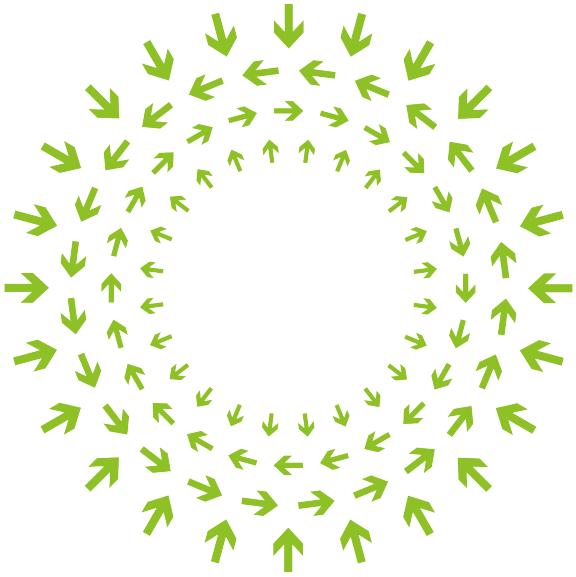

OPEN

Compute Project

Data Center PTP Profile

(Initial draft for review)

<Revision>

Author: <Primary>

Author: <secondary. Delete if unnecessary>

Table of Contents

TABLE OF CONTENTS 2

1. LICENSE (OCP CLA OPTION) 4

1. LICENSE (OWF OPTION) 5

1 SCOPE & OVERVIEW 6

2 TERMINOLOGY 7

3 PTP PROFILE DEFINITION 7

4 ADDITIONAL TOPICS BEYOND PTP 7

5 REFERENCE MODELS 8

5.1 Model 1 – TC Model 8

5.2 Model 2 – TC+BC Model 9

6 MODEL 1 - ADDITIONAL REQUIREMENTS 10

7 PTP PROFILE 11

7.1 PTP Profile 11

7.2 Clock Types 11

7.3 Message Types 11

7.4 Transport mechanisms required, permitted, or prohibited 12

7.5 Clock identity 12

7.6 Path delay Measurement Mechanism 12

7.7 Class of Service 12

7.8 Profile Isolation and Domain Number 12

7.9 One-step and two-step operation 13

7.10 End-to-End TC with two-step operation 13

7.11 PTP message rate 14

7.12 PTP inter-message interval 14

7.13 Unicast Communication 15

7.13.1 Unicast Discovery 16

7.13.2 Unicast Negotiation 19

7.14 Best Master Clock Algorithm and clock attributes 22

7.15 Network Limits and Error Budget for Model 1 24

8 REFERENCES 26

1. License **(OCP CLA Option)**

Contributions to this Specification are made under the terms and conditions set forth in Open Compute Project Contribution License Agreement ("OCP CLA") ("Contribution License") by:

[Contributor Name(s) or Company name(s)]

Usage of this Specification is governed by the terms and conditions set forth in **[select one: Open Compute Project Hardware License – Permissive ("OCPHL Permissive"), Open Compute Project Hardware License – Copyleft ("OCPHL Reciprocal")] ("Specification License").**

Note: The following clarifications, which distinguish technology licensed in the Contribution License and/or Specification License from those technologies merely referenced (but not licensed), were accepted by the Incubation Committee of the OCP:

[insert "None" or a description of the applicable clarifications].

NOTWITHSTANDING THE FOREGOING LICENSES, THIS SPECIFICATION IS PROVIDED BY OCP "AS IS" AND OCP EXPRESSLY DISCLAIMS ANY WARRANTIES (EXPRESS, IMPLIED, OR OTHERWISE), INCLUDING IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, FITNESS FOR A PARTICULAR PURPOSE, OR TITLE, RELATED TO THE SPECIFICATION. NOTICE IS HEREBY GIVEN, THAT OTHER RIGHTS NOT GRANTED AS SET FORTH ABOVE, INCLUDING WITHOUT LIMITATION, RIGHTS OF THIRD PARTIES WHO DID NOT EXECUTE THE ABOVE LICENSES, MAY BE IMPLICATED BY THE IMPLEMENTATION OF OR COMPLIANCE WITH THIS SPECIFICATION. OCP IS NOT RESPONSIBLE FOR IDENTIFYING RIGHTS FOR WHICH A LICENSE MAY BE REQUIRED IN ORDER TO IMPLEMENT THIS SPECIFICATION. THE ENTIRE RISK AS TO IMPLEMENTING OR OTHERWISE USING THE SPECIFICATION IS ASSUMED BY YOU. IN NO EVENT WILL OCP BE LIABLE TO YOU FOR ANY MONETARY DAMAGES WITH RESPECT TO ANY CLAIMS RELATED TO, OR ARISING OUT OF YOUR USE OF THIS SPECIFICATION, INCLUDING BUT NOT LIMITED TO ANY LIABILITY FOR LOST PROFITS OR ANY CONSEQUENTIAL, INCIDENTAL, INDIRECT, SPECIAL OR PUNITIVE DAMAGES OF ANY CHARACTER FROM ANY CAUSES OF ACTION OF ANY KIND WITH RESPECT TO THIS SPECIFICATION, WHETHER BASED ON BREACH OF CONTRACT, TORT (INCLUDING NEGLIGENCE), OR OTHERWISE, AND EVEN IF OCP HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1. License (OWF option)

Contributions to this Specification are made under the terms and conditions set forth in Open Web Foundation Contributor License Agreement ("OWF CLA 1.0") ("Contribution License") by:

[Contributor Name(s) or Company name(s)]

Usage of this Specification is governed by the terms and conditions set forth in the Open Web Foundation Final Specification Agreement ("OWFa 1.0").

Note: The following clarifications, which distinguish technology licensed in the Contribution License and/or Specification License from those technologies merely referenced (but not licensed), were accepted by the Incubation Committee of the OCP:

[insert "None" or a description of the applicable clarifications].

NOTWITHSTANDING THE FOREGOING LICENSES, THIS SPECIFICATION IS PROVIDED BY OCP "AS IS" AND OCP EXPRESSLY DISCLAIMS ANY WARRANTIES (EXPRESS, IMPLIED, OR OTHERWISE), INCLUDING IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, FITNESS FOR A PARTICULAR PURPOSE, OR TITLE, RELATED TO THE SPECIFICATION. NOTICE IS HEREBY GIVEN, THAT OTHER RIGHTS NOT GRANTED AS SET FORTH ABOVE, INCLUDING WITHOUT LIMITATION, RIGHTS OF THIRD PARTIES WHO DID NOT EXECUTE THE ABOVE LICENSES, MAY BE IMPLICATED BY THE IMPLEMENTATION OF OR COMPLIANCE WITH THIS SPECIFICATION. OCP IS NOT RESPONSIBLE FOR IDENTIFYING RIGHTS FOR WHICH A LICENSE MAY BE REQUIRED IN ORDER TO IMPLEMENT THIS SPECIFICATION. THE ENTIRE RISK AS TO IMPLEMENTING OR OTHERWISE USING THE SPECIFICATION IS ASSUMED BY YOU. IN NO EVENT WILL OCP BE LIABLE TO YOU FOR ANY MONETARY DAMAGES WITH RESPECT TO ANY CLAIMS RELATED TO, OR ARISING OUT OF YOUR USE OF THIS SPECIFICATION, INCLUDING BUT NOT LIMITED TO ANY LIABILITY FOR LOST PROFITS OR ANY CONSEQUENTIAL, INCIDENTAL, INDIRECT, SPECIAL OR PUNITIVE DAMAGES OF ANY CHARACTER FROM ANY CAUSES OF ACTION OF ANY KIND WITH RESPECT TO THIS SPECIFICATION, WHETHER BASED ON BREACH OF CONTRACT, TORT (INCLUDING NEGLIGENCE), OR OTHERWISE, AND EVEN IF OCP HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1 Scope & Overview

Scope:

This document defines a PTP profile for time-sensitive applications within a data center environment. The document is developed within the OCP Timing Appliances Project [1]. The PTP profile is based on IEEE Std IEEE1588™-2019 [2]. When applicable, the profile also references and reuses information from other PTP profiles or other industry specifications. The document provides a set of requirements for implementing, deploying and operating timing appliances within a data center. A timing appliance is an element that is PTP-capable or PTP-aware (eg., NIC card, switch/router, software module, timing card, monitoring device, etc.).

