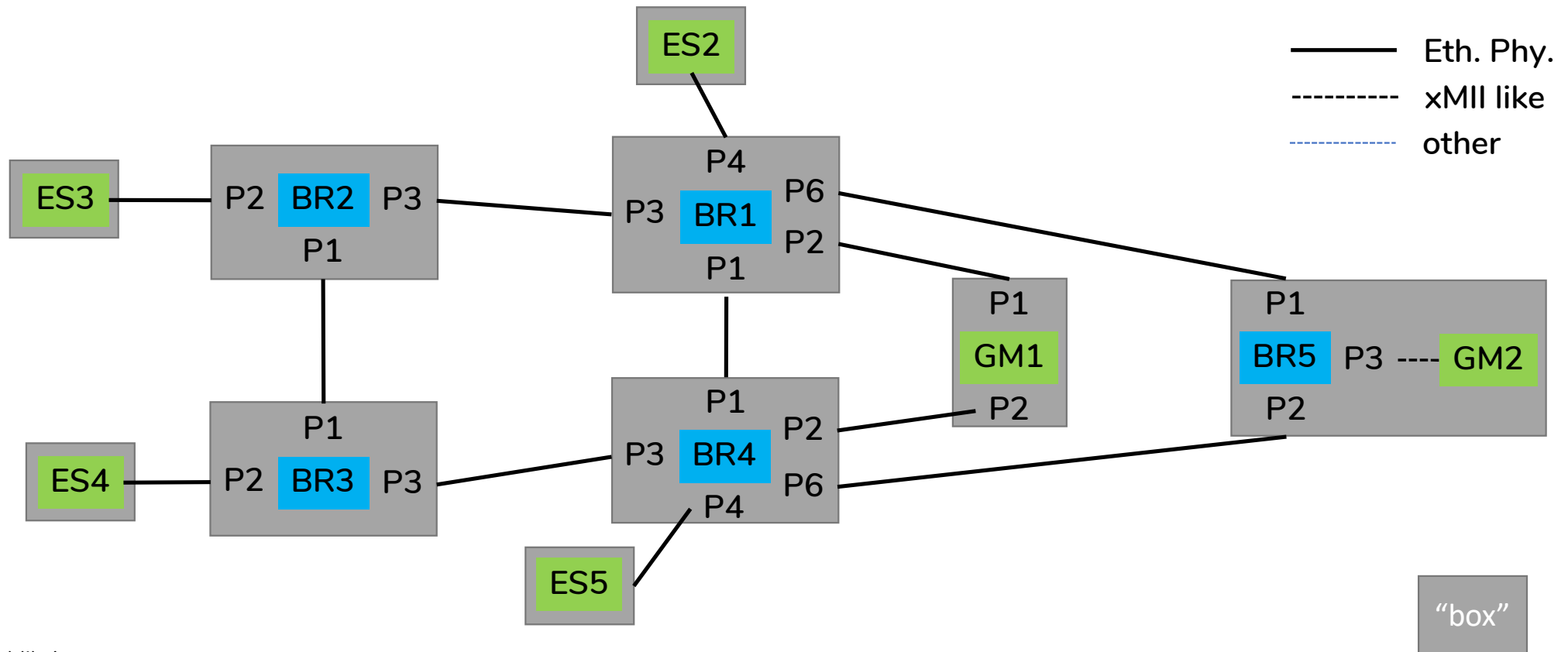
NOT  
Synced!

c) SYNC\_TIMEOUT: On the external port in SlavePort state, the port failed to receive time sync event messages according to the requirements of this standard.

d) NOT\_QUALITY: Time synchronization does not conform to the quality requirements of the respective application or TSN profile.



# Remember this Network?



Thanks Mike!

<https://www.ieee802.org/1/files/public/docs2020/dm-potts-as-hot-standby-multiple-domains-0220-v01.pdf>

IEEE contribution



# Agenda

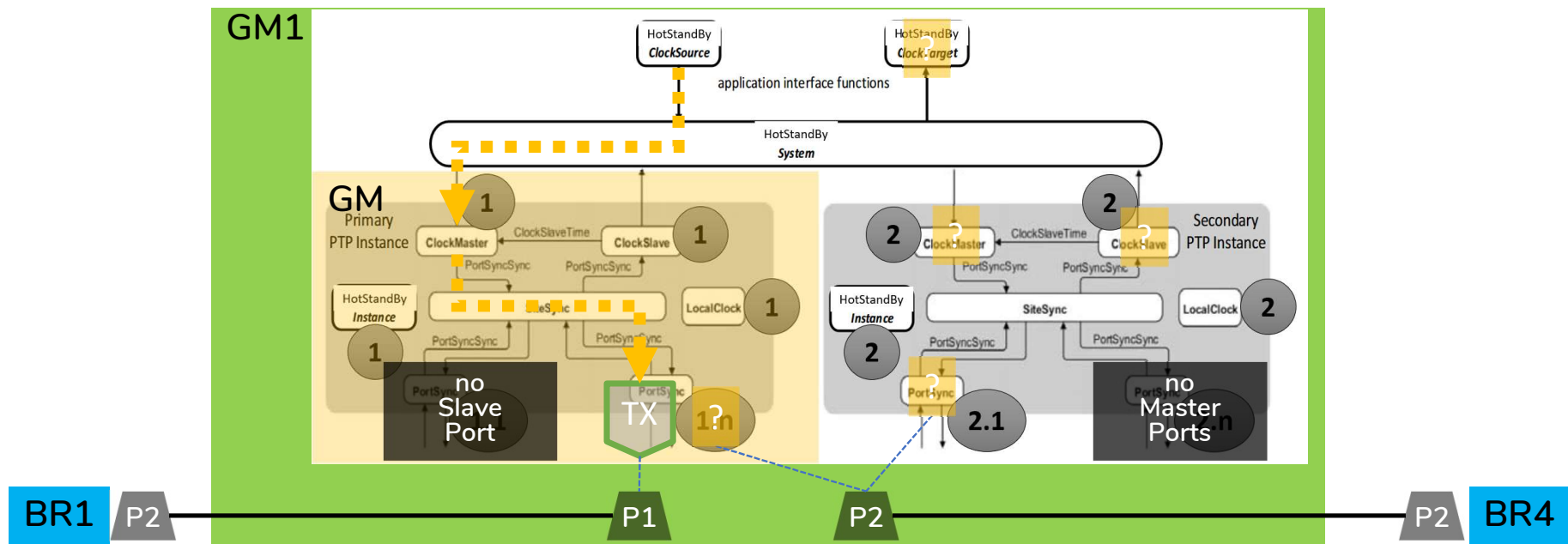
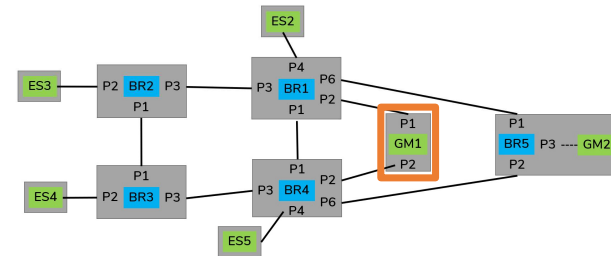
- How does the current ASdm Draft describe/enable
  - GrandMaster 1
  - GrandMaster 2 (at Bridge 5)
  - Bridge 5
  - Bridge 1 (or 4)
  - EndStation 2 (or any other)
- Simplify GM2Bridge5 within the current ASdm Draft
- Motivating NEW features NOT within the current ASdm Draft
  - Redundant TimeAware-Bridge
  - Redundant GM
  - Holdover Bridge
- One more open question on the current ASdm Draft
- The issue of checking time-bases against each other



# GrandMaster 1

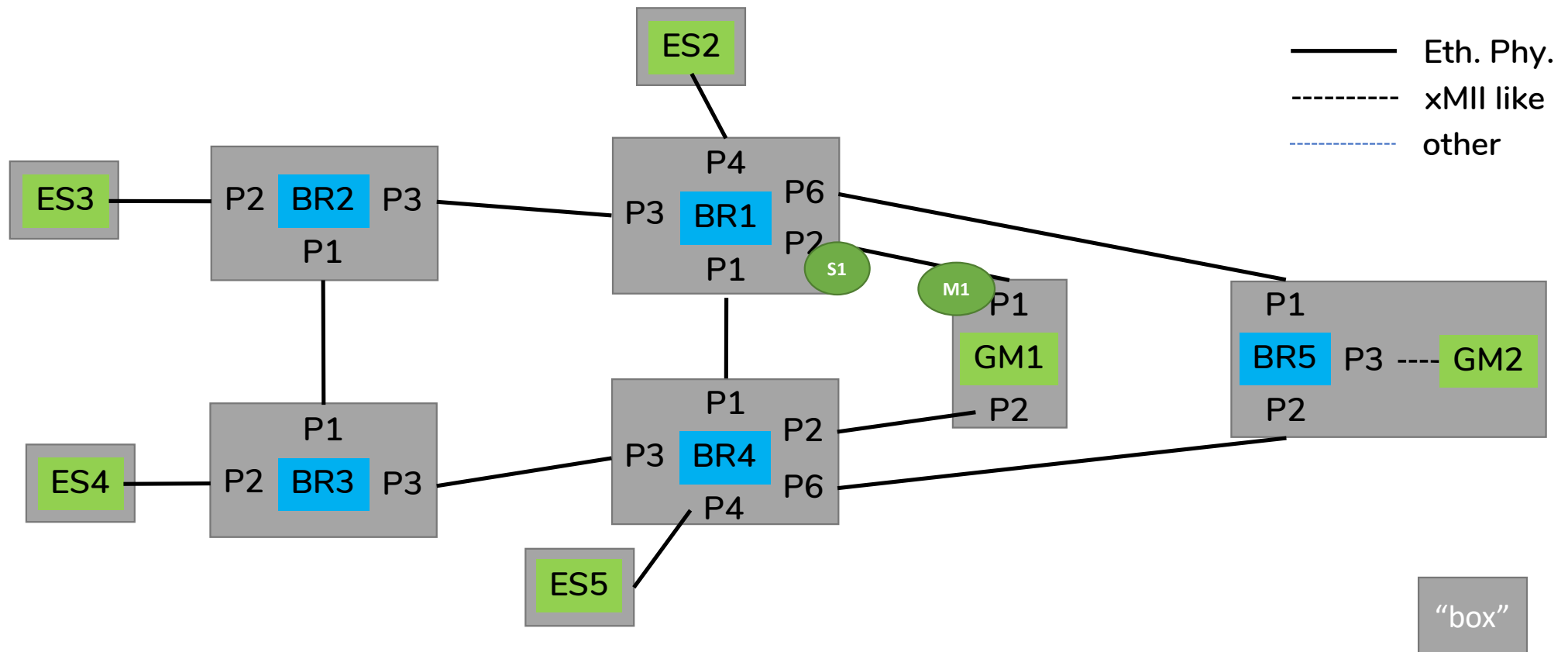
## 17.4.3.1 Primary grandmaster

When the primary PTP Instance is grandmaster (i.e., no external port in SlavePort state), the HotStandbySystem shall transfer phase and frequency from the HotStandbyClockSource to the ClockMaster of the primary PTP Instance (see Figure 17-1). If no external source of time is implemented, the HotStandbyClockSource is equivalent to the LocalClock.





# GM1 connected to BR1



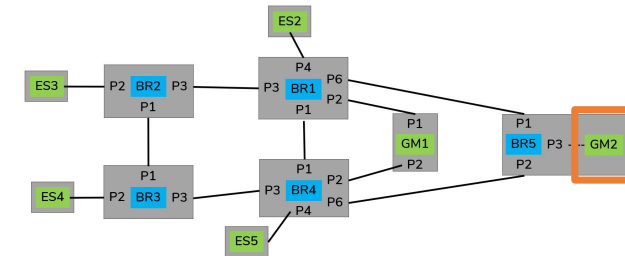
Thanks Mike!

