TIGER 2

Together IP, GMPLS and Ethernet Reconsidered

M2.4

Synchronization service in Carrier-Grade Ethernet environments

Abstract:
In the context of mobile backhaul network migration from Time Division Multiplexing (TDM) technologies towards Packet Switched Networks (PSNs) supported by Carrier-Grade Ethernet as one of the preferred technologies, frequency and time distribution represents a major challenge. After briefly reviewing the main eligible protocols for such a synchronization purpose regarding their related constraints and dependencies, this document presents several innovative proposals targeting an overall performance improvement, in terms of achievable accuracy and stability, but also in terms of optimum use and management of all synchronization resources.

Editors:
Dinh Thai Bui (ALBLF), Michel Le Pallec (ALBLF)

Authors:
Dinh Thai Bui (ALBLF), Michel Le Pallec (ALBLF), Alon Geva (RAD)

Date of publication:
February 25th 2010

CELTIC
Telecommunication Solutions

EUREKA

Table of Content

Index of Tables

Table of Figures

Acronym list

1588v2: Std 1588™-2008
BC: Boundary Clock
BMCA: Best Master Clock Algorithm
CAPEX: CAPital Expenditure
CES: Circuit Emulation Service
CGE: Carrier-Grade Ethernet
DNU : Do Not Use
DSLAM: Digital Subscriber Line Access Multiplexer
EWMA: Exponentially Weighted Moving Average
E2E: End-to-End
ESMC: Ethernet Synchronization Messaging Channel
ESSM: Ethernet Synchronization Status Messages
FDD: Frequency Division Duplexing
GbE: Gigabit Ethernet
HSDPA: High Speed Downstream Packet Access
HSUPA: High Speed Upstream Packet Access
IEEE: Institute of Electrical and Electronic Engineers
IETF: Internet Engineering Task Force
IP: Internet Protocol
ISP: Internet Service Provider
LAN: Local Area Network
MA: Moving Average
MO: Mobile Operator
NE: Network Element
NodeB: UMTS BTS
NTP: Network Time Protocol
OPEX: OPerational Expenditure
OUI: Organization Unique Identifier
OSSP: Organization Specific Slow Protocol
P2P: Peer-to-Peer
PDF: Probability Density Function
PDH: Plesiochronous Digital Hierarchy
PDV: Packet Delay Variation
PDU: Payload Data Unit
PLL: Phase-Locked-Loop
ppb: part-per-billion
ppm: part-per-million
PPS: Pulse per Second
PRC: Primary Reference Clock
PSN: Packet Switched Network
PTP: Precised Time Protocol (other name of IEEE 1588 protocol)
QoS: Quality-of-Service
QL: Quality Level
RoI: Return-on-Investment
SDH: Synchronous Digital Hierarchy
SN: Sequence Number
SSM: Synchronization Status Message
SSF: Server Signal Fail
STM-1: Synchronous Transport Module 1 (155.44Mbit/s)
STP: Spanning Tree Protocol
SyncE: Synchronous Ethernet
TICTOC : Timing over IP Connection and Transfer of Clock
TC: Transparent Clock
TDD: Time Division Duplexing
TDM: Time Division Multiplex
TLV: Type Length Value
TE: Traffic Engineering
ToD: Time-of-the-Day
ToP: Timing-over-Packet
TQL: Time Quality Level
TS: Time Stamp
Tx: transmitter
UMTS: Universal Mobile Telecommunications System
LTE: Long Term Evolution (3GPP 4G)
UMTS: Universal Mobile Telecommunication System (3GPP 3G)

1 Introduction

In the context of a strong development in data services (the majority of applications are now over IP), especially in the mobile networks, traditional TDM infrastructure is not any more able to meet the following equation: offering more bandwidth and guaranteeing a viable RPU (Revenue Per User) (as explained in the deliverable D20). This gives strong arguments to operators to migrate their transport infrastructure towards Packet Switched Networks (PSNs) with Carrier-Graded Ethernet (CGE) being one technology of choice. However, many technical challenges are raised for this migration and the synchronization network requirements represent a major hurdle.

Indeed, during the migration of the transport infrastructure from the TDM towards the PSN, synchronization is lost, as the traditional Ethernet is not being capable of distributing neither a frequency reference nor a time/phase reference. Alternatives such as the GPS (Global Positioning System) raise many technical issues and cost drawbacks (coverage, interference, cost of the numerous receivers, cabling and engineering costs, radio-frequency jamming, etc) as well as political concerns (some countries avoid from being GPS dependant).