Overview:

Time is a key element to get the highest efficiency in a distributed system. The performance of a distributed system depends in part on the level of synchronization between the elements. Several industries such as telecom, power, industrial, automotive, professional audio and video have embraced the need for highly accurate and reliable distribution and synchronization of time across packet networks. Although the use case scenario for each of the industries is different, they all share one common thing and that is, time synchronization. Each use case scenario defines a set of requirements and configurations. These configurations are specified in a ‘PTP profile’. This document defines a PTP profile for data center applications and data center network infrastructure. The profile specifies the set of PTP features and attribute values applicable to a PTP instance that operates in a single device (eg., such as a switch, router, server) and within exactly one PTP domain. Additionally, this specification also addresses additional requirements that are outside the definition of a PTP profile.

2 Terminology

Note: The IEEE1588 committee is starting a project to recommend alternative terminology that is more inclusive to some of the terminology such as master/slave currently used in IEEE Std 1588-2019. The project has not yet decided on the alternative terminology. It would be desirable to use the new terms chosen by the IEEE1588 committee. Unfortunately, new terms in IEEE won't be decided and published immediately.

Note: The contributors of this recommendation will do their best to minimize any non-inclusive terms. However, there are instances where this might not easily be possible (such as the name of a parameter or a code variable).

3 PTP Profile Definition

A PTP profile is "a document, or a portion of a document, specifying the set of PTP features and attribute values applicable to a PTP instance, and written by an organization following the specification of IEEE Std IEEE1588-2019. The profile allows organizations to specify selections of attribute values and optional features of PTP for the purpose of meeting requirements of a particular application. A PTP profile applicable to data center is defined in this document.

A PTP profile is a set of required options, prohibited options, and the ranges and defaults of configurable attributes. A profile should define, for example, the following:

- Best master clock algorithm options
- Configuration management options
- Path delay measurement option (delay request-response or peer delay)
- Range and default values of all configurable attributes and dataset members
- PTP Instances types
- Options required, permitted, prohibited
- Uncertainty specifications
- Transport mechanisms required, permitted, or prohibited
- If relevant, the value of the observation interval τ used for PTP Variance measurements.

4 Additional topics beyond PTP

Additional requirements or guidelines outside the definition of a PTP profile can also be specified. Some of those are:

- End-to-end network architecture
- Application requirements, time error budget
- End-to-end network limits and measurable metrics (e.g., max|TE|, MTIE, TDEV, etc.)
- Failure scenarios and holdover accuracy
- Syntonization
- Classes of service and prioritization of PTP messages
- Operations and management of synchronization tree, data models
- Measurement interfaces
- Security

- Holdover

5 Reference Models

The figures in this section show two reference models. Each model consists of three layers. The time reference layer consists primarily of sourcing a time reference and the PTP grandmaster (GM) functionality. The network fabric layer consists of set of network elements that support PTP clocks such as transparent clock (TC) or boundary clock (BC). The server layer consists of a group of end-hosts that support PTP clocks such as ordinary clock (OC), and where the time-sensitive applications reside.

In Model 1, the network fabric layer consists of a chain of transparent clocks. In Model 2, the network fabric layer consists of a chain of transparent clocks and a single boundary clock that is directly connected to the OC.

The PTP profile specified in this document applies to Model 1. Model 2 is for future definition.

5.1 Model 1 – TC Model

The high-level characteristics of Model 1 shown in Figure 1 are:

- GM, TC, OC clocks are used throughout.
- GM has a single network physical port and always distribute time towards the network fabric layer and server layer. The GM defined in this PTP profile is a master-only[1] OC with a single PTP port according to 9.2.2.2 of IEEE Std 1588-2019.
- TC can have multiple network physical ports (eg., 16, 48). The TC can have multiple PTP ports.
- OC has a single network physical port and always receives time from the network fabric layer and the GM. The OC defined in this PTP profile is a slave-only[2] OC according to 9.2.2.1 of IEEE1588-2019.
- In this profile, an OC can never be a GM and a GM can never be an OC.
- All network physical ports will be 40GE, 100GE or higher. All links are optical fibers.
- Hardware timestamping (physical layer) is enabled throughout.
- In normal operating mode, an OC has connectivity to more than 1 GM.
- There are a number of GMs that are either active or standby.
- An OC communicates with a GM based on unicast discovery and unicast negotiation protocol.
- The communication is performed via IPv6.
- The end-to-end delay mechanism is enabled throughout.
- The number of TCs between GM and OC is constant. For example, if the number of TC = 5, then there will be 7 clocks in total (i.e., including 1 GM and 1 OC) with 6 links interconnecting the clocks.

[1] See Section 3 - Terminology

[2] See Section 3 - Terminology

- Forward path direction (GM to OC) and reverse path direction (OC to GM) might not be congruent. That is, PTP packets in the forward direction might traverse different set of TCs from PTP packet in the reverse direction.
- Delay asymmetry due to fiber links is assumed to be negligible in comparison to the time error requirements.

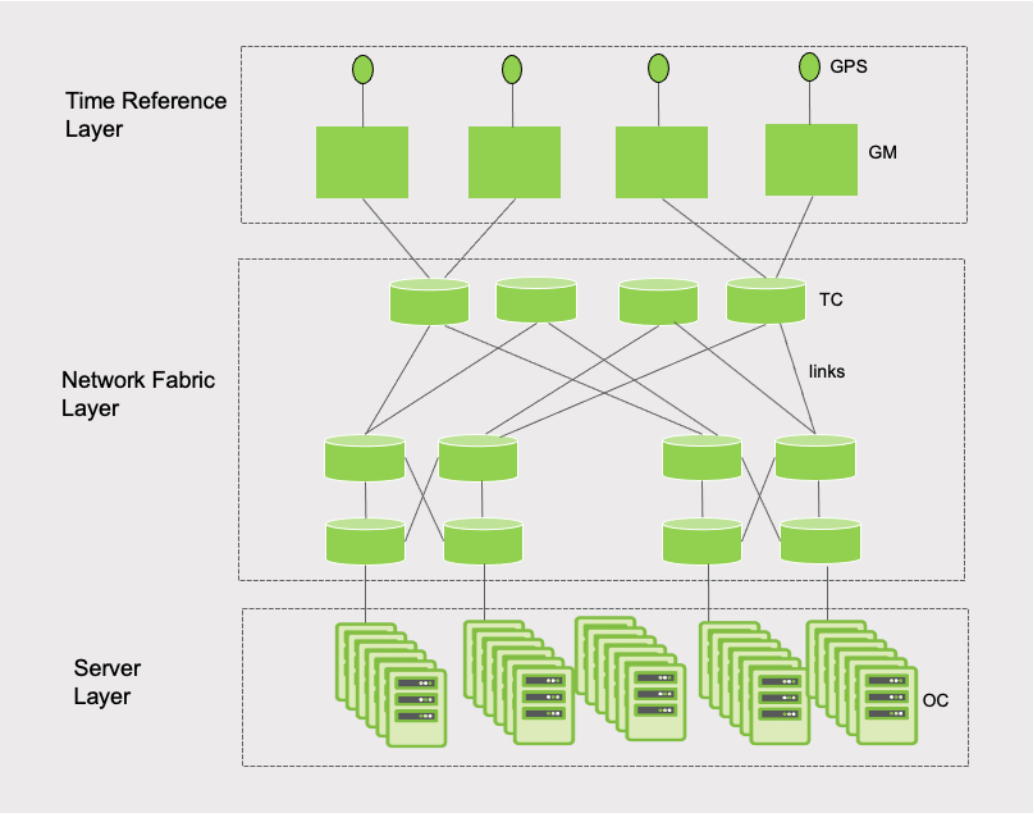

Figure 1. Model 1 – Chain of Transparent Clocks

5.2 Model 2 – TC+BC Model

Model 2 is shown in Figure 2. The main difference between Model 2 and Model 1 consists in the last TC being a BC. This model is for future definition.