<https://www.ieee802.org/1/files/public/docs2020/dm-potts-as-hot-standby-multiple-domains-0220-v01.pdf>

IEEE contribution

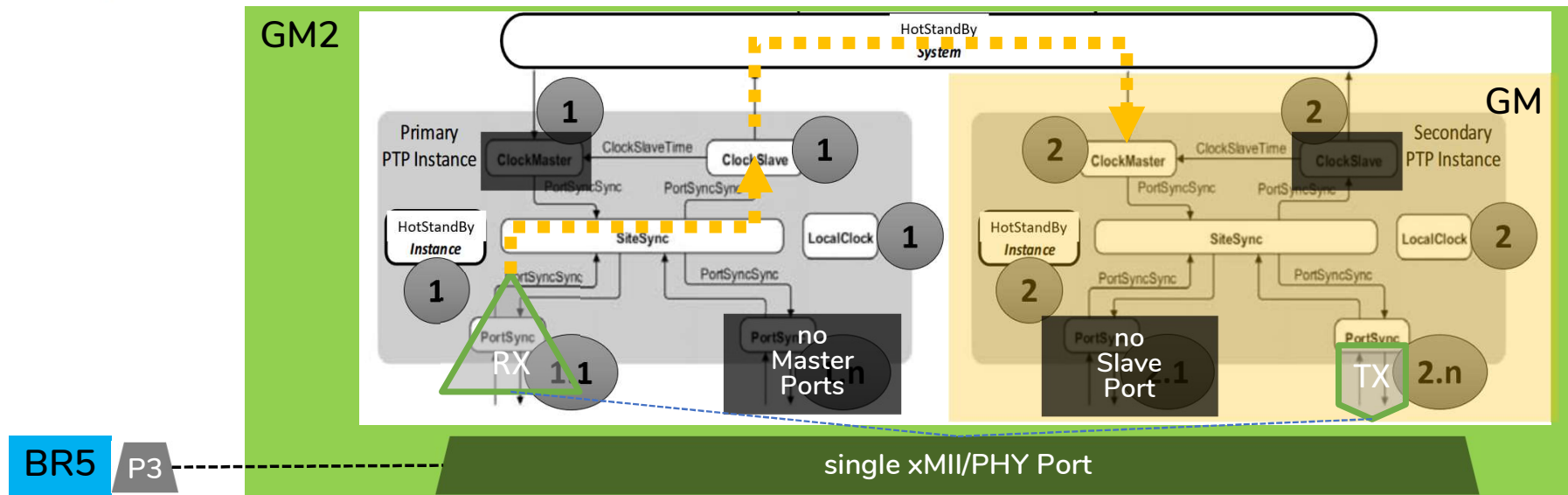


# GrandMaster 2 at Bridge 5



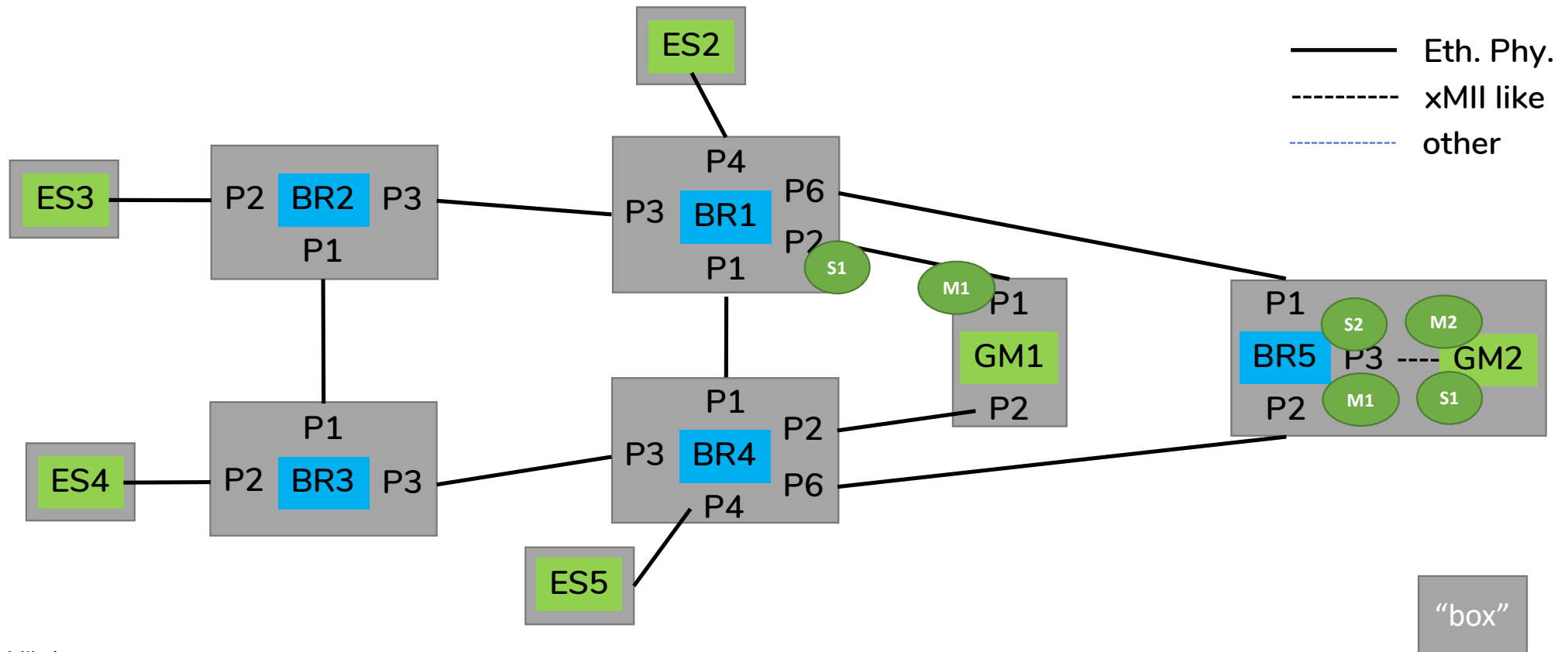
## 17.4.3.2 Secondary grandmaster in REDUNDANT state

When the secondary PTP Instance is grandmaster (i.e., no external port in SlavePort state), and the HotStandbySystemState is REDUNDANT, the HotStandbySystem shall transfer phase from the ClockSlave of the primary PTP Instance to the ClockMaster of the secondary PTP Instance (see Figure 17-1). By using phase from the primary PTP Instance, the secondary grandmaster can maintain continuity in the event of a fault in the primary grandmaster.





# GM2 connected to BR5



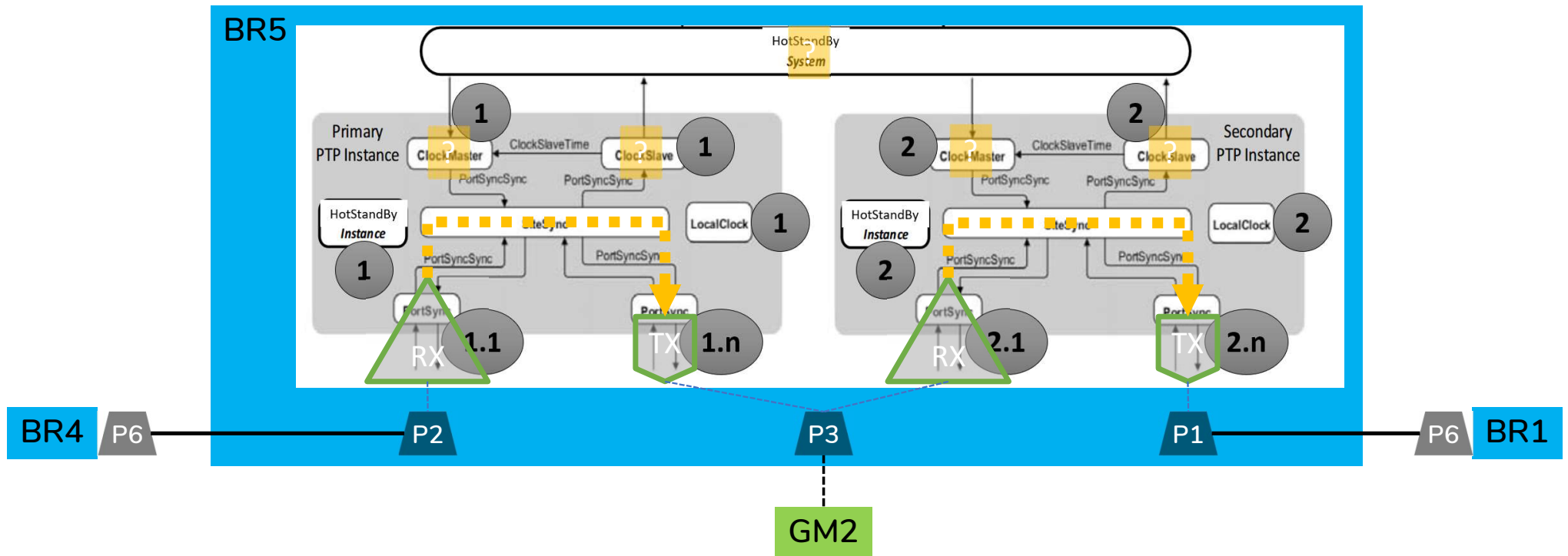
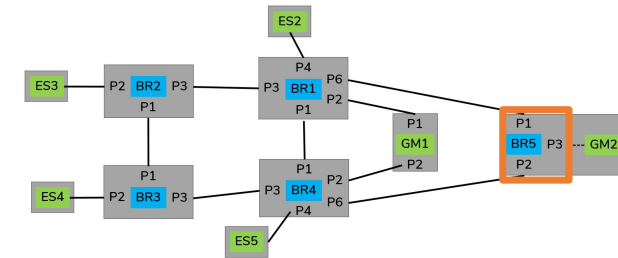
Thanks Mike!

<https://www.ieee802.org/1/files/public/docs2020/dm-potts-as-hot-standby-multiple-domains-0220-v01.pdf>

