

EBU

OPERATING EUROVISION AND EURORADIO

TECHNICAL REVIEW

Using PTP for Time & Frequency in Broadcast Applications

Part 3: Network design for PTP

DECEMBER 2019

Thomas Kernen, Mellanox Technologies

Nikolaus Kerö, Oregano Systems

The main purpose of an EBU Technical Review is to critically examine new technologies or developments in media production or distribution. All Technical Reviews are reviewed by 1 (or more) technical experts at the EBU or externally and by the EBU Technical Editions Manager. Responsibility for the views expressed in this article rests solely with the author(s).

To access the full collection of our Technical Reviews, please see:
tech.ebu.ch/publications

If you are interested in submitting a topic for an EBU Technical Review, please contact: tech@ebu.ch

Abstract

As the broadcast industry continues to transform itself, leveraging new infrastructure and transport mediums for content production and delivery, the concept of the All-IP Studio is making inroads. Therefore, a significant focus has been placed on Ethernet and IP transport as the basis of these new systems, including the transport of Video, Audio and Ancillary data with the likes of the SMPTE ST 2110 standards. This multipart series covers a specific aspect of this ongoing transformation; the transport and use of phase and frequency for the purpose of timing over a converged IP network.

In this series we will start from the basics of the IEEE 1588 Precision Time Protocol (PTP) [1], its relationship to the broadcast industry and the related network requirements. As we build out from the basics, we will cover specific PTP design considerations, both from a network and end-node perspective. This series will then focus on more advanced topics, such as PTP redundancy, where we will drill deeper into the technical details.

1. Introduction

As described in the previous parts of this series, the IEEE 1588 Precision Time Protocol (PTP) plays a significant role in new systems designed around an IP centric transport model, especially those that are based on the SMPTE ST 2110 document series.

In this part we will focus on the IP network design specifics that need to be considered when planning for the transport of PTP over such networks. Since there are several different requirements linked to content production, size, location, distances and so forth, infrastructure topologies may vary significantly from one use case to another. Therefore, we will address the network requirements as a series of parameters to be taken into consideration during the design phase. This approach will allow for tuning the network model to best fulfil the use case requirements.

Ultimately, the design will be challenged by the technical and functional constraints imposed by the devices that are selected as the network endpoints.

2. PTP messaging path selection

Before we get into design details, there are several upfront questions that should be reflected upon. These will establish some of the base requirements for the PTP transport design process.

How are PTP messages distributed from the elected Grand Master (and the passive/backup GMs) to all the devices on the network?

The GM redundancy model and the distribution of the PTP messages across the network do play a significant role in how to architect one of the key PTP components; even more so in redundant networks prevalent in broadcasting environments.

Two schools of thought tend to prevail here: The first is to have a single active GM for all network(s) so that PTP devices, wherever they may be connected, including those on a fully redundant A/B (Red/Blue) network, will always be locked to the same GM. Other (auxiliary) GMs will remain in passive state unless the current GM fails. Typically, what is required is a network “feeder” layer to distribute the GM PTP messages to multiple networks and/or different network segments. This layer should be solely dedicated to the transport of PTP messages between the different network fabrics.

The second approach independently locks multiple GMs to a common primary reference, typically a GNSS (Global Navigation Satellite System) source such as GPS or Galileo, and in some cases to multiple sources at the same time for better accuracy and stability. Consequently, different parts of the networks may be locked to varying GMs. If all GMs derive their time information from a common GNSS feed, the assumption is that in a well-designed system the relative time difference of all PTP Slaves will remain well below 50ns, regardless of the GM they are locked to. Figure 1 shows a long-term measurement of the relative offset between two PTP Slaves each locked to a different GM. The data was derived by comparing the 1PPS signals of both devices.