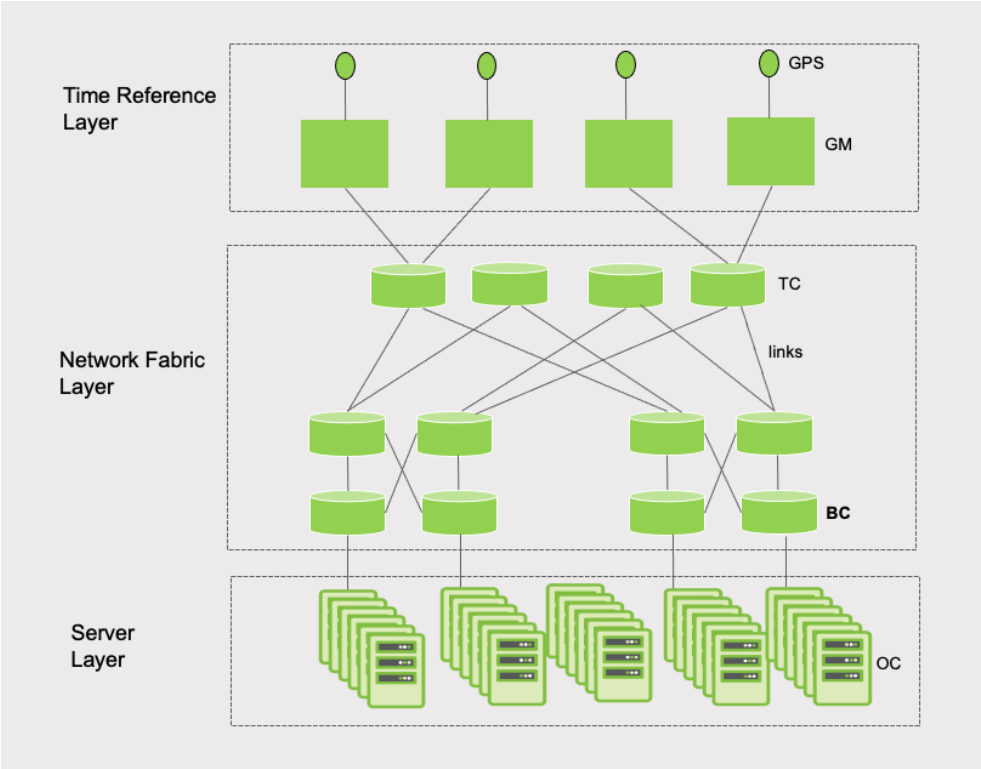

Figure 2. Model 2 – Chain of Transparent Clocks and Boundary Clocks

6 Model 1 - Additional Requirements

- Communication between PTP clocks is based on IPv6
- Communication between each type of clock must be unicast
- The PTP clock discovery and selection algorithm do not rely on multicast or broadcast communication
- The higher layer applications require UTC. The PTP protocol transports the PTP Timescale (i.e., TAI) plus all information to derive the UTC Timescale from the TAI timescale. It is up to the application to perform timescale conversion
- The maximum time error between any two OCs must be within ±5 microseconds, i.e., $|T_{OC,j} - T_{OC,k}| \leq 5\ \mu s$ for $k \neq j$
- The maximum time error between a GM and any OCs must be within ±2.5 microseconds. i.e., $|T_{GM} - T_{OC}| \leq 2.5\ \mu s$
- The maximum time error between any two GMs must be within ±100 nanoseconds, i.e., $|T_{GM,j} - T_{GM,k}| \leq 100$ ns for $k \neq j$
- The maximum time error generated by a TC must be within ±100 nanoseconds, i.e., $|T_{TC,j}| \leq 200$ ns
- In normal operating conditions, each OC has connectivity into multiple GMs. Under failure of a GM, an OC must have connectivity to at least another GM
- The PTP implementation is based on open source linuxptp [3]

7 PTP Profile

The PTP profile is based on IEEE1588-2019.

7.1 PTP Profile

The information below identifies the profile. The profile is defined by OCP.

profileName: PTP profile for data center application (DC-PTP Profile 1)
profileNumber: 1
profileVersion: 1.0 (Version primaryVersion.revisionNumber)
profileIdentifier: TBD (3 octets OUI/CID + profileNumber + profileVersion)
organizationName: Open Compute Project (OCP)
sourceIdentification: This profile is specified by OCP and can be downloaded from https://www.opencompute.org (url link is TBD)

Note: Obtaining the OUI/CID is work in progress.

7.2 Clock Types

The profile allows for the following clocks to be used.

GM	An ordinary clock that synchronizes to an external reference and distributes time via PTP.
BC	Future
TC	An end-to-end transparent clock that does not interact with the PTP protocol except for making correction field adjustments.
OC	An ordinary clock that synchronizes to a GM via PTP.

Some of the requirements that pertain to the GM and that are outside the PTP Profile are defined in the ‘Open Source Grandmaster’ [4].

7.3 Message Types

The profile allows for the following messages:

a) Announce
b) Sync
c) Follow_Up
d) Delay_Req
e) Delay_Resp
f) Signaling
g) PTP management

7.4 Transport mechanisms required, permitted, or prohibited

The required transport mechanism is UDP over IPv6 per Annex D of IEEE1588-2019. The networkProtocol value must be 0002 (hex), the addressLength value must be 16 and the addressField must represent the IPv6 source address.

The transport mechanism IPv4 per Annex E may be required.

The UDP checksum needs to be updated anytime a PTP message is being processed.

7.5 Clock identity

The clockIdentity must be an EUI-64 as specified in 7.5.2.2 of IEEE Std 1588-2019. The EUI-64 must be globally unique. If the EUI-64 is formed from an existing EUI-48, it must be done by appending two octets after the final six octets of the EUI-48 such that the 64 bits of the clockIdentity are not the same as the bits of any EUI-64 that has previously been assigned or may be assigned in the future by an authorized assignee of the MA-L, MA-M, or MA-S from which the EUI-48 was assigned. This means that either the entity that forms the EUI-64 owns the MA-L, MA-M, or MA-S from which the EUI-48 was formed, or the owner of that MA-L, MA-M, or MA-S has given the entity that forms the EUI-64 the sole right to the clockIdentity being formed.

Note: When using the MAC address, the clock identity is created by appending two octets after the final six octets of the MAC address. Note that in IEEE Std 1588-2008 the clock identity was formed by adding the two octets ‘FFFE’ between the 3rd octet and 4th octet of the MAC address, however, that mapping has been deprecated by the IEEE.

7.6 Path delay Measurement Mechanism

The path delay measurement mechanism must be the delay request-response mechanism. The value of the data set member portDS.delayMechanism must be E2E.

7.7 Class of Service

PTP event messages should set the DSCP field of the IPv6 Traffic Class field to the highest class of service possible. This should minimize latency and delay variation as PTP packets traverse a set of transparent clocks.

In Model1, the GM and OC should set the traffic class value.

7.8 Profile Isolation and Domain Number

The sdoId is a new parameter in 1588-2019. A recognized standards organization, industry trade association, regulatory or government organization, or other organization as described in 20.3.2 of IEEE Std 1588-2019, can obtain an sdoId from the IEEE Registration Authority (RA). The sdoId is used to

ensure that a PTP profile is isolated from any other PTP profiles running on the same network that are developed by other organizations.

An organization can obtain only one sdoId. If the organization develops multiple PTP profiles and requires that they be isolated, the isolation is further done using domainNumber. If an organization does not obtain an sdoId, the PTP profile will use the sdoId 0x000.

This PTP profile does not require an sdoId since it will be the only profile within the data center.