Within this framework, the ITU-T's Synchronous Ethernet (SyncE) is seen as the preferred technology to distribute a frequency reference and the IEEE‘s 1588V2 (PTPV2) is considered, by major telecom actors, as the preferred technology to distribute a time or phase reference. Concerning this latter aspect, the IETF's Network Time Protocol (NTP) is also to be considered, as per existing deployments.

In this document, studies will focus on the mobile backhauling as the target application for the network synchronization over the PSN. Such focus is motivated by the related stringent mobile synchronization requirements driven by an optimal use of rare radio resources under high speed mobility constraints (e.g. interferences due to Doppler effects). Through different deployment scenarios, potential issues will be pointed out and related network requirements will be described.

2 Terminologies

In order to fully understand the content of this document, some terminological definitions are provided below.

On-Path Support (OPS) node: timing support node along the communication path between a “time Master” and a “time Slave”. This node allows for improving the time distribution. Namely Boundary Clocks, Transparent Clocks, Synchronous Ethernet Elements are on-path support nodes.

Syntonization: frequency synchronization. Two oscillators are syntonized when they have the “same” frequency within a defined error. The latter is defined as the syntonization accuracy. When two oscillators are syntonized the difference between their respective phases remains constant with time.

Time/phase: even though wireless technologies only require network phase synchronization, the term “time synchronization” will be also used in the context of mobile backhauling. Time and phase will be indifferently used.

3 Mobile network synchronization requirements

Within the mobile network infrastructure, radio interfaces impose the most stringent synchronization constraints as explained in Chapter 1. Two technology types are to be distinguished:

- FDD (Frequency Division Duplex) technologies only require network syntonization
- TDD (Time Division Duplex) technologies require both network syntonization and alignment in phase (or phase synchronization).

The mobile network requirements at the radio interface for different mobile technologies are summarized in the following table:

Radio System	**Frequency accuracy**	**Time accuracy**
GSM	50 ppb	NA
UMTS FDD	50 ppb	NA
UMTS TDD	50 ppb	2.5 µs (1.5µs*)
CDMA2000	50 ppb	3 µs (10 µs worst case)
TD-SCDMA	50 ppb	3 µs
LTE FDD	50 ppb	NA
LTE TDD	50 ppb	3 µs
WIMAX 802.16e	8 ppm	0.5-25µs according to system profile

(*) some standards define max base station to base station time difference and not a difference against a common reference as here

Table 1 – Mobile networks: synchronization requirements at the air interface level

It is to be noted that radio base stations introduce noises into the synchronization signal. Thus, requirements are typically more stringent at the network interface level (output of the backhaul network) than at the air interface level. As an illustration, a 16 ppb value at the network level is often mentioned while targeting a 50 ppb frequency accuracy at the radio interface level [1].

4 Key protocols and their related dependencies

4.1 Protocol overview

The following sections shortly describe the three main synchronization protocols that will be the study focus within the TIGER II project.

The ITU-T Synchronous Ethernet protocol is purely dedicated to the distribution of a frequency reference within an IEEE 802.3 Ethernet-capable network. It is based on a hierarchical synchronization method using a synchronous physical layer. As specified by the ITU-T within a migration perspective, SyncE is aligned with SDH/synchronous optical network (SONET) in the sense that a SyncE clock element can be inserted seamlessly into a SDH/SONET clock hierarchy. This approach should be efficient in terms of performance as SyncE is expected to be at the same performance level as SDH/SONET.

The IEEE 1588v2 protocol, also called as the Precision Time Protocol version 2 (PTPv2), provides a standard protocol for synchronizing clocks within a PSN. PTPv2 was approved as a revision standard by the IEEE standards Board in March 2008 with the intention to evolve the IEEE 1588v1 from its first deployed local area network (LAN) environment. Many enhancements have been added for this purpose, including concepts of transparent clock, boundary clock, and unicast support. Some of these elements will be detailed later in this document. Unlike PTPv1, PTPv2 is specified for different underlying layers — 802.3/Ethernet, User Datagram Protocol (UDP)/Internet Protocol version 4 (IPv4), and UDP/Internet Protocol version 6 (IPv6). From this perspective, the protocol is more suitable for meeting the needs of the telecom industry than its predecessor. The PTPv2 protocol is designed to distribute a time reference across a communication network between a master clock and a slave clock.