IEEE contribution

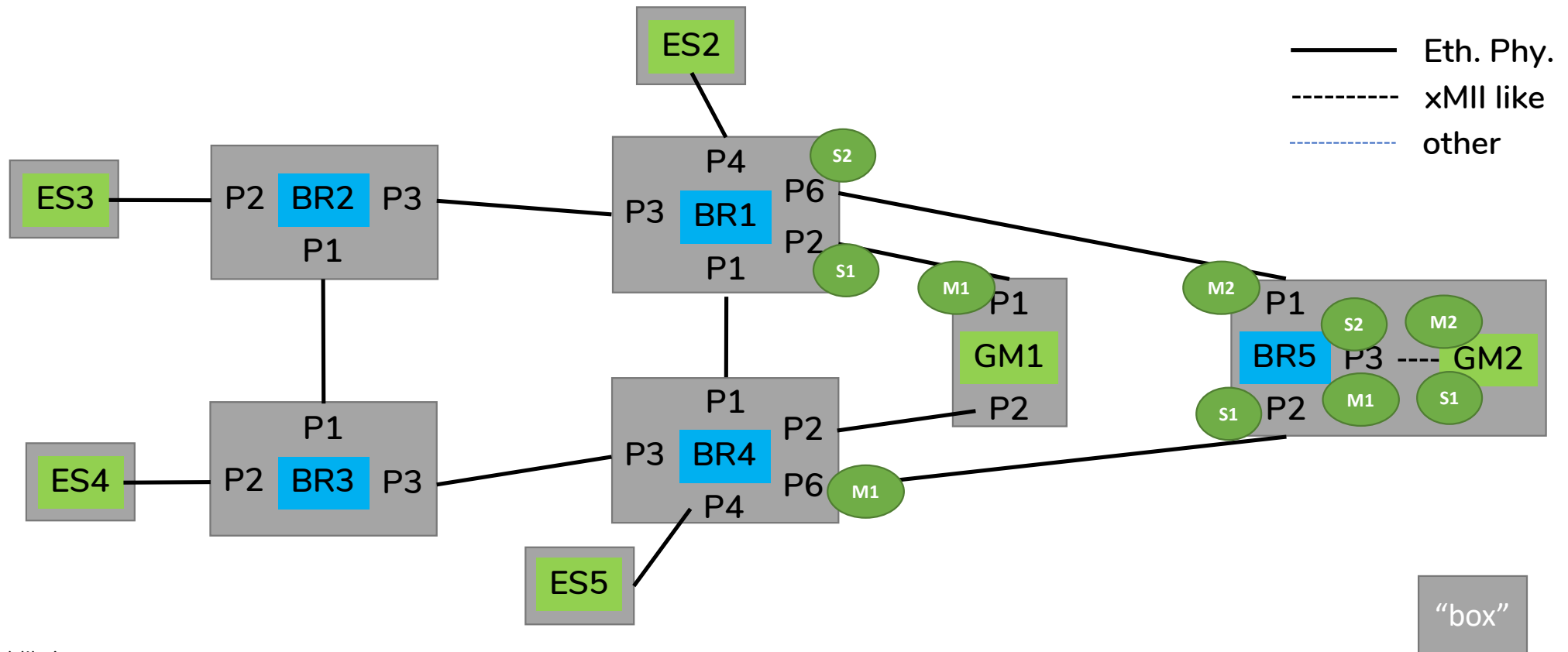


# Bridge 5





# BR5 connected to BR1 and BR4



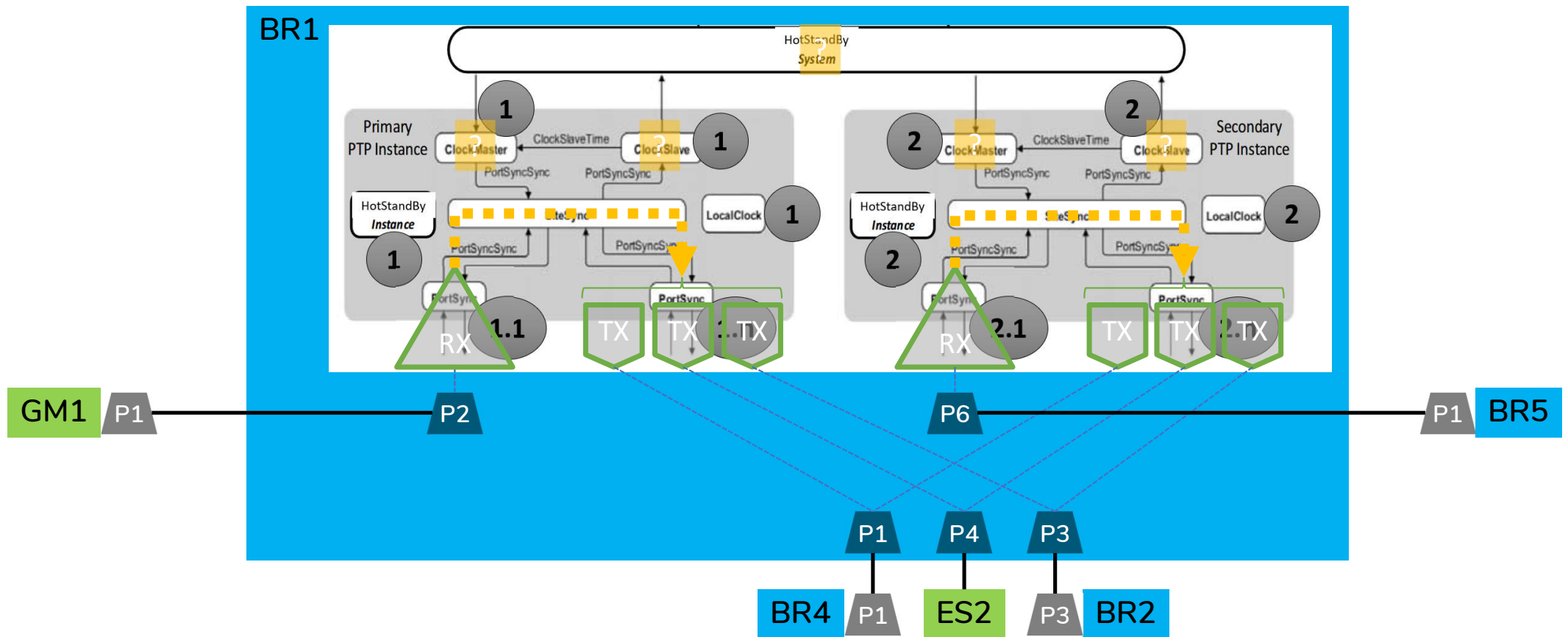
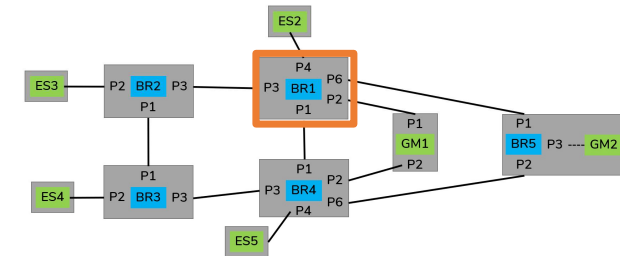
Thanks Mike!

<https://www.ieee802.org/1/files/public/docs2020/dm-potts-as-hot-standby-multiple-domains-0220-v01.pdf>

IEEE contribution

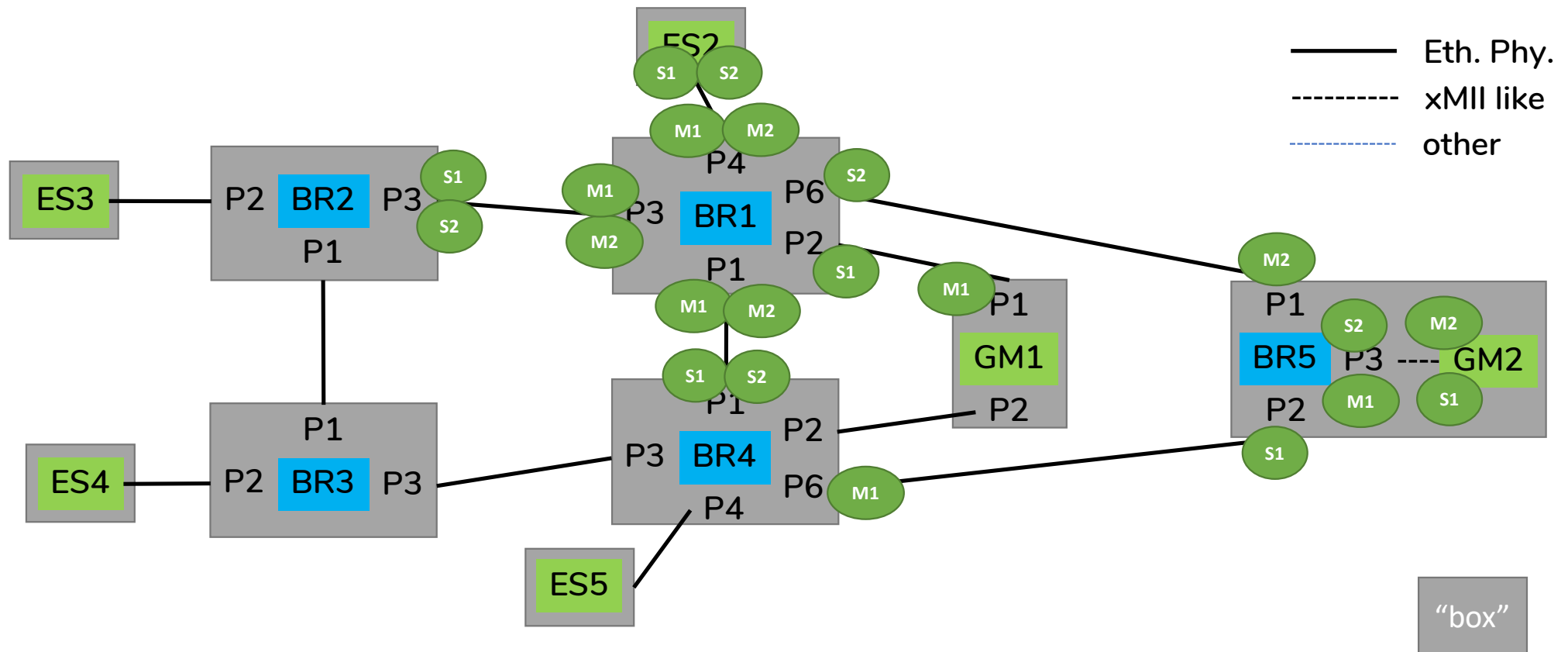


# Bridge 1





# ES2, BR2, and BR4 connected to BR1



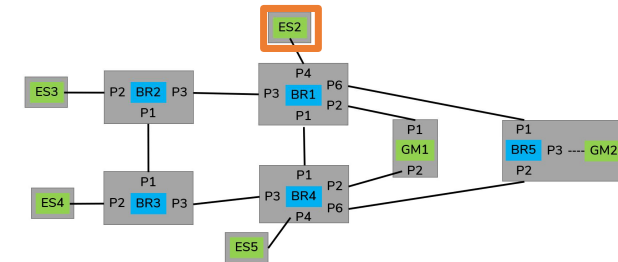
Thanks Mike!

<https://www.ieee802.org/1/files/public/docs2020/dm-potts-as-hot-standby-multiple-domains-0220-v01.pdf>

IEEE contribution

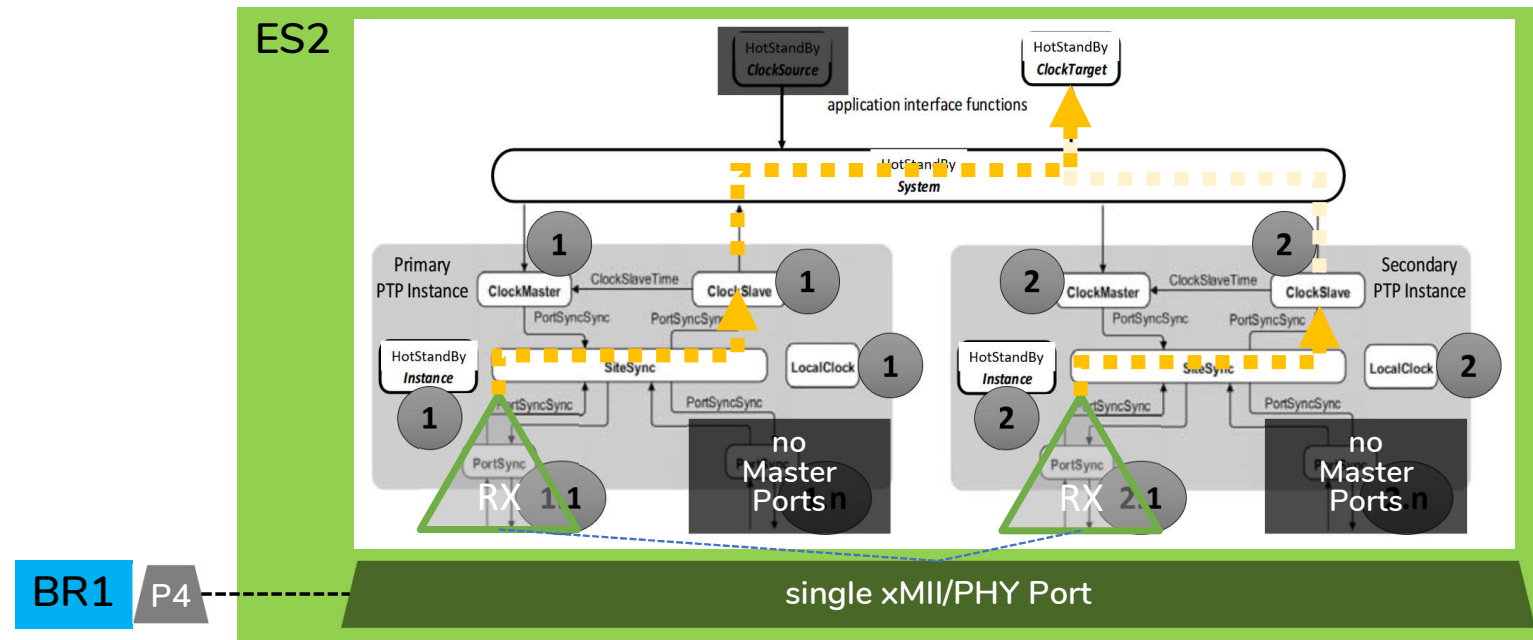


# EndStation 2



## 17.4.3.3 Slave

When the HotStandbySystem is a slave (i.e., neither PTP Instance is a grandmaster), and HotStandbySystemState is REDUNDANT, the HotStandbySystem shall transfer phase and frequency from the ClockSlave of the primary PTP Instance to the HotStandbyClockTarget (application).





# What does ASdm do for the network?

- Limited Functionality in GrandMaster1
- Limited Functionality in Bridges 5 and 1
- Bridging time between ClockSlave1 and ClockMaster2 in GrandMaster2
- Switching between ClockSlave1 and ClockSlave2 in the EndStations
- The redundancy of time provided by ASdm is focussed on EndStations which receive multiple time-inputs!