Note – The sdoId is backward compatible with IEEE Std 1588-2008. The first nibble of the sdoId, i.e., the majorSdoId, corresponds to the transportSpecific field of IEEE Std 1588-2008. The final 8 bits of the sdoId, i.e., the minorSdoId, was reserved in IEEE Std 1588-2008 and was specified as 0x00.

7.9 One-step and two-step operation

A GM defined in this profile must support one-step or two-step operation on transmit, or can support both on transmit.

A TC defined in this profile must support one-step operation on transmit (i.e., egress) and may support two-step operation on transmit (i.e, egress).

All PTP clocks must support both one-step and two-step operation on receive (i.e., ingress).

A PTP port can transmit a Sync message as one-step or two-step. If the transmission of the Sync message is one-step, the twoStepFlag of the PTP common header is set to FALSE, otherwise it is set to TRUE. For PTP messages other than Sync, the twoStepFlag must always be set to FALSE. All PTP Ports must be capable of receiving and processing one-step and two-step Sync messages.

Note: one-step operation reduces the number of PTP messages transmitted by a PTP port. This may be applicable when considering scalability of unicast communication that a GM can serve. A one-step operation might ease meeting the requirements regarding the transmission of Sync messages specified in 9.5.9 of IEEE Std 1588-2019.

Note: IEEE Std 1588-2019 allows one-step versus two-step operation to be on a PTP port basis. However, IEEE Std 1588-2019 does not describe this capability. This profile requires that all PTP ports on a per clock basis be the same.

7.10 End-to-End TC with two-step operation

This section applies to the scenario where two-step TC operation may be used.

If an end-to-end TC uses two-step operation, each Delay_Req and corresponding Delay_Resp message must traverse that same end-to-end TC. This is because the end-to-end TC timestamps the Delay_Req message on ingress and egress and computes the residence time of the Delay_Req message. However, in the two-step case the TC updates the residence time of the corresponding Delay_Resp message. This is described in detail in 10.2.2.2.2 and 10.2.2.2.3 of IEEE Std 1588-2019. The former subclause describes the one-step case and specifies that the “<residenceTime> of the Delay_Req message must be added to the correctionField of the Delay_Req message by the egress PTP Port of the TC prior to the retransmission of the Delay_Req message.” In this case, it is the Delay_Req message that is altered by

the TC, and not the Delay_Resp message. However, the latter subclause describes the two-step case, and specifies that the “<residenceTime> must be added to the correctionField of the Delay_Resp message associated with the Delay_Req message prior to transmission of the Delay_Resp message on the egress PTP Port, which is the ingress PTP Port for the Delay_Req message.”

If all the TCs are two-step, the Delay_Req and Delay_Resp must traverse the same set of transparent clocks (links and network elements) between the GM and OC in order to meet the subclause requirements. This property might not always hold true when using for example packet spraying, load balancing and equal cost multipath techniques. This is particularly applicable to data center environments and a reason for requiring the use of one-step TCs.

If all the TCs are one-step, the Delay_Req and Delay_Resp need not traverse the same set of TCs (links and network elements) between the GM and OC.

7.11 PTP message rate

Table 1 defines the range of message rates for Announce, Sync, Delay_Req, and Delay_Resp messages. A GM must support the full range. An OC should support the whole range but can support a subset of the range. The message rate selected by an OC relates to the performance expected. A TC is agnostic to the PTP message rate.

Message	Upper end of logMessageInterval range	mean rate corresponding to upper end of range (pps)	Lower end of logMessageInterval range	mean rate corresponding to lower end of range (pps)
Announce	0	1	-3	8
Sync	+3	0.125 (1 per 8 s)	-7	128
Delay_Req & Delay_Resp	0	1	-7	128

Table 1. Range of logMessageInterval for a PTP Port

7.12 PTP inter-message interval

The requirements for the actual inter-message intervals for unicast Announce, Sync, Delay_Req, and Delay_Resp messages are specified in 16.1 of IEEE Std 1588-2019. There are requirements for:

- (a) the arithmetic mean of the successive inter-message interval computed over a suitable number of successive intervals
- (b) the distribution of the inter-message intervals

For Announce and Sync messages, the arithmetic mean of the inter-message intervals must be within ±30% of the granted inter-message period. For Delay_Req and Del_Resp messages, the arithmetic mean of the Delay_Req inter-message intervals must not be less than 90% of the granted inter-message interval for the Delay_Resp messages. The purpose of this requirement is to ensure that the GM port receives Delay_Req messages at rates that it is able to handle. If the mean inter-message interval of the Delay_Req messages is less than 90% of the granted inter-message interval for the Delay_Resp

messages, the master port (i.e., grantor) may ignore any Delay_Req messages in excess of the granted interval.

For the distribution of the inter-message intervals, at least 90% of the inter-message intervals must be within ±30% of the granted mean inter-message interval. This requirement applies to Announce, Sync, and Delay_Req.

Consider N successive inter-message intervals Δt_i , i = 1, 2, …, N, where $\Delta t_i = (t_i - t_{i-1})$ is as shown in the figure. The arithmetic mean of the inter-message intervals, Δt_{av}, is

$$\Delta t_{av} = \frac{1}{N}\sum_{i=1}^{N} \Delta t_j$$

For example, if the master port grants Sync or Announce messages with logMessageInterval equal to 0, the mean inter-message interval is 1 s. This means that (a) the average of the durations of a suitable number of successive inter-message intervals Δt_{av} must be between 0.7 s and 1.3 s, and (b) 90% of the actual inter-message intervals must have durations that are between 0.7 s and 1.3 s. In addition, if the GM port grants Delay_Resp messages with logMessageInterval equal to 0, then (a) the average of the durations of a suitable number of successive Delay_Req inter-message intervals must be greater than or equal to 0.9 s, and (b) 90% of the actual Delay_Req inter-message intervals must have durations that are between 0.7 s and 1.3 s.

In principle, the mean Sync rate and the mean Delay_Req/Delay_Resp rate need not be the same. If the actual delay on the PTP communication path is changing sufficiently slowly (after the OC has processed any correction field), then infrequent delay measurements compared to the mean Sync interval might give acceptable performance. In this case, the mean Delay_Req/Delay_Resp rate can be chosen to be smaller than the mean Sync rate. The Sync rate that is chosen depends on the implementation of the OC filter and how much noise the oscillator at the OC generates. If the oscillator has a large noise generation, then the Sync rate would likely be larger. In this case, the OC would use new Sync information more frequently to correct for time error.

7.13 Unicast Communication

PTP communication in this profile must communicate using unicast. Most PTP profiles in the industry are based on multicast, except for two of the ITU-T telecom profiles that are based on unicast [5, 6].

Both unicast discovery (section 17.4 of IEEE Std 1588-2019) and unicast negotiation (section 16.1 of IEEE Std 1588-2019) must be supported. In Model 1, each OC first uses unicast discovery to determine the potential GMs, and then uses unicast negotiation to request Announce messages from the potential GMs. The OC then invokes the Best Master Clock Algorithm (BMCA) to determine which of the potential GMs becomes the actual GM, i.e., the active GM. Finally, the OC uses unicast negotiation to request Sync and Delay_Resp messages from the active GM and uses the Sync, Delay_Resp, and Delay_Req information to synchronize to the actual GM. The other potential GMs are available as backup, i.e., standby GMs in the event that the active GM fails.

The unicast negotiation feature is permanently enabled. The unicastNegotiationPortDS.enable member (of the unicastNegotiationPortDS) must be TRUE for each PTP port (there is a unicastNegotiationPortDS for each PTP port). This dataset member applies to GM and OC and is not applicable to TC.

The unicastFlag of all PTP messages must be set to TRUE.

7.13.1 Unicast Discovery

Unicast discovery is specified in 17.4 of IEEE Std 1588-2019.