Growing out of the work conducted by Dr. David Mills at the University of Delaware, the IETF's Network Time Protocol (NTP) has taken advantage of lengthy deployment experience with various standards versions:
- NTPv0 (RFC 958), 1985
- NTPv1 (RFC 1059), 1988
- NTPv2 (RFC 1119), 1989 and
- NTPv3 (RFC1305), 1992.

Today, there are millions of NTP servers and stations deployed worldwide, and NTP software is implemented on almost every workstation and server platform [2]. The current NTPv4 version is nearly formalized as a standard IETF RFC and comparatively to the previous version, NTPv4 targets are:

- To accommodate Internet Protocol version 6 address space,
- To extend potential accuracy to the tens of microseconds for modern workstations and fast LANs by including fundamental improvements in the mitigation and discipline algorithms,
- To ease NTP deployment by including a dynamic server discovery scheme, so that in many cases specific server configuration is not required,

- To specify an autonomous authentication protocol,
and
- To correct certain errors in NTPv3 design and implementation and to include an optional extension mechanism.

4.2 SyncE constraints and performance dependencies

Similarly to SDH/SONET, SyncE has being designed for being independent from the traffic load. Thus, the experience with SDH/SONET should be beneficial to SyncE in that the former has shown it has been capable of synchronizing very large networks with high accuracy. However, making use of the physical layer to distribute a reference frequency imposes the SyncE/SDH chain to be contiguous and many issues can break this rule:

- A SyncE equipment in the chain is not 100% compliant to the standard
- Interoperability issue between 2 consecutive SyncE equipments
- Mis-configuration or mis-installation; e.g. an Ethernet port is peer by mistake to a SyncE port
- ESMC discrepancy; e.g. Synchronization Status Messages (SSM) corrupted
- etc

Apart from these technical aspects, regarding the full deployment of hardware support on Ethernet nodes, cost is a major obstacle which could prevent from having a full migration of the traditional Ethernet towards SyncE, thus from having a contiguous chain of physical frequency distribution.

4.3 Timestamp protocols constraints and performance dependencies

4.3.1 NTP and PTPV2 comparison

Unlike PTPv2, which has been specified for different underlying layers, IETF NTPv4 relies on UDP/IP (version 4 or 6) and is thus a “pure layer 3” solution in terms of connectivity. All future developments of NTPv4 have been officially shifted to the IETF Transmitting Timing over IP Connections and Transfer of Clock (TICTOC) working group. This decision has been driven by the desire to converge existing time protocols and thus to ensure — as a first step — 1588v2/NTPv4 coexistence within the same network. Similar to PTP, NTP has been conceived for a time distribution purpose, but can be used for syntonization as well. Server-to-client time distribution methodology over an NTP-capable network (over IP Packet Network) is conceptually close to the 1588v2 master/slave principle regarding the “2-way signalling” (synchronization message exchanges in both Master-to-Slave and Slave-to-Master directions).

Thanks to many mechanisms introduced such as the hardware timestamping, the Transparent Clock, the maximum message rate, etc, PTPV2 is often preferred to NTP for the distribution of an accurate time (and thus frequency). Nevertheless, as final tasks within the TICTOC Transition, the NTPV4 working group is specifying a “Carrier-Class” NTP, addressing the hardware time stamping and maximum message rate requirements. With these additional features, the “Carrier-Class” NTP should provide similar performance with regards to 1588V2 in an end-to-end scheme. Such a scheme is presently investigated by the ITU-T SG15/Q13 within the context of the first IEEE1588V2 frequency-only Telecom profile (frequency distribution purpose without neither Boundary Clock nor Transparent Clock support).

In the rest of this document, IEEE1588V2 will particularly be the investigation focus without forgetting that the (Carrier-Class) NTP could be a relevant candidate as well for most of the scenarios.