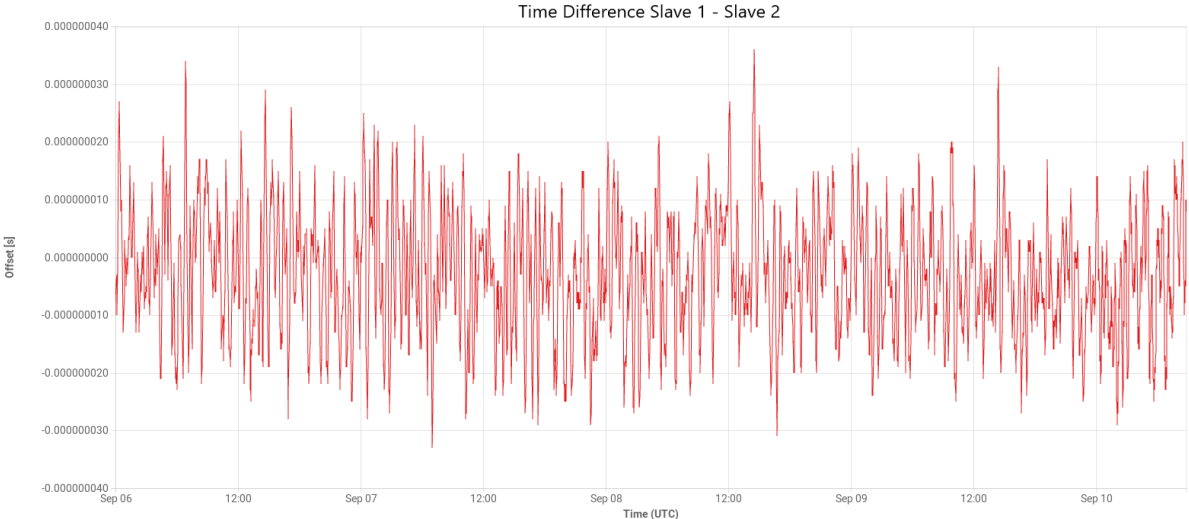


Figure 1: Comparison of 1pps plot from 2 separate GMs

Since the GMs are traced back to the same primary reference, their time information can be declared with the “traceable” flag in the PTP messages, which can be carried over into the corresponding SDP file generated by a sender. Phase/frequency alignment may therefore be adequate for the majority of media applications.

In either case, continuous in-band and out-of-band monitoring of all GMs is required to ensure that the overall PTP subsystem is operating within expected boundaries. We will further develop the topic of PTP monitoring in a subsequent part of this series.

How do I configure my Grand Master hierarchy?

The (Grand)Master election is based on the Best Master Clock Algorithm (BMCA) which will elect according to the group of parameters in the following precedence. They are announced by any PTP device that takes part in the election process:

Priority1, Clock Class, Clock Accuracy, Clock Variance, Priority2 and Source PortID.

Priority1 and Priority2 are user settable fields; the values of other fields are derived from the reference clock that the PTP device is currently using and the MAC address of the interface.

In most cases, the GM redundancy scheme is something that would be planned and accounted for, hence using the Priority1 and Priority2 fields allows the creation of a specific hierarchy amongst the candidate devices. An assumption is that all candidate GMs are equal with respect to the quality of the local oscillator and the performance of the timestamping hardware and are locked to a primary reference that is common to all. In so doing, the Clock Class, Accuracy and, most likely, Variance will be identical.

Priority1 is the “sledgehammer” field and should be used with great caution. Configuring this field will override any Clock quality changes in the GM, thereby impacting performance in the case that the GM is degraded due to loss of lock to the primary reference, resulting in reduced Accuracy or increased Variance. Therefore, using this field for setting all candidate GMs to the same value is a reasonable approach and avoids causing hierarchy issues.

Priority2 allows for a more granular approach such that, once the tie between the different GM candidates is down to a specific denominator, the use of this field controls the specific order in which GMs should be elected. This removes election uncertainty between GMs assuming no other external factors are involved.

Is the primary intent to run PTP traffic in-line with essence or as a separate service?