In Model 1 of this PTP profile, a table of potential GMs is configured in each OC. The table is sometimes referred to as the Unicast Master Table (UMT) and is defined in the unicastDiscoveryPortDS data set (see 17.4.3 of IEEE Std 1588-2019). This data set contains the following members:

a) maxTableSize: the maximum number of potential GMs that can be in the table
b) logQueryInterval: the logarithm to base 2 of the mean time interval, in seconds, between successive requests that the OC makes to a potential GM for Announce messages (if a request is not granted);
c) actualTableSize: the number of potential GMs currently in the table; and
d) portAddress: an array containing the protocol addresses, i.e., IPv6 addresses of the potential GMs.

Each OC uses unicast negotiation (see 7.13.2) to request Announce messages from each potential GM contained in the unicastDiscoveryPortDS. If a potential GM does not grant the request, the OC attempts again after a time interval corresponding to logQueryInterval. The received Announce messages cause a state decision event (see 7.14 below), which causes the BMCA to be invoked. This results in one of the potential GMs becoming the active GM. Any other potential GMs are standby GMs. If the active GM fails, the OC will stop receiving announce messages and the accounceReceiptTimeout will expire. This will invoke the BMCA. The BMCA will result in one of the standby GMs (i.e., the best of the remaining potential GMs) becoming the active GM. If there are no GMs in the unicastDiscoveryPortDS or if none of the GMs in the unicastDiscoveryPortDS grants Announce messages to the OC, the OC will go into either free-run or holdover.

After the GM is selected, the OC uses unicast negotiation to request Sync and Delay_Resp messages from the GM. Upon being granted Sync messages, the OC receives the Sync messages from the GM. Upon being granted Delay_Resp messages, the OC sends Delay_Req messages to the GM and receives a Delay_Resp message in response to each Delay_Req message.

Figure 4 shows an example which consists of 1000 OCs divided into 2 groups, each with 500 OCs. There are 4 potential GMs, designated 1 through 4, respectively. GM 1 and GM 2 are potential GMs for OC group 1 and their IPv6 address is entered into the unicastDiscoveryPortDS of each OC of group 1. GM 3 and GM 4 are potential GMs for OC group 2 and their IPv6 address is entered into the unicastDiscoveryPortDS of each OC of group 2. The attributes of the GMs are set such that GM 1 is better than GM 2 as determined by the BMCA and GM 3 is better than GM 4 as determined by the BMCA. Assuming the GMs all have the same clockClass, clockAccuracy, and offsetScaledLogVariance, this can be done by configuring the priority2 attributes such that priority2 for GM 1 and GM 3 is less than priority2 for GM 2 and GM 4, respectively. This assumes that priority1 is set to the same default value in all GMs to prevent it from accidentally overriding the effect of clockClass, clockAccuracy, and offsetScaledLogVariance. This is done in other PTP profiles such as ITU-T Rec. G.8275.2, which is also based on unicast discovery and unicast negotiation. Alternatively, if clockClass, clockAccuracy, offsetScaledLogVariance, and prioirity2 are the same in each potential GM but the clockIdentities of GM 1 and GM 3 happen to be less than the clockIdentities of GM 2 and GM 4, respectively, GM 1 and GM 3 will also be chosen as the active GMs for groups 1 and 2, respectively. In addition, in this final case where the potential GMs have the same clock attributes, it might not matter which is active and which is standby. The BMCA will result in GM 1 and GM 3 being the active GMs for OC groups 1 and 2, respectively, and GM 2 and GM 4 being the standby GMs for OC groups 1 and 2, respectively. The use of clockIdentites is the tiebreaker.

Example 1 also shows that a standby GM is not utilized if the active GM of the respective OC group has not failed. In this example, failures of both active GMs can be tolerated; however, two standby GMs, are not utilized unless there are failures.

For Model1, the BMCA can be the default BMCA as specified in IEEE Std 1588-2019. It can also be based on ITU-T G.8275.2 given that stepsRemoved = 1 for Model 1 and that localPriority attribute has no significance given the OC has a single port in this profile. Both BMCAs produce the same behavior for Model 1.

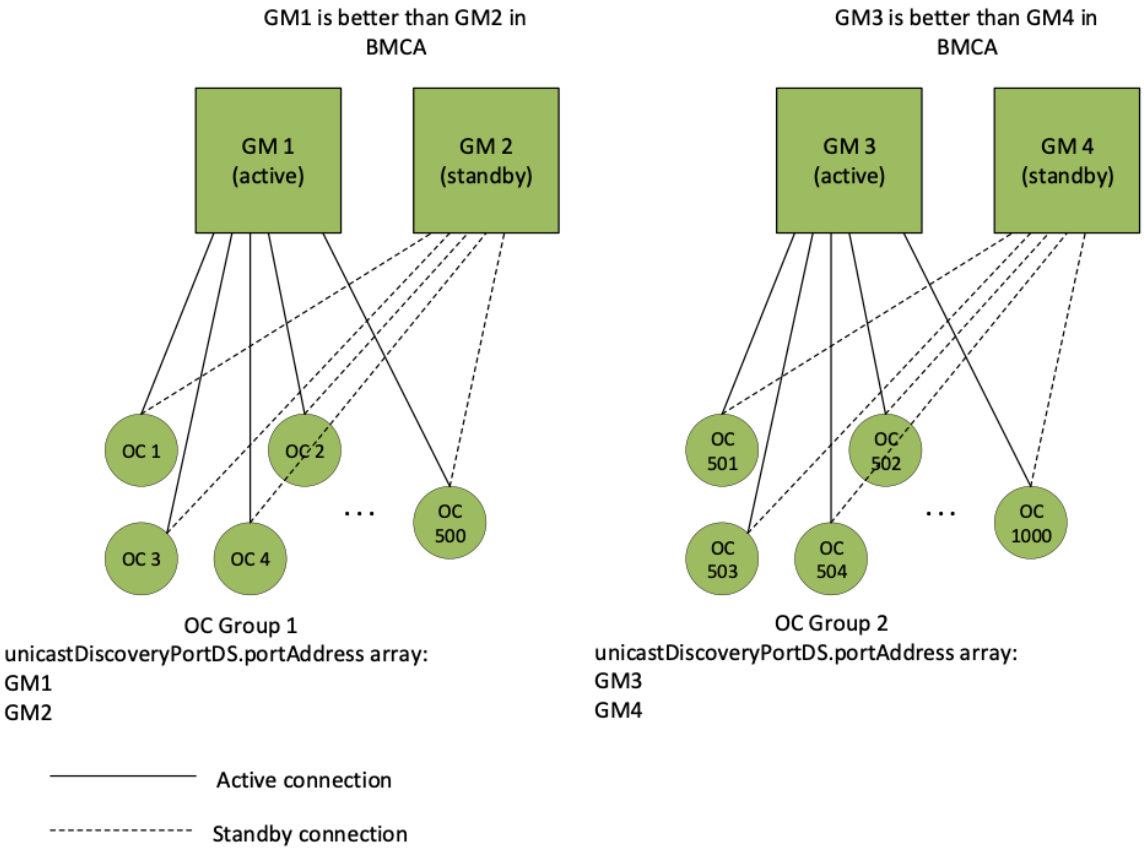

Figure 3. Example1 of Active/Standby GMs across two groups each with 500 OCs

Figure 4 shows an example which consists of 2 OC groups with 3 potential GMs, designated 1 through 3, respectively. GM 1 and GM 3 are potential GMs for OC group 1 and are entered into the unicastDiscoveryPortDS of each OC of group 1. GM 2 and GM 3 are potential GMs for OC group 2 and are entered into the unicastDiscoveryPortDS of each OC of group 2. The attributes of the GMs are set such that GM 1 and GM 2 are each better than GM 3 as determined by the BMCA. As in the example above, this can be done by configuring the priority2 attributes such that priority 2 for GM 1 is less than priority 2 for GM 3, and priority2 for GM 2 is less than priority2 for GM 3. This will also occur if the clockIdentities of GM 1 and GM 2 are each less than the clockIdentity of GM 3 and all the other attributes of GMs 1, 2, and 3 are the same. The BMCA will result in GM 1 and GM 2 being the active GMs for OC groups 1 and 2, respectively. GM 3 will be the standby GM for both groups 1 and 2. If either GM 1 or GM 2 fails, GM 3 will become the GM for the group whose GM has failed. If both GM 1 and GM 2 fail, then either GM 3 will become the GM for both OC groups 1 and 2, and therefore must be able to handle the load of both groups or only a single failure (i.e., of a single GM) can be tolerated.