4.3.2 IEEE 1588V2 constraints and performance dependencies

For a clear understanding of the document, the following figure briefly recalls the principle of the PTPV2 protocol (very similar to NTP):

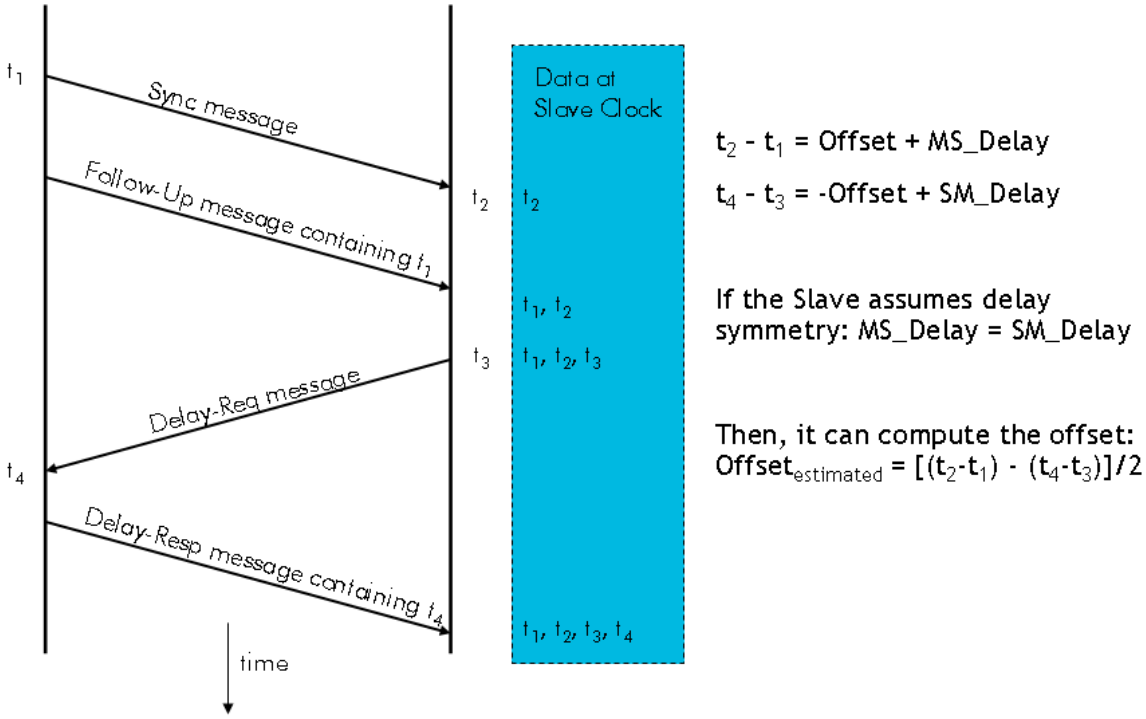


Figure 1 – PTPV2 principles

PTPV2 performance mainly depends on 2 parameters:

- The delay asymmetry: the difference between the transmission delay of the (Mater/Slave) Sync message and the related Delay-Req message.
- The Packet Delay Variation (PDV): the difference between the transmission delay of a PTP message and a reference transmission delay (transmission delay of the first message, theoretical minimum delay, theoretical maximum delay, etc).

The above dependency can be seen as the error is computed between the estimated offset and the ideal offset. The latter offset allows the Slave to adjust its timescale exactly to the Master one:

$$t_2 = t_1 + d_{MS} + offset_{ideal}$$

$$t_4 = t_3 + d_{SM} - offset_{ideal}$$

with d_{MS} being the transmission delay of the Sync message and d_{SM} being the transmission delay of the associated Delay-Req (Slave/Master). Thus,

$$offset_{ideal} = \frac{(t_2 - t_4) - (t_1 - t_3)}{2} - \frac{d_{MS} - d_{SM}}{2} = offset_{estimated} - \frac{d_{MS} - d_{SM}}{2}$$

The offset estimation error is therefore the instantaneous delay asymmetry:

$$error = offset_{estimated} - offset_{ideal} = \frac{d_{MS} - d_{SM}}{2}$$

This can be developed further as:

(Equation 1)

$$\boxed{error = \frac{PDV_{MS} - PDV_{SM}}{2} + \frac{Delay\_Asymmetry}{2}}$$

with,

$$PDV_{MS} = d_{MS} - d_{MS\min}$$

$$PDV_{SM} = d_{SM} - d_{SM\min}$$

$$Delay\_Asymmetry = d_{MS\min} - d_{SM\min}$$

Within **Equation 1**, the first term represents the impact of PDV on PTPV2 performance and the second term the impact of delay asymmetry. Delay asymmetry, as defined above, while remaining deterministic can be measured and injected into the Slave computation module in order to reach a better time accuracy. PDV however has random components which are real issues (it is to be noted that in the above equations, the random components of delay asymmetry has been arbitrary integrated to the PDVs. These random components could be also integrated to the delay asymmetry as another definition of this latter parameter).