The question may appear odd, but if you think about it, this is akin to “Shall I sync from SDI or Black & Burst?” There is no obligation to have PTP traffic in-line with essence. The synchronisation of the PTP stack in the endpoint can be done on any interface that can process PTP event messages using hardware timestamping and therefore drive the clock of that device with sufficient accuracy. From a PTP stack perspective, the interface choice makes no difference. It is purely an administrative design consideration. This choice also dictates which interfaces PTP messages will be flowing over: Data (essence) vs. management, or a dedicated PTP network. In

any case, a well-designed NIC with hardware PTP timestamping, as discussed in part 2 of this series, is strongly recommended to reach the desired accuracy.

3. PTP “aware” vs. “non-aware” requirements

In part 1 of the series, we discussed PTP-aware network devices, and in part 2 we further explained how these devices function, specifically the Transparent Clocks (TC) and Boundary Clocks (BC). In the same manner that hardware PTP timestamping improves the accuracy for a PTP node running as an endpoint, the BCs and TCs perform the same task at the network level. Additionally, in the case of BCs, they provide the capability of segmenting the PTP traffic on a per-port basis, thereby isolating the majority of PTP messages to the devices connected to that physical port. Other than this specific ability, there are no drawbacks to using TCs vs. BCs and combining them in a network design. Either of these devices will increase the accuracy vs. PTP “non-aware” switches.

The impact of PTP “non-aware” switches in network environments that transport media essence has been well documented [\[2\]](#), explaining the significant increase in offset that can occur due to packet congestion and loss. As the transition to an All-IP system rolls out and the dependency on PTP increases, the use of PTP “non-aware” devices should be prevented or restricted to very specific setups where the impact of PTP loss or inaccuracies is well understood and accounted for.

4. Network architecture

As mentioned in the introduction, the design of the IP network is often influenced by factors such as space, power, cooling, port density, distance between devices and so on. Some factors may be temporary, others may be long term. They may either be planned due to physical constraints (layout) or logical ones (product choices). No matter the technical inputs you are working with, or around, you will most likely need to compromise.

Many discussions start with “should I build a modular or a distributed network?” In other terms, we are talking about a large centralised IP switching fabric with a high port count or a distributed “spine-leaf” fabric that is typically made up of fewer ports on smaller devices interconnected to a series of core devices to exchange traffic.

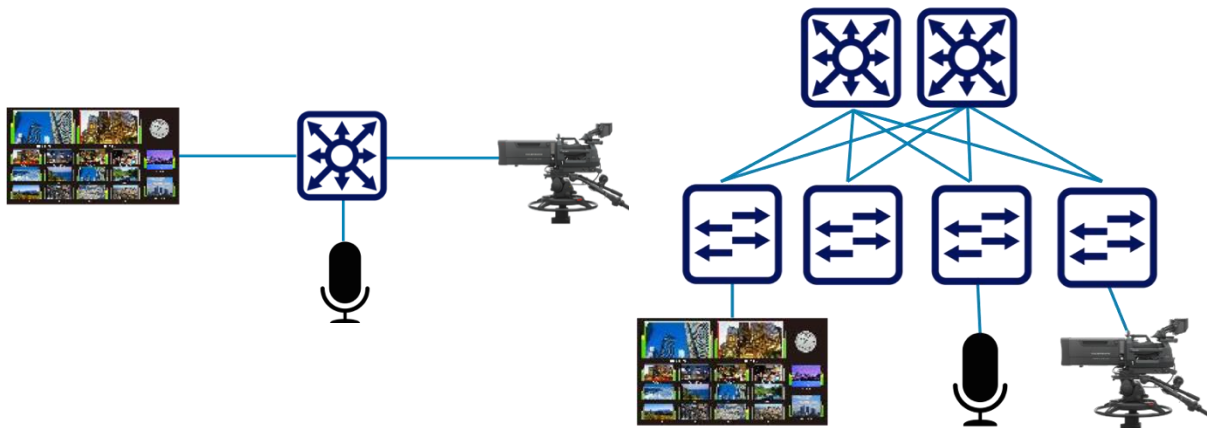


Figure 2: Modular vs. Spine/Leaf network design

Whilst PTP isn't the main driver in the definition of the network topology, it must nevertheless be taken account of. A significant impact on design and performance may occur if it is overlooked. In all cases, the network must be engineered to avoid oversubscription on any of the links. This is essential not only because of the impact on the essence being transported, but also to avoid PTP messages from being delayed (jitter) or discarded (loss). These would introduce asymmetries in the PTP path and cause accuracy issues that the PTP stack would need to attempt to compensate for.