In Example 2, there is only a single standby GM, and therefore only a single GM is not utilized if there are no failures (unlike Example 1, where two GMs are not utilized if there are no failures). However, either the single standby GM must handle a higher load if both active GMs fail, or else only a single active failure can be tolerated.

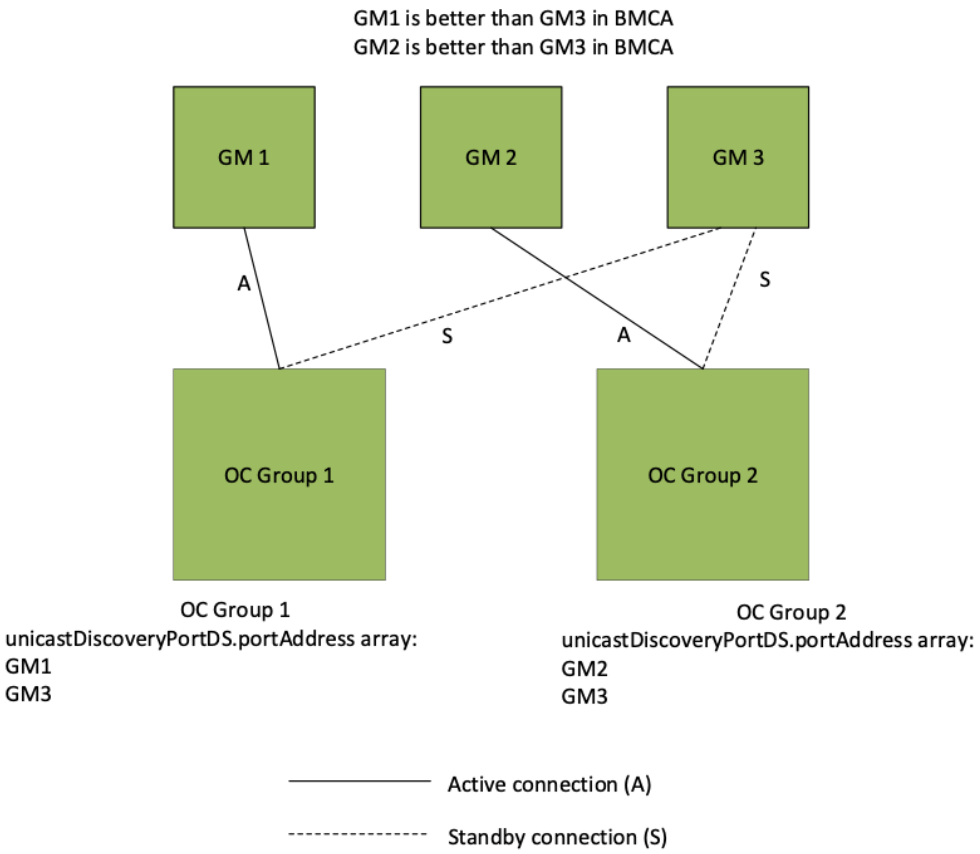

Figure 4. Example2 one active GM for each group and one standby GM for both groups

Example 1 illustrates the case of full redundancy where there is one standby GM for each active GM. Example 2 illustrates the case of partial redundancy where there are fewer standby GMs than active GMs. To balance the load among the active GMs, the OCs should be divided as evenly as possible among the active GMs. To balance the load among the standby GMs and also achieve maximum robustness to failures, the standby GMs should be assigned to equal numbers of OC groups. For example, if there are 10000 OCs, 10 potential active GMs, and 2 potential standby GMs, the OCs should be divided into groups of 1000 OCs each (i.e., 10000 OCs/10 GMs), and each of the 10 potential active GMs should be entered in the unicastDiscoveryPortDS of the OCs of exactly one group. Each potential standby GM should be entered in the unicastDiscoveryPortDS of the OCs of exactly 5 groups (and no group should have two standby GMs entered in the unicastDiscoveryPortDS of any of its OCs). With this approach, a standby GM serves as a backup for up to 5 OC groups. If a standby GM can handle the load of up to N groups ($N \leq 5$), then N active failures can be tolerated.

7.13.2 Unicast Negotiation

As described in 7.13.1, an OC requests Announce messages and then selects the best potential GM using the BMCA. The OC then requests Sync and Delay_Resp messages from that GM. After the OC is granted Sync messages, the GM sends Sync (and Follow_Up if the communication is two-step) messages to the OC. After the OC is granted Delay_Resp messages, the OC sends Delay_Req messages to the GM and the GM responds with Delay_Resp. The requesting of Announce, Sync and Delay_Resp messages is done using the unicast negotiation feature of IEEE Std 1588-2019. The unicast negotiation feature is performed using the following four TLVs:

a) REQUEST_UNICAST_TRANSMISSION
b) GRANT_UNICAST_TRANSMISSION
c) CANCEL_UNICAST_TRANSMISSION
d) ACKNOWLEDGE_CANCEL_UNICAST_TRANSMISSION

Each TLV is attached in a Signaling message.

The sending, receiving, and processing of unicast negotiation TLVs by OCs and GMs must comply with the requirements of 16.1 of IEEE Std 1588-2019 and its subclauses. The following text in this section is a summary description of the unicast negotiation process.

TCs do not participate in the unicast negotiation process. However, they do forward the unicast Signaling messages that contain the unicast negotiation TLVs exchanged between the OCs and GMs.

The unicast negation process is illustrated in Figures 5, 6, 7 for requesting Announce, Sync, and Delay_Resp messages, respectively. An OC requests unicast Announce, Sync, or Delay_Resp from a GM by sending a REQUEST_UNICAST_TRANSMISSION TLV to the GM. This TLV contains the messageType field, which indicates the type of message (i.e., Announce, Sync, Delay_Resp), the logInterMessagePeriod field, which is the logarithm to base two of the desired mean interval, in seconds, between successive messages of this type, and the durationField, which is the number of seconds for which the GM should continue to transmit these messages. The GM responds with a GRANT_UNICAST_TRANSMISSION TLV to either grant or deny the request. This TLV contains the messageType field, which indicates message being granted, the logInterMessagePeriod field, which is the logarithm to base two of the granted mean interval, in seconds, between successive messages of this type, the durationField, which is the granted number of seconds for which the GM will continue to transmit these messages, and the R (Renewal Invited) flag, which is TRUE if the GM considers that the grant is likely to be renewed if the OC requests a new grant after the current grant expires and FALSE otherwise. A value of zero for the durationField indicates that the grant has been denied. The granted logInterMessagePeriod and durationField need not be the same as the requested logInterMessagePeriod and durationField, respectively.

The duration of the grant begins when the GRANT_UNICAST_TRANSMISSION_TLV is transmitted and ends after a time interval equal to the value of the durationField has expired. Typically, the OC requests that the grant be renewed by sending a new REQUEST_UNICAST_TRANSMISSION TLV before the grant expires (i.e., before the end of the duration) so that the service will be continuous.

After the GM has granted Announce or Sync messages to the OC, the GM sends Announce or Sync messages to the OC. After the GM has granted Delay_Resp messages, the OC then sends Delay_Req messages to the GM and the GM responds with Delay_Resp.

An OC can cancel the grant by sending the CANCEL_UNICAST_TRANSMISSION TLV to the GM. This TLV contains the messageType field, which indicates the type of message whose grant is being canceled, and the R (maintainRequest) flag set to FALSE. The GM responds by sending the ACKNOWLEDGE_ CANCEL_UNICAST_TRANSMISSION TLV to the OC.