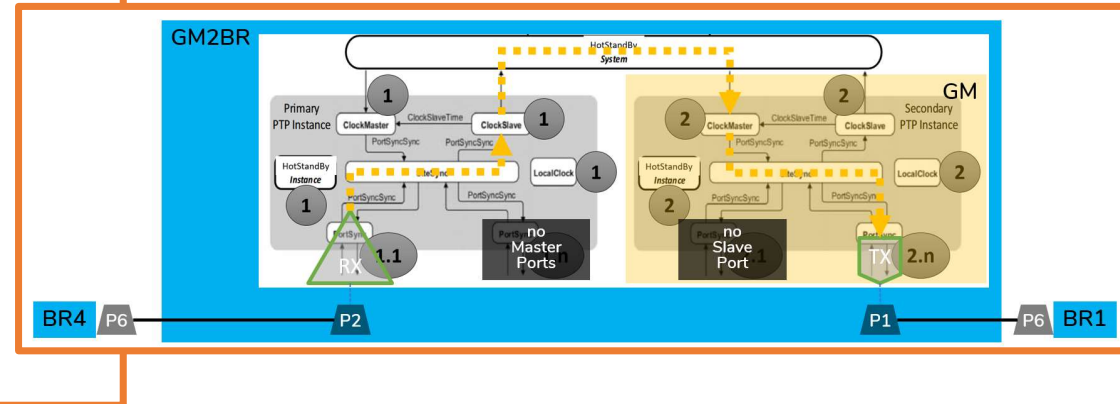
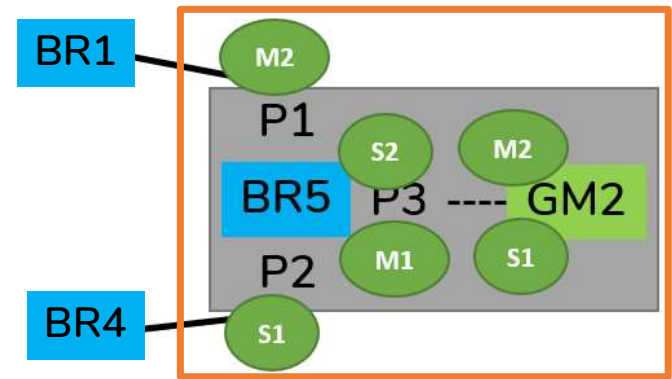
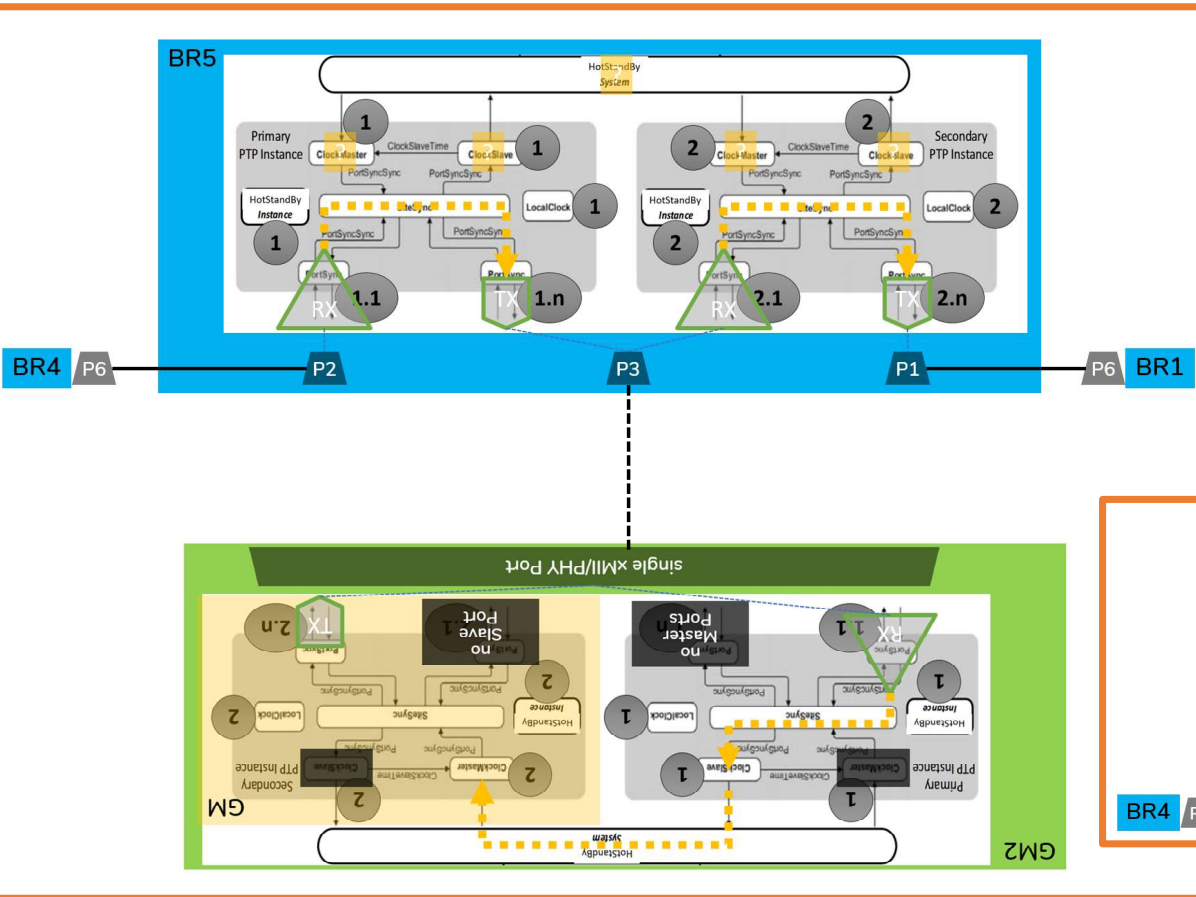
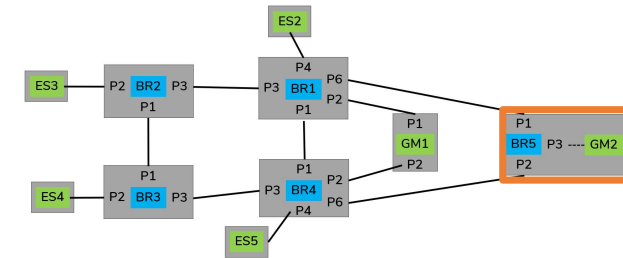
An abstract network diagram composed of numerous blue dots (nodes) connected by thin blue lines (edges). The network is sparse on the left and becomes increasingly dense towards the right side of the slide, forming a complex web of connections.

# Simplify GM2Bridge

within the current ASdm Draft



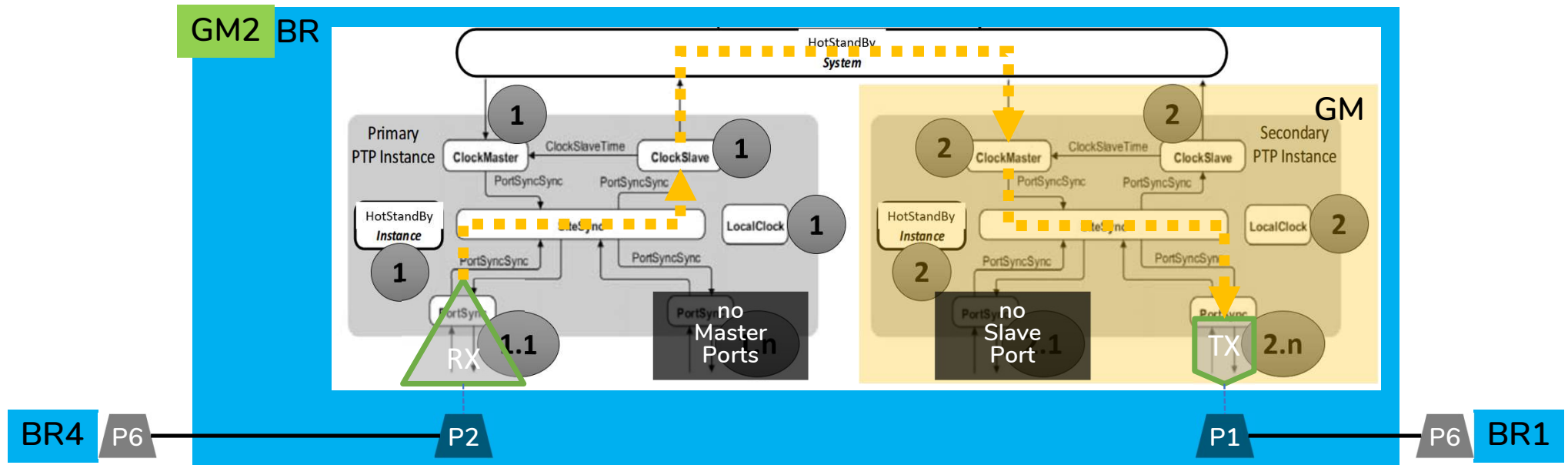
# GrandMaster 2 Bridge



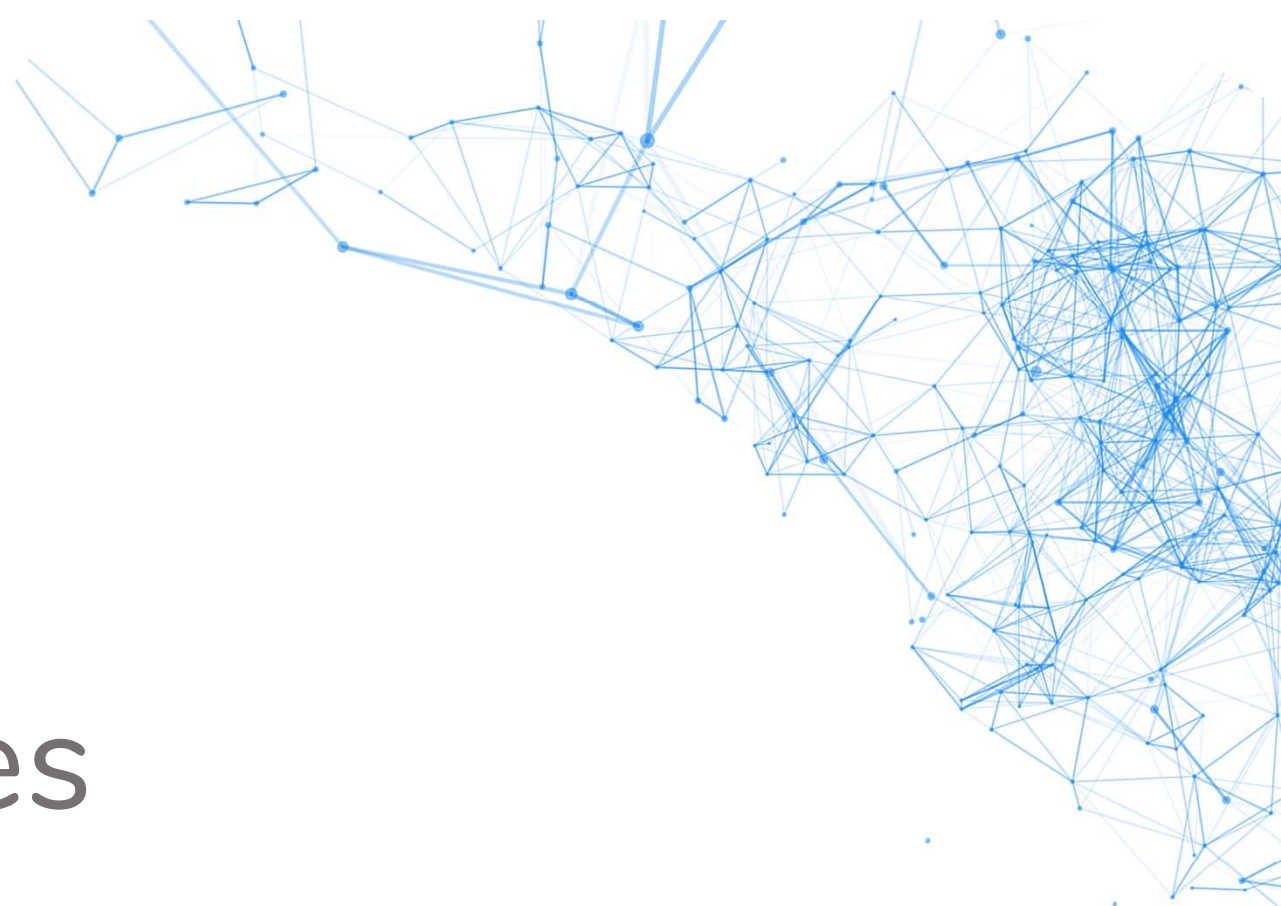


# Simplified GrantMaster 2 Bridge5

Is this covered by the current ASdm draft?







# NEW features

NOT in current ASdm Draft!

(potentially to be extended further after this discussion)

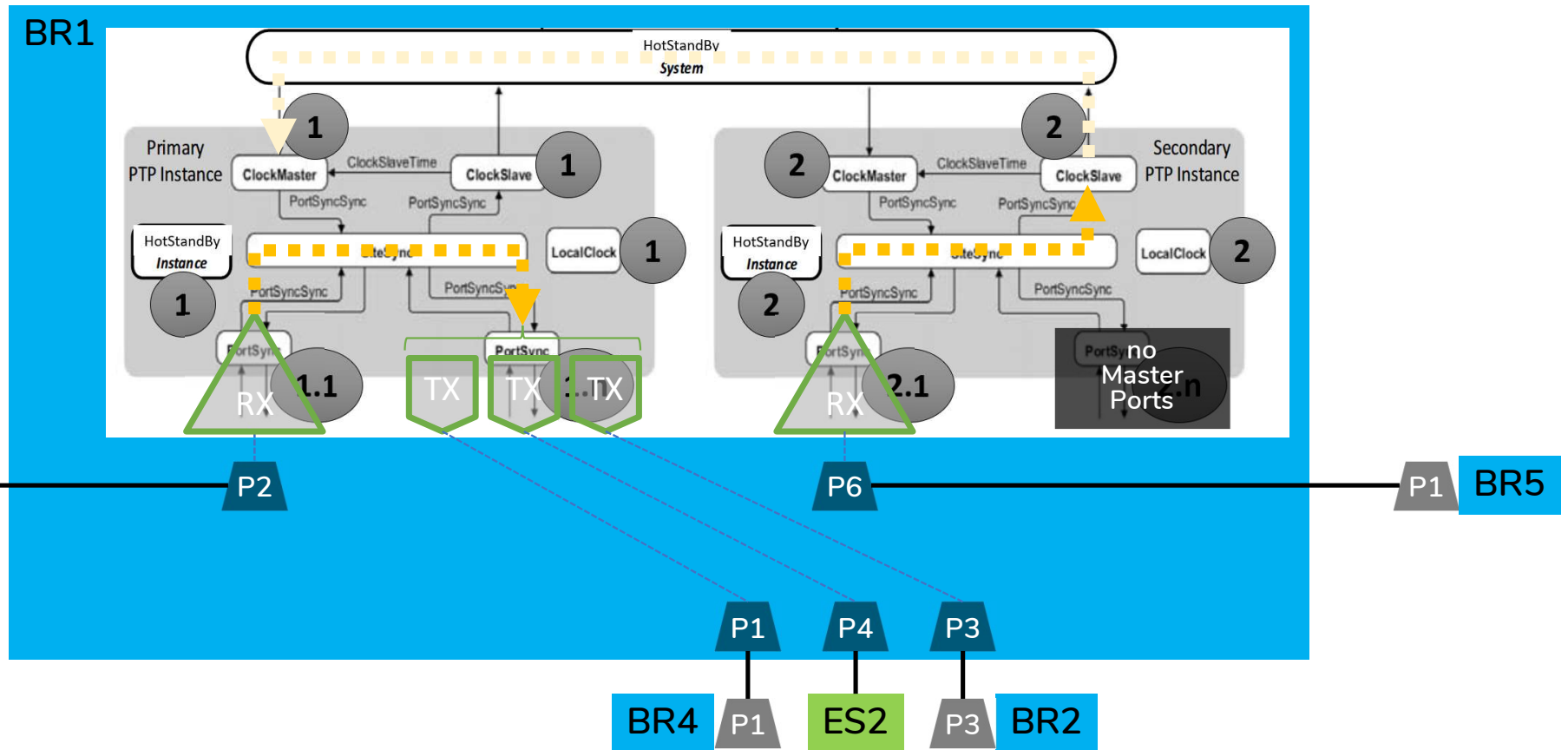
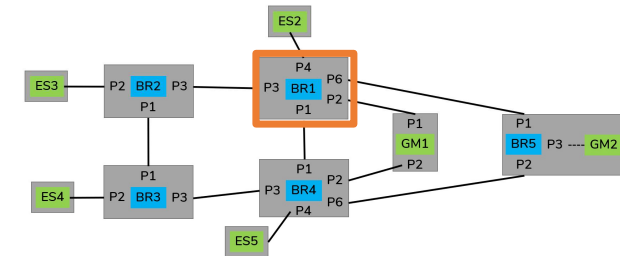