The capacity for a switch to handle many PTP messages originating from multiple devices per port may be required, especially in topologies where BCs are used on the core/spine and TCs or PTP non-aware switches are used downstream. In such cases the TC or PTP non-aware switches will forward all the PTP messages to the upstream device (assuming it is running the MASTER role), hence the fan-out may be of the order of 10s, 100s or more PTP devices connected to a single switch port. The processing of those PTP messages should be well understood when designing the network since you may inadvertently cause a denial of service against the MASTER port and possibly the whole BC or GM, if there are too many PTP messages to be processed. This same issue may occur with any PTP device that is listening to the PTP traffic.

Another area that is often discussed is around the type of interface, typically Layer 2 switched or Layer 3 routed ports. From a PTP perspective, these are treated identically. The mode in which the interface on the switch is operating shouldn't impact PTP. It is treated as a PTP port irrespective of the L2 or L3 configuration.

Some PTP implementations on switches will allow for PTP clock separation not only at a physical interface level but also at a logical level. These are implementation dependent and allow for additional granularity in configuring PTP:

- VLAN: PTP messages may be sent as untagged, fixed to a single VLAN ID or flexible whereby you may select which VLANs have a PTP clock enabled, such as part of an 802.1q trunked interface.

- Link Aggregation / Port Channel: Other than the 2 interface types mentioned above, many implementations allow for running PTP across aggregation links. It may be a case of configuring it for the aggregation link or the individual interface in the aggregation link, providing additional granularity towards devices that may or may not work well when running PTP over multiple interfaces.
- Virtual Routing and Forwarding (VRF): Some implementations allow for running PTP in a specific VRF context, as a means of isolating PTP traffic within a specific Layer 3 part of the network, akin to what can be done by specifying specific VLAN IDs at Layer 2.

These features may be of use as part of the design strategy for the deployment of PTP to further control which interfaces, endpoints and routing context have visibility of the PTP messages.

5. Other design considerations that may impact PTP

Where should I connect my GM(s)?

This question should be rephrased as: “Does the location at which my GM(s) are connected impact the accuracy or redundancy of my PTP infrastructure?” The location at which GM and candidate GM(s) are connected (either to the spine/core or leaf switch(es)) doesn’t change the way PTP messages are propagated.

Considerations are the cost of the port(s) used for the connection; 1, 10, 25 or 100 GE ports have different cost profiles and “footprint value” on the switches. Modern BC implementations cause limited oscillation, and in a centralised core or spine/leaf architecture, every PTP node is only a couple of hops away. The impact on SMPTE 2059-2 [3] based systems is therefore minimal.

How can I reduce the load of PTP messages on the network?

By default, the 2059-2 profile sends all PTP messages as multicast. This implies that every PTP node must listen to all the multicast messages sent, whether relevant to it or not. Therefore, the load on a shared segment where multiple PTP devices are communicating increases as the number of connected devices grows. This can cause issues with the PTP message processing on endpoints, due to having to listen and process all PTP messages before discarding the non-relevant messages.

By using a “mixed-mode”, sometimes also called a “hybrid mode” of operation, whereby messages from a PTP node running as a Slave to the Master are sent using unicast and the respective responses from the Master to the Slave are also unicast as per the IEEE 1588 standard, the overall PTP load on the network is reduced to a minimum. The Master port will process the same number of messages whereas the Slaves will only process the multicast announce, sync and management messages, and the unicast delay response messages from its own unicast delay requests.

How about PTP Management TLV messages?

The acknowledgement (Ack) messages of those PTP management TLV messages should be sent as unicast from the PTP node to the GM that issued the message as per ST 2059-2. This will prevent multicast Ack storms that propagate across the network and may cause a snowball effect of responses to responses, whatever the response status may be.

How can I prevent misconfigured PTP devices from impacting PTP stability?