If a GM cannot continue to provide the granted messages before the durationField has expired, it can inform the OC by sending the CANCEL_UNICAST_TRANSMISSION TLV to the OC with the G (maintainGrant) flag set to FALSE. The OC responds by sending the ACKNOWLEDGE_ CANCEL_UNICAST_TRANSMISSION TLV to the GM. The GM should (i.e., this is recommended but not required) continue to send the messages until it receives the ACKNOWLEDGE_ CANCEL_UNICAST_TRANSMISSION TLV or it has sent an implementation-specific number of CANCEL_UNICAST_TRANSMISSION TLVs to the OC.

A Signaling message can contain more than one TLV.

In this PTP profile, all requests are made by an OC and all grants are made by a GM. An OC cannot grant services and a GM cannot request services.

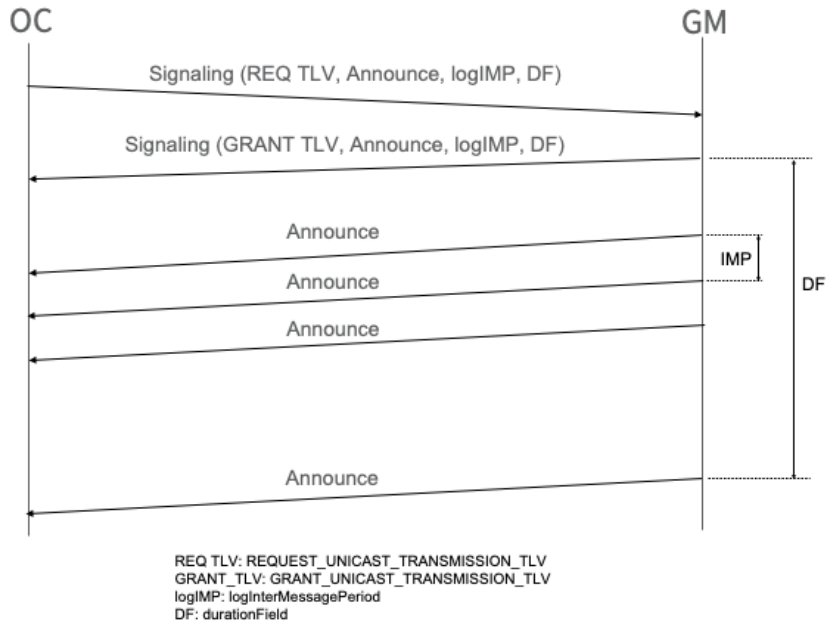

Figure 5. Unicast negotiation for Announce messages

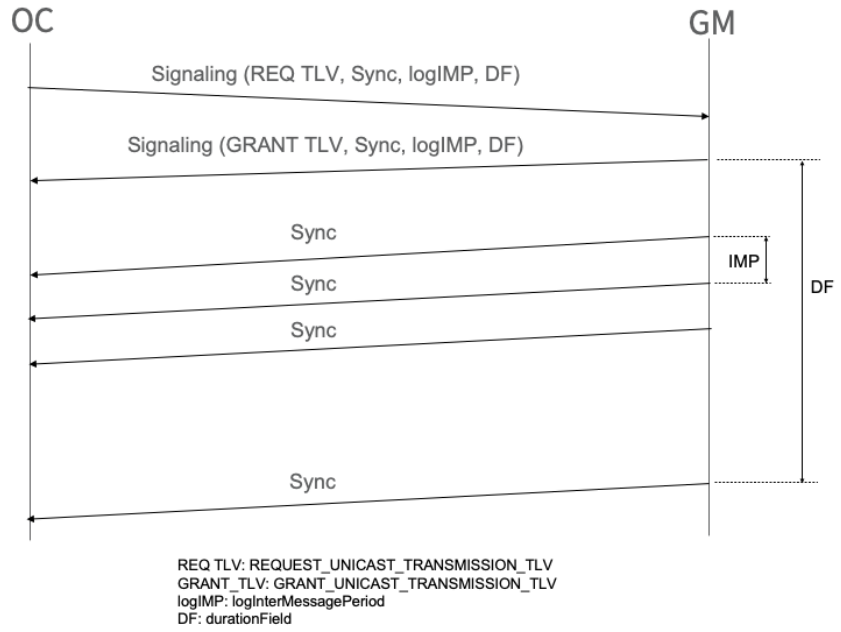

Figure 6.Unicast negotiation for Sync messages

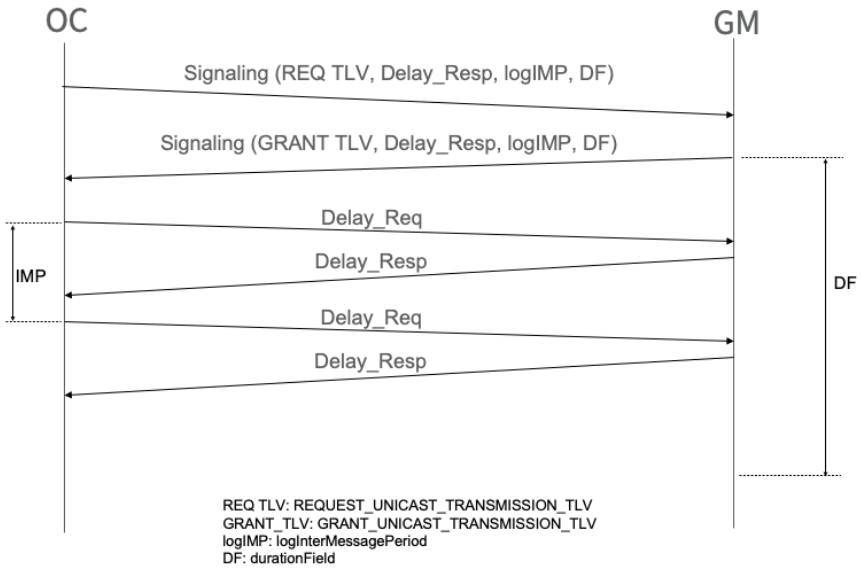

Figure 7. Unicast negotiation for Delay_Resp messages

7.14 Best Master Clock Algorithm and clock attributes

This profile is based on the default BMCA of IEEE Std 1588-2019. It can also be based on ITU-T G.8275.2 given that stepsRemoved = 1 for Model 1 and that localPriority attribute has no significance given the OC has a single port in this profile. Both BMCAs produce the same behavior for Model 1.

The clock attributes for the GM and OC are given in Table 2. The attributes clockClass, clockAccuracy, offsetScaledLogVariance are set in the defaultDS and represent properties of the local clock, which is either the internal oscillator or an external time source that provides time to the GM outside of PTP or that is integrated with the GM. The clockAccuracy and offsetScaledLogVariance are based on the max|TE| (maximum absolute time error) and TDEV (time deviation) requirements for the PRTC-A (Primary Reference Time Clock A) specified in ITU-T Rec. G.8272 [7]. For clock class, the GM data sheet should specify the maximum amount of time necessary to transition from clockClass 6, 7, 52.

The priority1 attribute is not used and is set to 128. It is not used in this PTP profile because it has higher preference in the BMCA than all the other attributes. A misconfiguration could cause an OC to choose the wrong GM as the active GM or standby GM.

The priority2 attribute can be configured to force specific potential GMs to be active or standby GMs for specific OC groups. If priority2 is not used i.e., default value of 128 in all potential GMs, then the selection of the actual GM by the BMCA is based on clockIdentity.

The attribute slaveOnly is TRUE for an OC and FALSE for a GM. The attribute masterOnly is FALSE for an OC and TRUE for a GM.

The attribute ptpTimescale is always TRUE because this PTP profile uses the PTP timescale. The other timePropertiesDS attributes have values in the GM based on whether the values are traceable to a primary reference or in the case of timeSource based on the actual source of time for the clock.

The synchronizationUncertain attribute is optional. It is carried as a flag in the Announce message. This is new in IEEE1588-2019 and might not be supported if the PTP nodes are based on previous version of IEEE1588. If it is not used, its value is FALSE. If it is used at an OC, it is set to TRUE if:

- The synchronizationUncertain flag in the Announce message received from the GM is set to TRUE, or
- The state of the PTP port of the OC is UNCALIBRATED

Otherwise synchronizationUncertain for the OC is set to FALSE. If the synchronizationUncertain attribute is used at a GM, it is set to TRUE if the GM time or frequency, or both, are not traceable to a primary reference, otherwise it is set to FALSE.