# Redundancy in the Network

- Carrying 2 time domains on the same link (in the same direction)
  - only protects the EndStation from a GM or remote Bridge failure
  - does not protect against link failures
  - does not protect from failures in the transmitting master of that link
- Can one achieve the same redundancy for the EndStations without carrying two domains over many links?

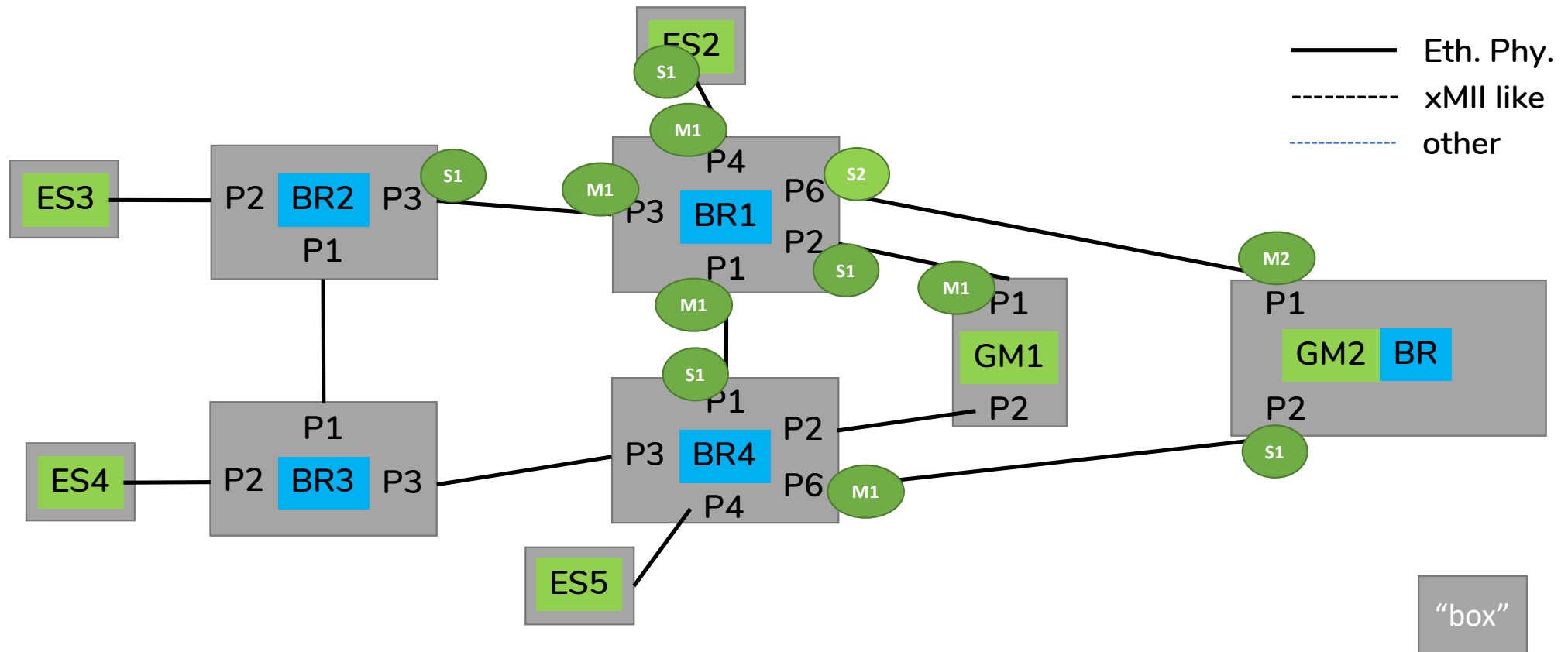


# Redundant TimeAware-Bridge



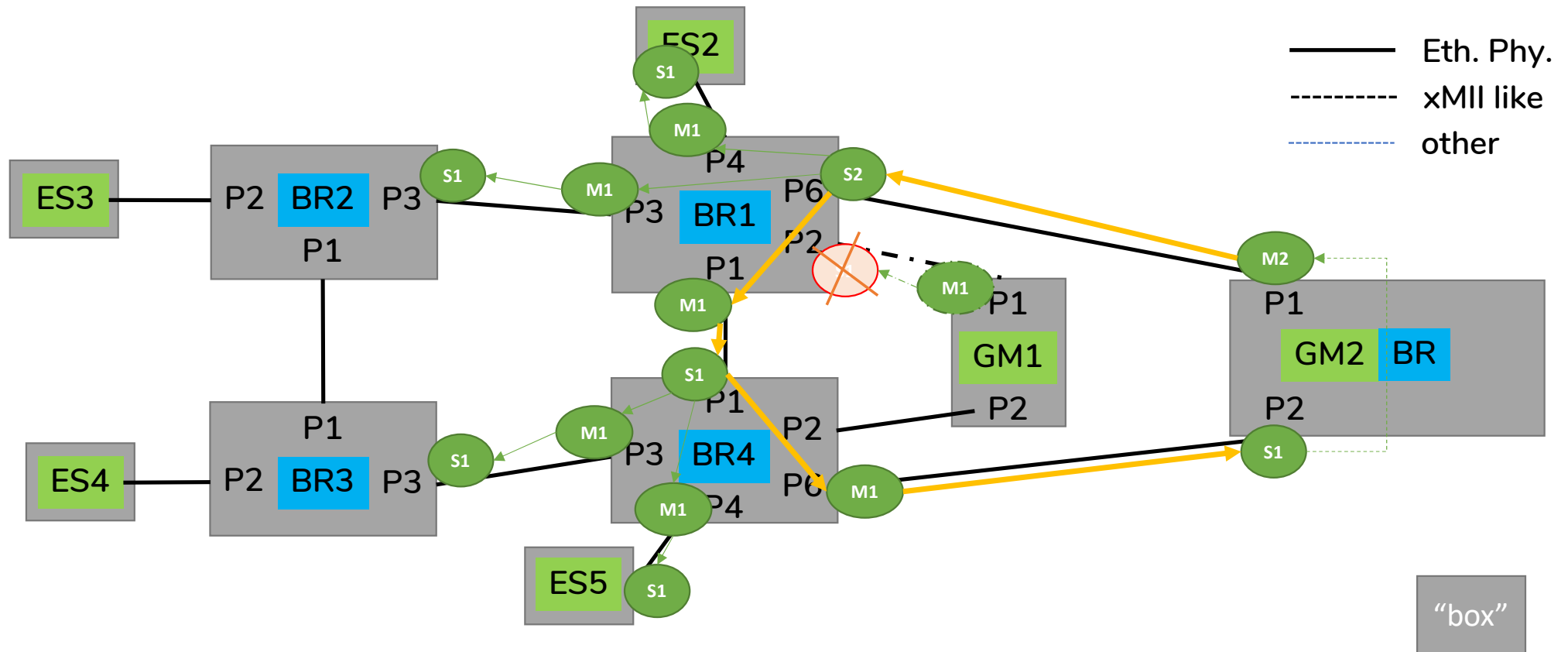


# Simplified Structure



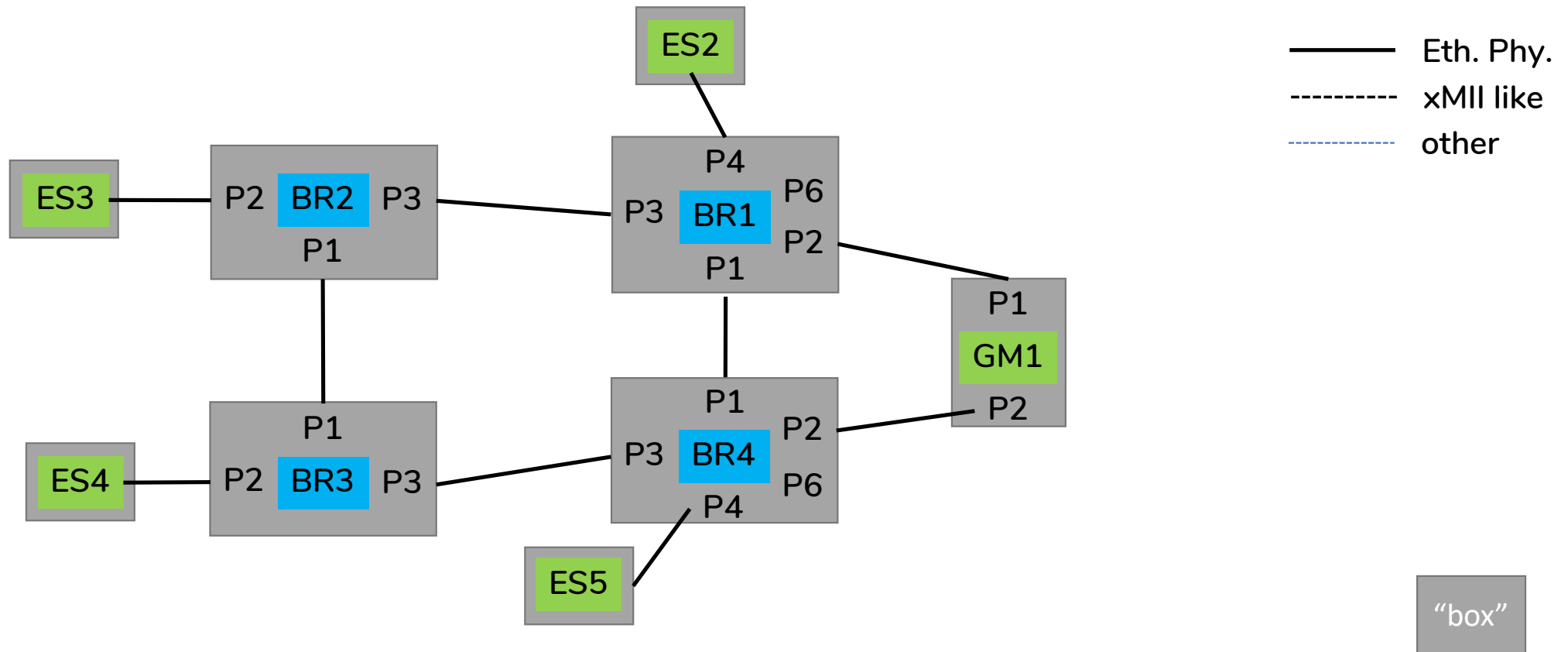


# Risk of Circular Time



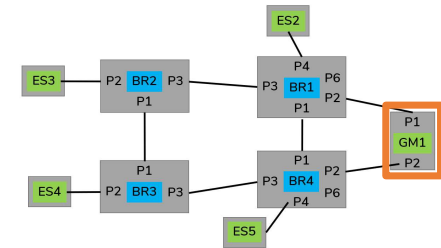


# Single GM Redundancy

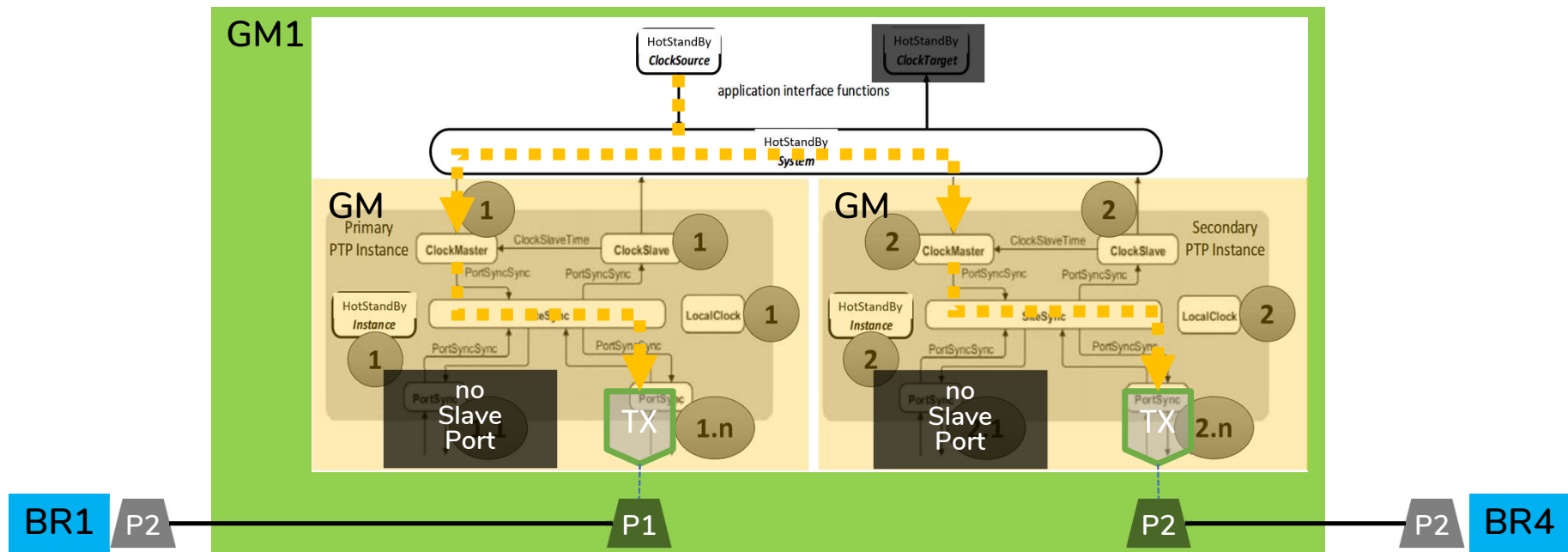




# Redundant GrandMaster

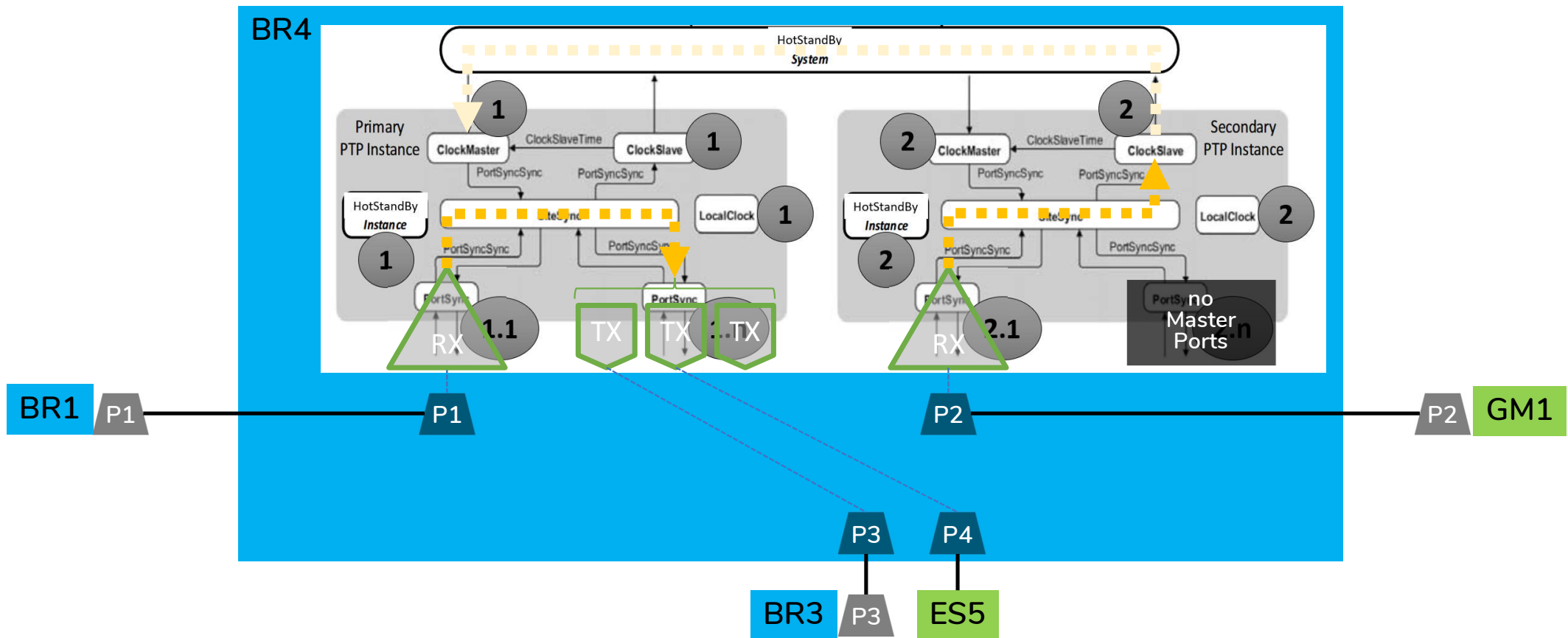
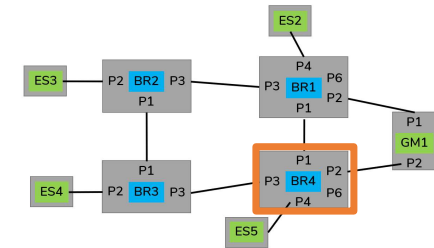