From a general perspective, the Packet Delay Variation has different random components [3,4], namely:

- The transmission Delay (variation): transmission delay is the result of the velocity of the signal between two endpoints of a transmission link and the distance between these two endpoints. The transmission delay variation is due to many factors depending on the transmission technology employed (e.g. wireless, wire-line, etc). For example on a copper link, temperature variations can modify the transmission delay by shrinking or extending the transmission distance.
- The processing delay (variation): the processing delay is the delay resulting from the processing of a timing packet within a Network Element (NE). It strongly depends on the NE hardware and software. For example, the processing delay variation can result from a specific CPU (Central Processing Unit) configuration: a same CPU on a given port could be in charge of in-going and outgoing packets. Consequently the traffic load in one direction can have a detrimental effect on the other direction.
- The buffering/queuing delay (variation): buffering or queuing delay is the total amount of waiting time of the timing packet within different buffers or queues in a given NE before being processed and finally transmitted to the next NE on the communication path. Buffering delay variation is partially

due to the “competition” between timing packet flows and other packet flows, and between different timing packet flows themselves according to their relative arrival time at different waiting queues and the priority policies implemented.

It is to be noted that in a telecom environment, the buffering/queuing delay (variation) is generally the main component of the PDV.

5 Deployment scenarios

Considering deployment scenarios of the synchronization network, three representative use–cases will be distinguished within Tiger II synchronization task, thanks to two high level variables:

- Backhaul infrastructure ownership: the Mobile Operator (MO) owns the backhaul infrastructure or not,
- SyncE availability within the MO owned infrastructure: SyncE fully or partially deployed.

5.1 Mobile Operator operating all backhaul segments and SyncE fully deployed

The first considered deployment scenario is illustrated by Figure 2 below.

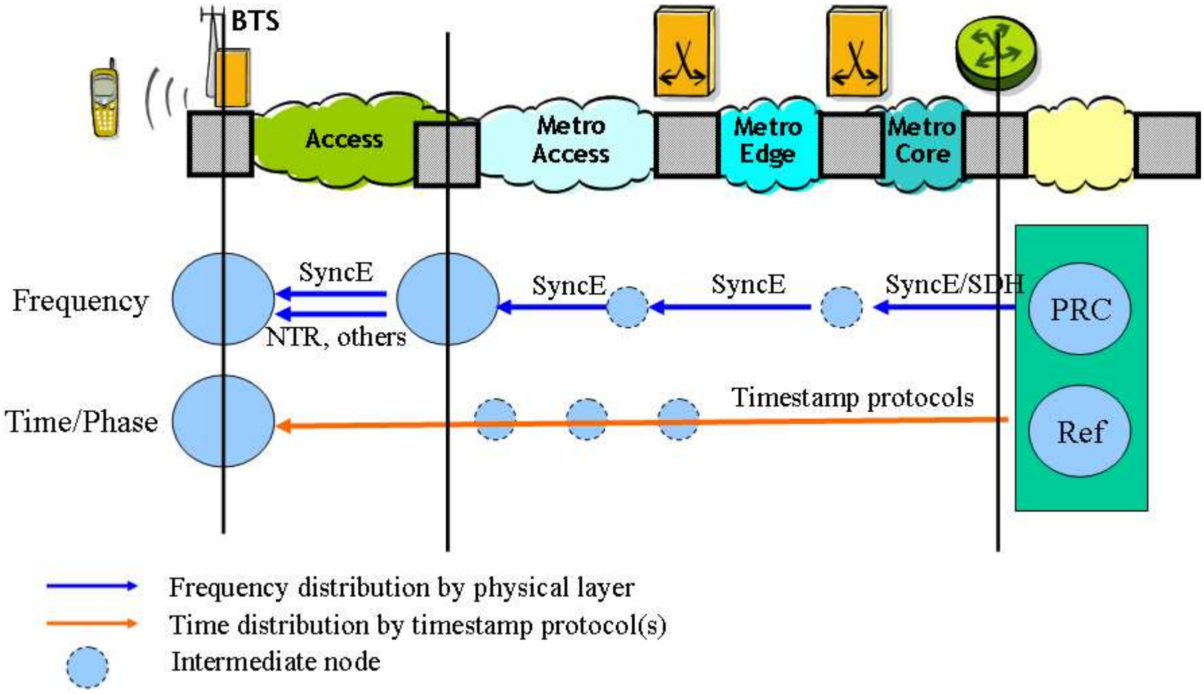


Figure 2 – MO owning all backhaul segments with SyncE fully deployed

This is an ideal scenario or a targeted architecture where SyncE is fully deployed for a syntonization purpose. The Mobile Operator (MO) owns all the backhaul segments and thus has the flexibility to implement on-path support nodes in order to optimize the protocol performance with respect to end applications. Both time and frequency references are likely to be co-located and time distribution paths are likely to follow frequency distribution paths in order to take advantage of the syntonization supports.