The data set members listed in Table 2 are not applicable to TCs. TCs do not participate in the BMCA.

Data set	Member	Value	
		GM	OC
defaultDS	clockClass	6 (traceable to a primary reference time source) 7 (in holdover, and within holdover specifications) 52 (in holdover but out of holdover specifications, or in free-run)	255
defaultDS	clockAccuracy	0x21 (100 ns)	0xFE (unknown)
defaultDS	offsetScaledLogVariance	0x4E5D (PTPVAR = 1.144×10^{-15} s^2, or TDEV = 30 ns)	0xFFFF (maximum possible value, signifying unknown)
defaultDS	priority1	128 (not used in this profile)	128 (not used in this profile)
defaultDS	prioirity2	Configurable over [0, 255]. Default value is 128	Configurable over [0,255]. Default value is 128
defaultDS	slaveOnly	FALSE	TRUE
portDS	masterOnly	TRUE	FALSE
timePropertiesDS	currentUtcOffset	If known, the value traceable to a primary reference that provides UTC. Otherwise the value when the node was designed	currentUtcOffset of E_{best}, after running BMCA
timePropertiesDS	currentUtcOffsetValid	TRUE if the values of currentUtcOffset, leap59, and leap61 are based on values obtained from a primary reference providing UTC; otherwise set to FALSE	currentUtcOffsetValid of E_{best}, after running BMCA
timePropertiesDS	leap59	If known, to a value traceable to a primary reference; otherwise set to FALSE	leap59 of E_{best}, after running BMCA
timePropertiesDS	leap61	If known, to a value traceable to a primary reference; otherwise set to FALSE	Leap61 of E_{best}, after running BMCA

timePropertiesDS	timeTraceable	TRUE if the time is traceable to a primary reference; otherwise set to FALSE	timeTraceable of E_{best}, after running BMCA
timePropertiesDS	frequencyTraceable	TRUE if the frequency is traceable to a primary reference; otherwise set to FALSE	frequencyTraceable of E_{best}, after running BMCA
timePropertiesDS	timeSource	If known, to the appropriate value from Table 6/IEEE Std 1588-2019. Otherwise set to INTERNAL_OSCILLATOR	timeSource of E_{best}, after running BMCA
timePropertiesDS	ptpTimescale	TRUE	TRUE
currentDS	synchronizationUncertain	FALSE (default)	FALSE (default)

Table 2. Data set members and values

7.15 Network Limits and Error Budget for Model 1

This section is an initial analysis. The network limit from Section 5, is:

- The maximum absolute time error of any OC, relative to TAI, must be $\max|TE_{OC}| \leq 2.5$ μs.
- The time accuracy difference between any two OCs must be within ±5 microseconds, i.e., $|T_{OC,j} - T_{OC,k}| \leq 5$ μs for $k \neq j$.

The following effects contribute to $\max|TE_{OC}|$:

a) Timestamp granularity. This is due to the clock frequency used for timestamping generation.
b) Timestamp generation. This is due to timestamping generation not being at the exact location where the timestamp is being taken, i.e., at the reference plane (see 7.3.4.2 of IEEE Std 1588-2019)
c) Combination of residence time and free-run accuracy of a TC. In this profile, the TCs are assumed to be free-running. They are not syntonized either at the physical layer or via PTP
d) Number of TCs in the network
e) Noise generation in the OC oscillator. This is dependent on the PLL characteristics
f) GM accuracy. This is the maximum time error of the GM relative to TAI when traceable. This profile refers to ITU-T G.8272 PRTC-A specification
g) Constant time error. This is due to link and node asymmetry after any compensation
h) Time error allowance produced by or within the application (i.e., any additional error between the PTP layer and the application/server)
i) Effect of a transient if an OC loses its active GM and switches to a backup GM
j) Effect of long-term holdover of the GM or an OC if a backup GM is not available

Table 3 contains initial assumptions for the effects given above.

Effect	Value
Timestamp granularity	8 ns
Timestamp generation	8 ns
Maximum residence time in a TC	0.1 ms
Free-run accuracy of TC oscillator	100 ppm

Number of TCs	5
OC noise generation	100 (TBR)
GM accuracy relative to TAI when traceable	100 ns
Constant time error	200 ns
Time error allowance for the application	200 ns (TBR)
Effect of a transient if an OC loses its reference to active GM and switches to a standby GM	1400 ns (TBR - see below)
Effect of long-term holdover of the GM with clockClass 7 on an OC if a backup GM with clockClass 6 is not available	1400 ns over time T specified by the vendor (TBR see below)

Table 3. Maximum absolute time error budget

The maximum error introduced by a TC due free-run accuracy and residence time is $(0.1 \times 10^{-3}\ s)(10^{-4}) = 10^{-8}\ s = 10$ ns. A TC will also introduce errors of 8 ns due to timestamp granularity and 8 ns due to timestamp generation. These errors will be added at both ingress and egress, for a total of 32 ns. The total error introduced by a TC in going from ingress to egress is therefore 42 ns.

The errors due to timestamp granularity and timestamp generation are also introduced at the GM egress and the OC ingress. These errors will add 16 ns, for a total of 32 ns.

The above errors contribute to total time error at the OC (to the end application). First, they accumulate as a Sync message traverses the network from the GM to the OC and contributes to the error in the recovered time at the OC. Second, they also accumulate as the Sync and Delay_Req message traverses the network from GM to OC and OC to GM and contribute to the error in the mean path delay at the OC. The total error that accumulates as either the Sync or Delay_Req message traverses the network, assuming there are 5 TCs in the path, is 5(42 ns) + 32 ns = 242 ns. The total error in synchronized time is therefore the sum of the error for Sync and the error in measured path delay, i.e., 242 ns (error in Sync) + 242 ns (error in meanPathdelay) = 484 ns (error in offsetFromMaster). Finally, the 100 ns for the OC noise generation must be added to give 584 ns.

The error introduced by the GM, based on PRTC-A, is 100 ns.

The total allowance for constant time error due to link and node asymmetry is based on G.8271.1. G.8271.1 allows 800 ns for a network that consists of 20 hops with links that are likely much longer than those expected in a data center environment (i.e., the fiber length between nodes in a data center are within meters or tens of meters). Given that cTE is linearly additive and that the number of clocks consists of 5 TCs, 1 OC and 1GM, the total cTE is about ¼ the allocation found in G.8271.1. Therefore, the constant time error is 200 ns.

The total error at the input of the application is 584 ns + 100 ns + 200 ns + 200 ns = 1100 ns. This is well within the $\max|TE_{OC}| \leq 2.5\ \mu s$.

If the OC loses its connection to the network and enters holdover or the GM loses its connection to its time source (e.g., GPS) and enters holdover with clockClass = 7, it can be assumed that the application already has already built-up an error of 1100 ns relative to TAI. In worst case, the application could drift another 1400 ns before it exceeds the 2.5 μs requirement. This means that the holdover requirement for the OC or the GM can be taken as 1400 ns over a time period T. This period T should be specified by the OC or GM data sheet. In addition, if the OC switches from one active GM to another active GM, any transient during this switch must be within 1400 ns.

8 References

[1] OCP Timing Appliances Project Proposal, https://drive.google.com/file/d/1LC5Ld0r3U7us_jvmKeD_ZpBJaA7Kk0O4/view, July 2020

[2] IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems

[3] The Linux PTP Project, http://linuxptp.sourceforge.net

[4] Open Source Grandmaster, https://www.opencompute.org/wiki/Time_Appliances_Project

[5] ITU-T G.8275.2, Precision time protocol telecom profile for phase/time synchronization with partial timing support from the network

[6] ITU-T G.8265.1, Precision time protocol telecom profile for frequency synchronization

[7] ITU-T G.8272, Timing characteristics of primary reference time clocks