Is this covered by the current ASdm draft?



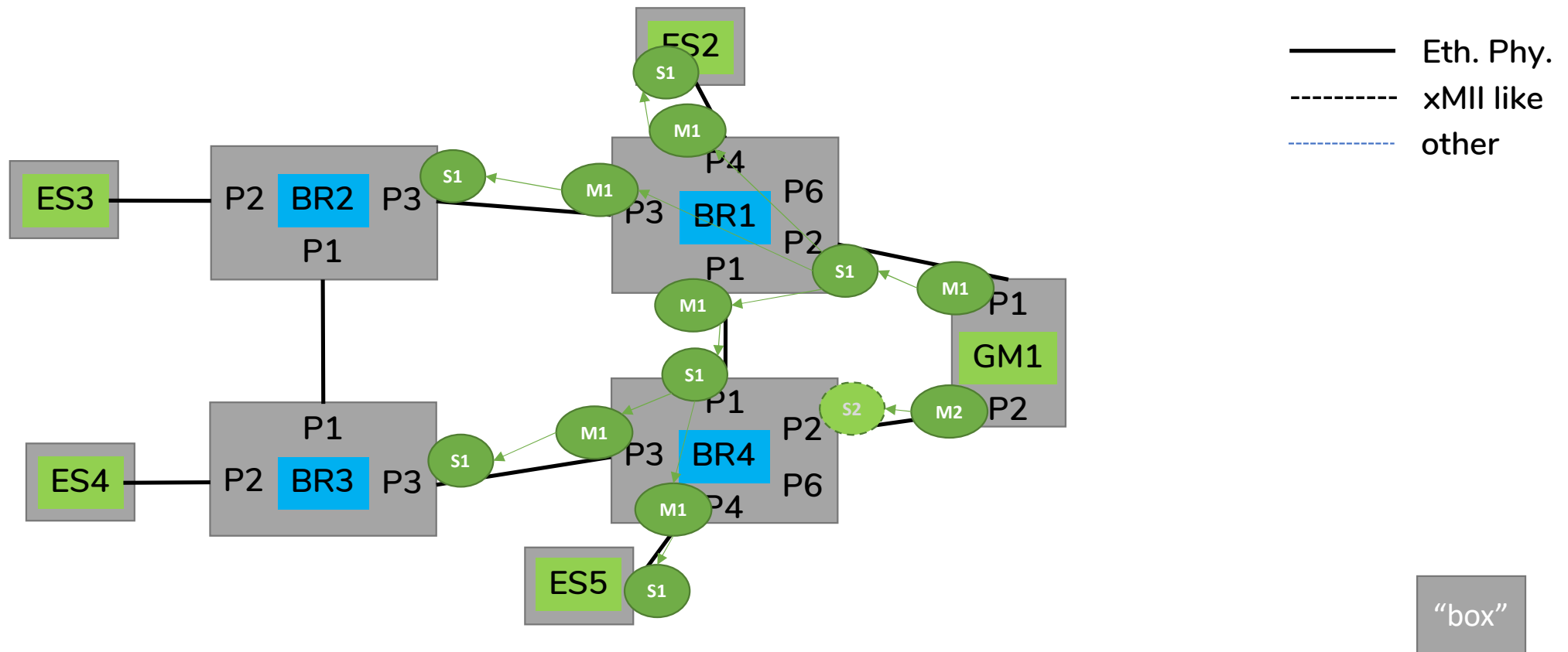


# Redundant TimeAware-Bridge



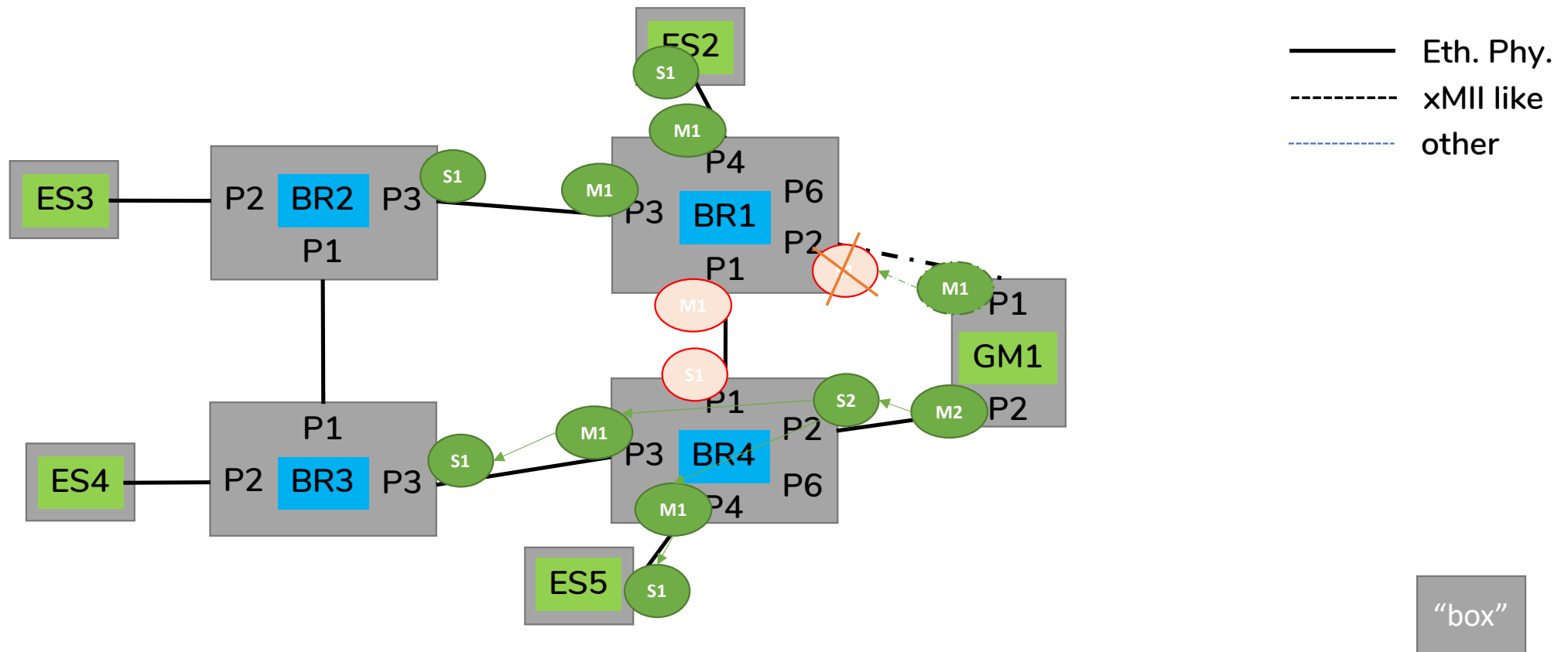


# Extended Redundancy





# Risk of Circular Time





The diagram illustrates a network topology with five main nodes represented as gray squares:

- Node 1 (Left):** Labeled "BR2" (blue) and "P1" (white). It has green square neighbors "ES3" (top-left) and "ES4" (bottom-left).
- Node 2 (Top-Right):** Labeled "BR1" (blue) and "P1" (white). It has green square neighbors "ES2" (top) and "ESS" (bottom).
- Node 3 (Bottom-Right):** Labeled "BR4" (blue) and "P1" (white). This node is enclosed in an orange rectangular border.
- Node 4 (Far Right):** Labeled "GM1" (green) and "P2" (white).
- Node 5 (Bottom-Center):** Labeled "BR3" (blue) and "P3" (white).