Most network vendors have a feature that allows the user to configure the PTP ports on the switch to be “forced” to a Master-only state. By doing so, even if PTP devices connected to that port send announce messages that could trigger a BMCA and Master election, those are ignored.

Another feature that is part of the IEEE 1588 standard is called AMT (Acceptable Master Table), which provides a whitelist of ClockIDs that a PTP device may listen to. The whitelist would typically contain the ClockIDs of the GMs and candidate GMs and is applied to end nodes and/or network switches that support this.

Can I run PTP over a Wide Area Network (WAN)?

As highlighted earlier, the accuracy and stability of PTP is dependent on PTP “aware” devices that perform accurate hardware timestamping in order to meet the requirements. As such, networks where the transport is built using an overlay model such as MPLS, VxLAN or other techniques that obfuscate the physical layer from the transport layer, will prevent hardware timestamping at each physical node. Therefore, PTP messages carried across these links will not be time stamped as they would on the local network infrastructure. Unless you have physical access to the Layer 1 transport such as optical wavelengths or dark fibre, you will not benefit from such a transport for PTP. This shouldn't be of a concern in most cases, as we explained above, if the primary reference clocks are common and traceable, you should be able to use a GM per location whilst still maintaining accurate and stable timing.

Running PTP over a WAN has been intensively investigated in other application domains such as the telecom industry. To reach accuracies better than 1 μ s, extensive linear and non-linear filtering combined with high message rates, is mandatory. These implementations require very long settling times as well as additional means to counteract asymmetries that are common in such environments.



6. Coming up next

In the next part of this series, we will delve deeper into why PTP via ST 2059-2 is a key component of the ST 2110 system and media flows.

7. References

- [1] IEEE 1588, "IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems," IEEE Instrumentation and Measurement Society, Piscataway, NJ, 2008.
- [2] "Analysis of Precision Time Protocol (PTP) Locking Time on Non-PTP Networks for Generator Locking over IP," SMPTE Motion Imaging Journal, Volume 123, Issue 2, March 2014.
- [3] SMPTE ST 2059-2:2015, SMPTE Profile for Use of IEEE-1588 Precision Time Protocol in Professional Broadcast Applications, Approved March 19th, 2015.

9. Author(s) biographies

	<p>Thomas Kernen is Senior Staff Architect with Mellanox Technologies Ltd. Prior to joining Mellanox, he spent over 20 years in the IP industry, including driving Cisco's entry into live media production, co-founding Internet Service Providers, Telecom carriers, and architecting Fibre to the Home networks. He has authored over 20 publications in leading journals. He holds six patents that cover both network and video coding optimizations for media transport and delivery.</p> <p>His current interests include defining architectures for transforming the broadcast industry to an All-IP infrastructure. Thomas is also a member of the IEEE Communications and Broadcast Societies. He currently serves as the Co-Chair of the Society of Motion Pictures and Television Engineers (SMPTE) 32NF Committee. He is also a frequent speaker at leading events, such as the SMPTE Annual Technical Conference, the National Association of Broadcasters, IBC, and the European Broadcasting Union Network Technology Seminar.</p>
	<p>After receiving a master's degree in Communication Engineering with distinction from the Vienna University of Technology, Nikolaus led the ASIC design division at the university's Institute of Industrial Electronics, successfully managing numerous research projects and industry collaborations. His research activities centred on distributed systems design, especially highly accurate and fault-tolerant clock synchronization. In 2001 he co-founded Oregano Systems Design & Consulting Ltd. as a university spin-off.</p> <p>While offering embedded systems design services to customers, Oregano successfully transferred Nick's research results into a complete product suite for highly accurate clock synchronization under the brand name syn1588®, for which Nick manages both development and marketing. He is an active member of the IEEE1588 standardization committee and the SMPTE 32NF standards group and holds frequent seminars on clock synchronization for both industry and academia.</p>

Published by the European Broadcasting Union, Geneva, Switzerland

ISSN: 1609-1469

Editor-in-Chief: Patrick Wauthier

E-mail: wauthier@ebu.ch

Responsibility for views expressed in this article rests solely with the author(s).