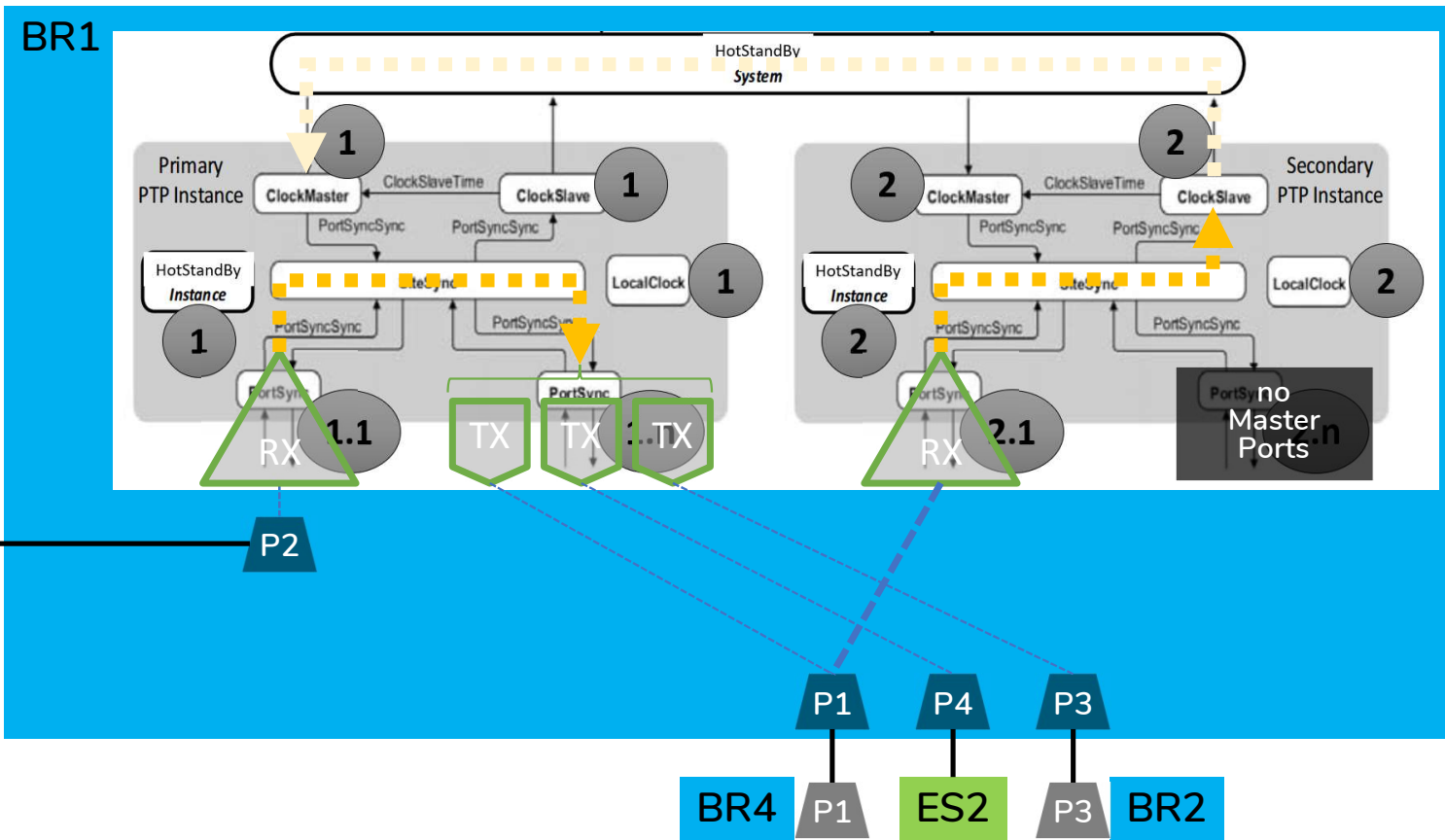
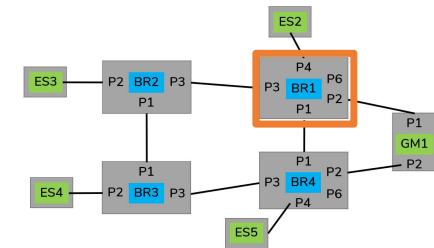
Connections between nodes are shown as black lines:

- Node 1 connects to Node 2 via link "P2".
- Node 1 connects to Node 5 via link "P3".
- Node 2 connects to Node 3 via link "P3".
- Node 2 connects to Node 4 via link "P6".
- Node 3 connects to Node 4 via link "P2".
- Node 3 connects to Node 5 via link "P3".
- Node 5 connects to Node 2 via link "P4".



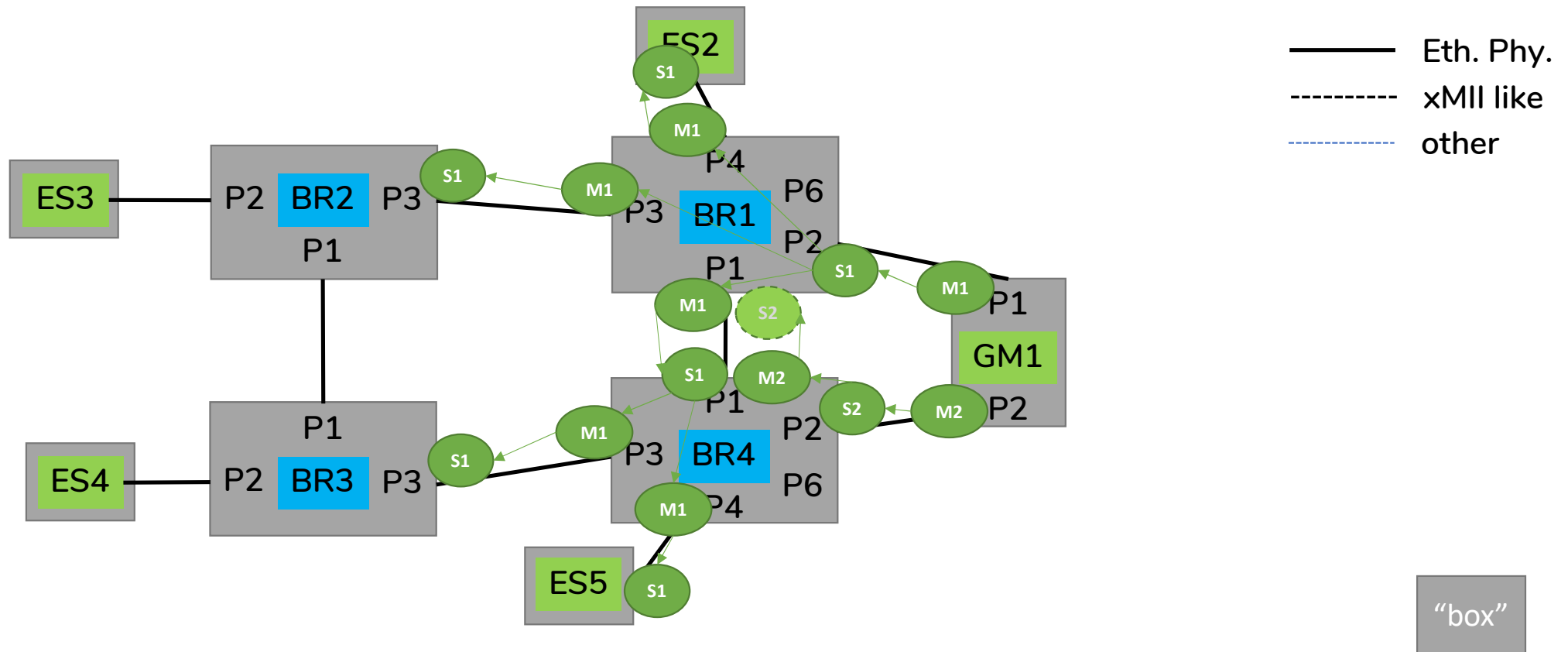


# Redundant TimeAware-Bridge



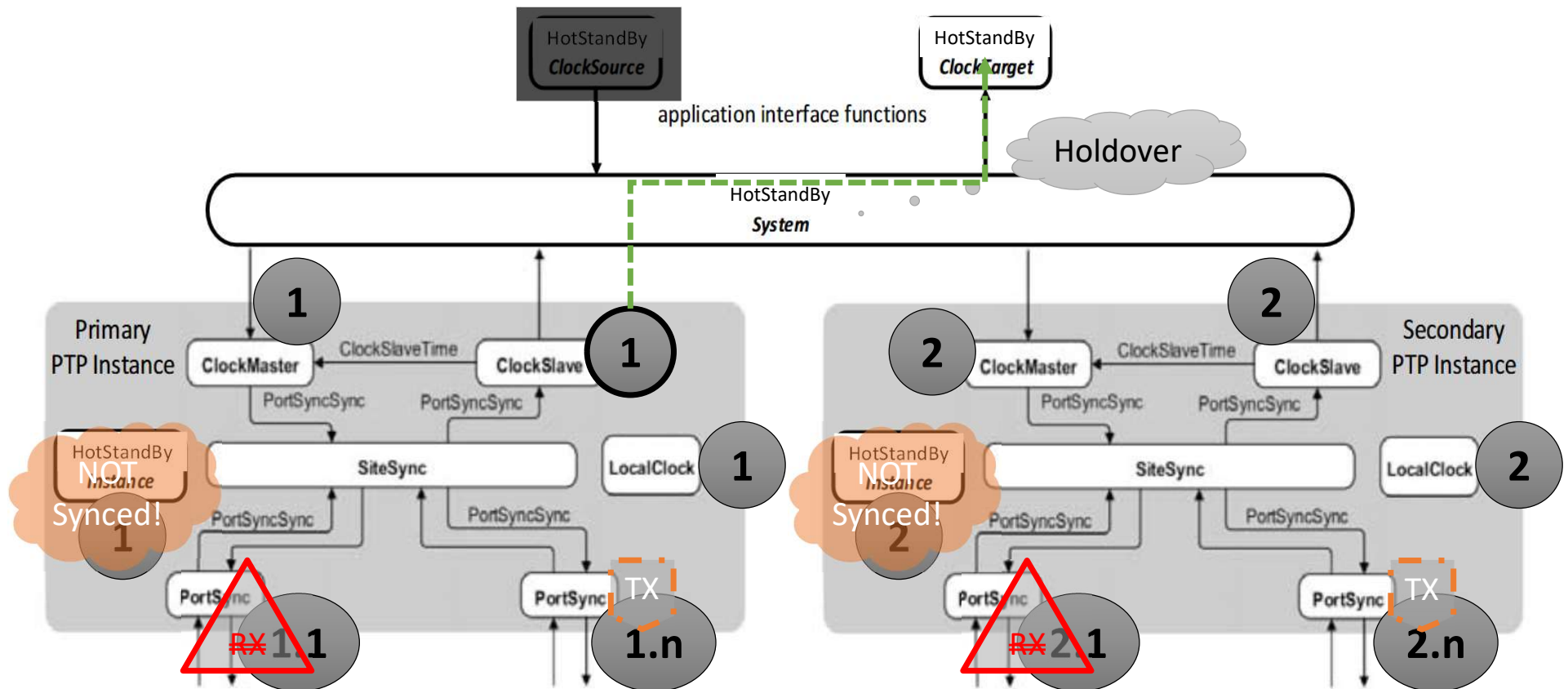


# Extended Redundancy





# Learn from: 17.12.4 Holdover





# 17.12.4 Holdover (Slave)

## 17.12.4 Holdover

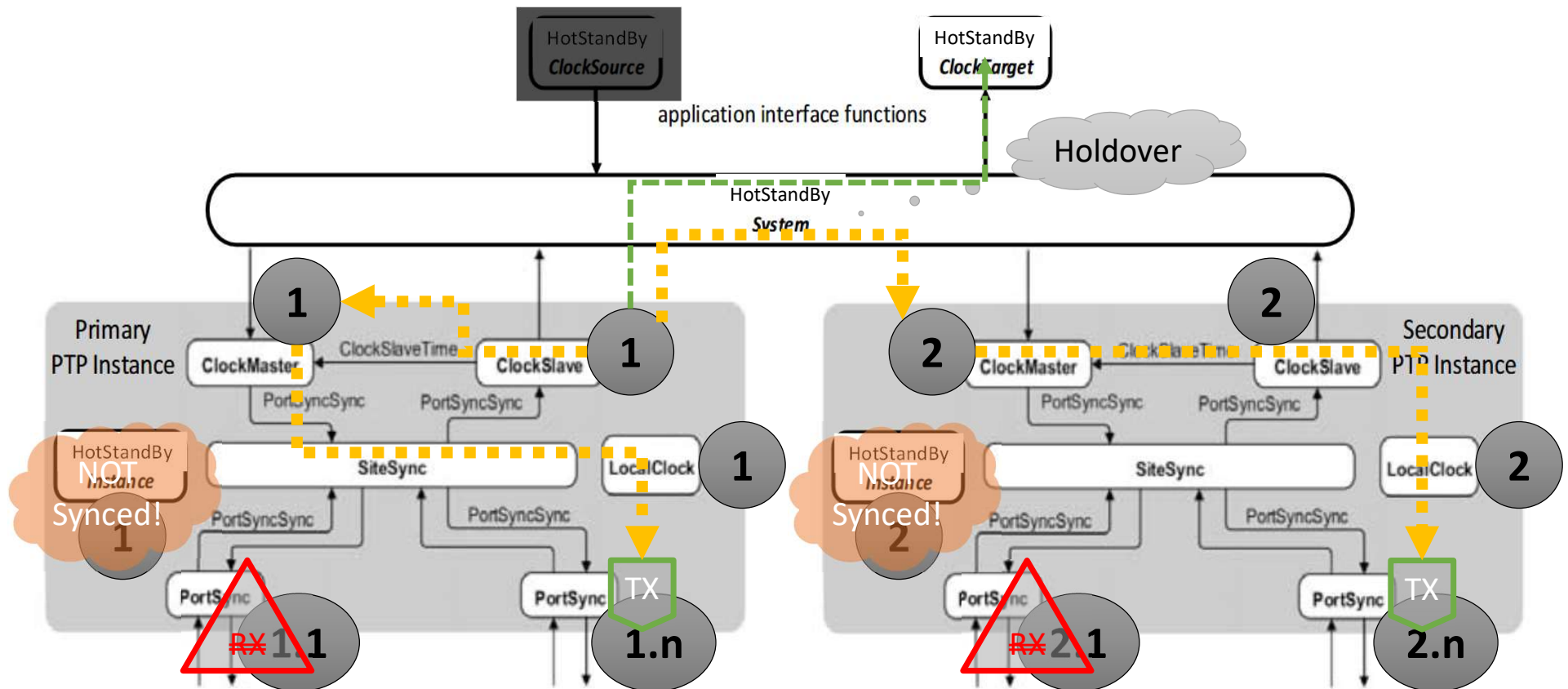
When `HotStandbySystemState` is `HOLDOVER`, the `HotStandbySystem` shall transfer phase and frequency from the `ClockSlave` of the primary PTP Instance to the `HotStandbyClockTarget`.

During `HOLDOVER` state, time synchronization performance is not required to meet the respective application or TSN profile requirements. Nevertheless, in order to mitigate drift, the primary PTP Instance should adjust phase/frequency of its local time using the data stored in `REDUNDANT` or `FAULT` state.

**<<Editor's note: It might be desired to use a term other than `HOLDOVER`, because often a distinction is made among holdover within limits, holdover not within limits, and free-run.>>**

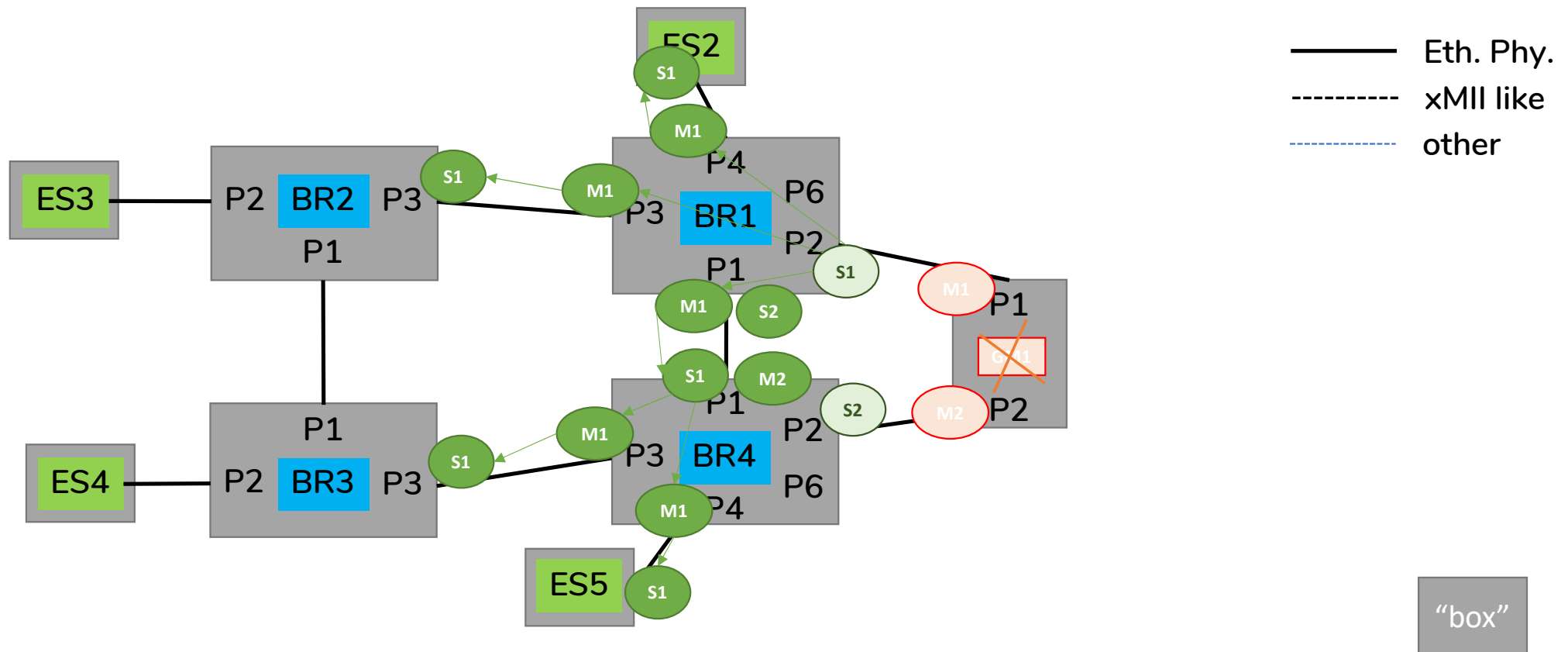


# Enable Holdover in Bridges





# Holdover in Bridge 1 and/or 4





An abstract network diagram composed of numerous blue dots (nodes) connected by thin blue lines (edges). The nodes are distributed across the upper right portion of the slide, with a higher density of connections on the right side, creating a complex, web-like structure that suggests a network topology.

# Open questions on the current ASdm Draft

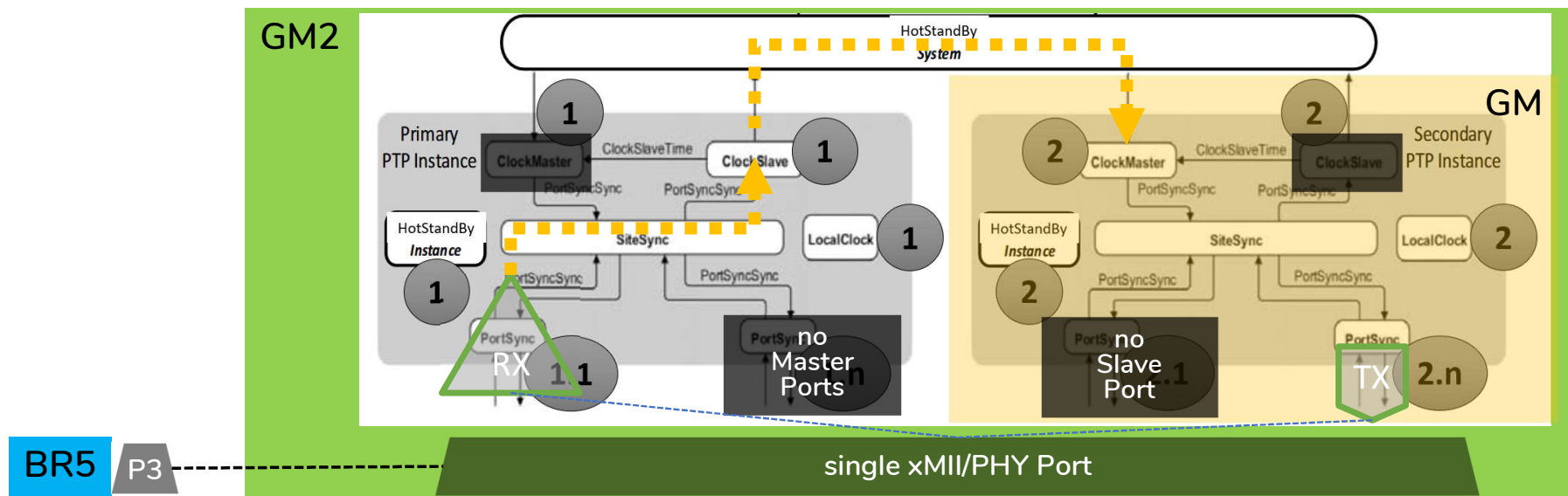


# GrandMaster in REDUNDANT?

## 17.4.3.2 Secondary grandmaster in REDUNDANT state

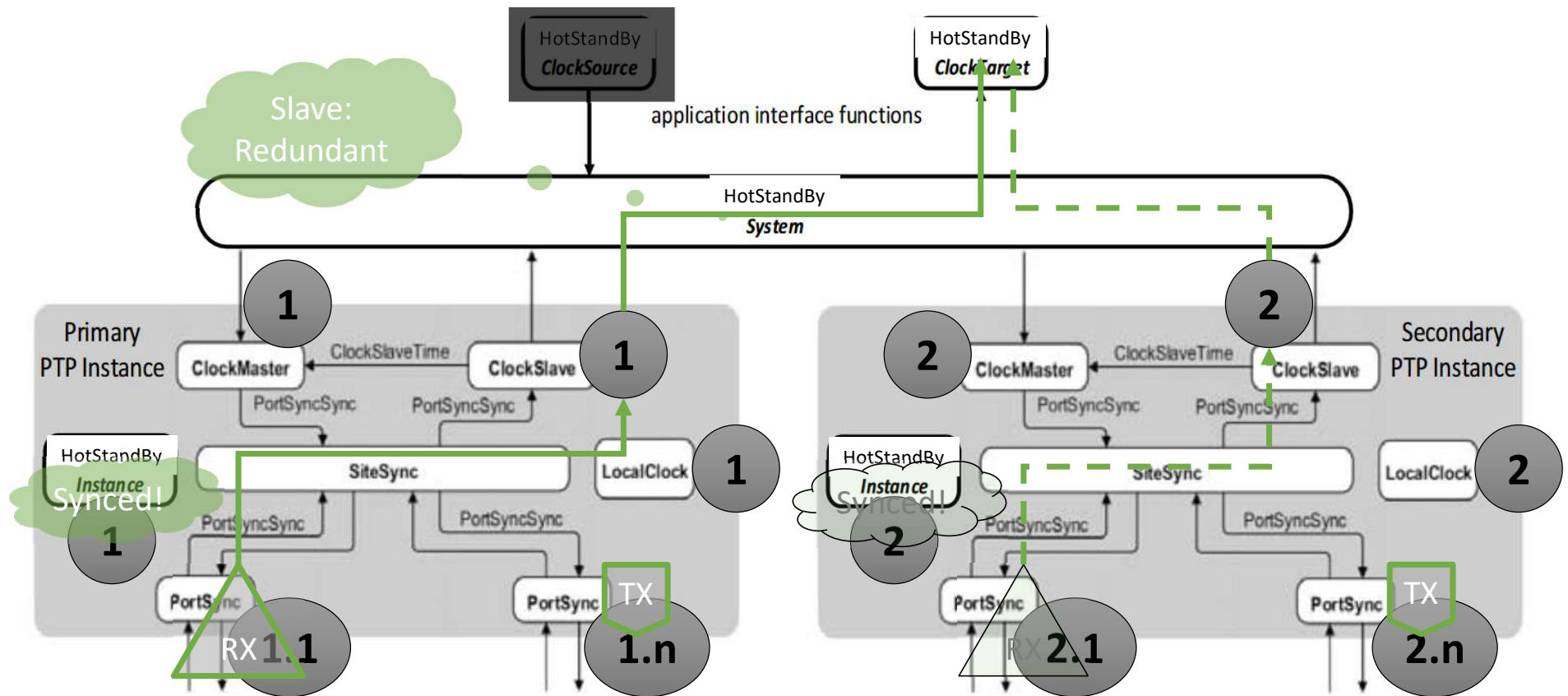
When the secondary PTP Instance is grandmaster, its **HotStandbySystemState** is REDUNDANT, and it receives time synchronization of the primary PTP Instance to the ClockMaster phase from the primary PTP Instance, the secondary grandmaster can maintain continuity in the event of a fault in the primary grandmaster.

- b) REDUNDANT: Both PTP Instances are synchronized according to the requirements of the respective application or TSN profile. Time synchronization is redundant.





# The two Time-Bases can not be compared in the current ASdm Draft.







THANK YOU

ETHERNOVIA

VIRTUALIZING VEHICLE COMMUNICATION