5.2 Mobile Operator operating all backhaul segments and SyncE partially deployed

This second considered deployment scenario is illustrated by Figure 3 hereafter.

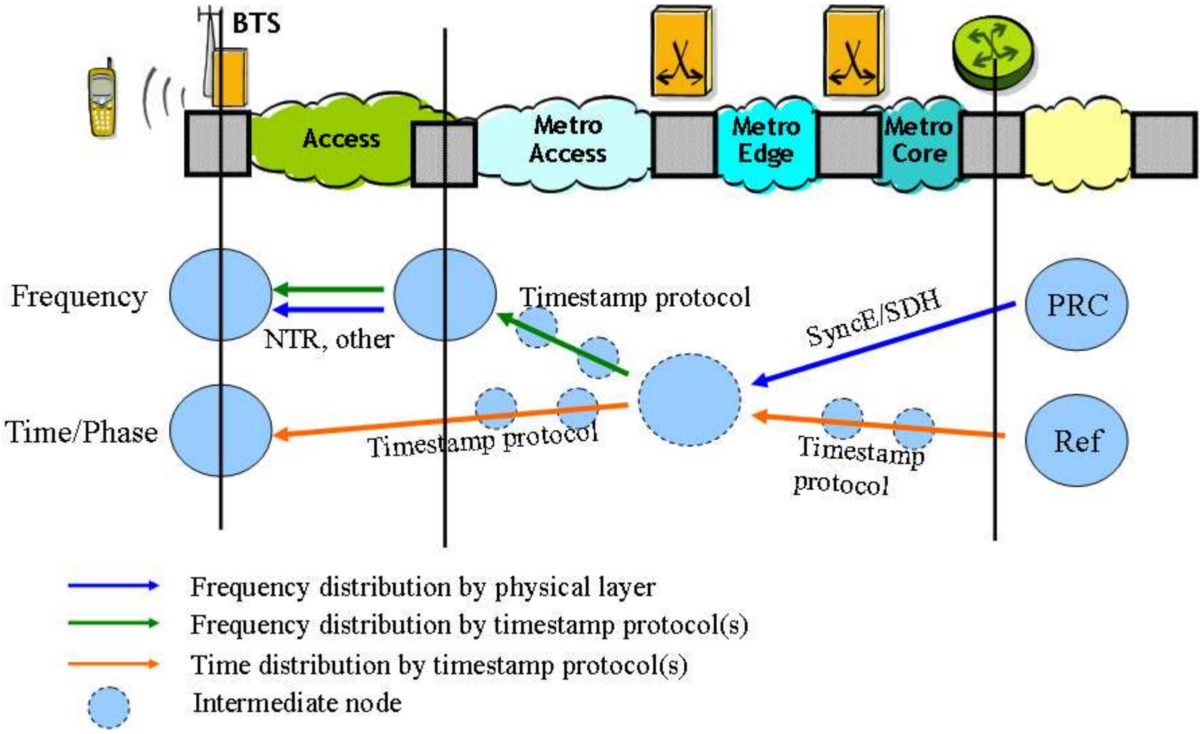


Figure 3 – MO owning all backhaul segments with SyncE partially deployed

This scenario is more realistic as it considers migration constraints for which SyncE would be partially deployed. Similarly to the previous case, the MO takes the great advantage of having control over the entire backhaul network. In order to distribute the reference frequency towards end applications, the MO can make use of timestamp protocols (e.g. PTPV2) in order to bound different physical layer based frequency distribution domains together or to extend one of them for reaching end applications. Obviously, the MO can also make use of timestamp protocols in order to distribute a time reference.

Remark: when making use of a timestamp protocol for distributing both frequency and time, respective time and frequency distribution topologies do not need to be congruent within the entire network segment while considering their different respective dependencies (PDV dependency for a syntonization purpose and both PDV and delay asymmetry for a time synchronization purpose).

5.3 Mobile Operator not operating all backhaul segments

The last considered deployment scenario is illustrated by Figure 4 below.

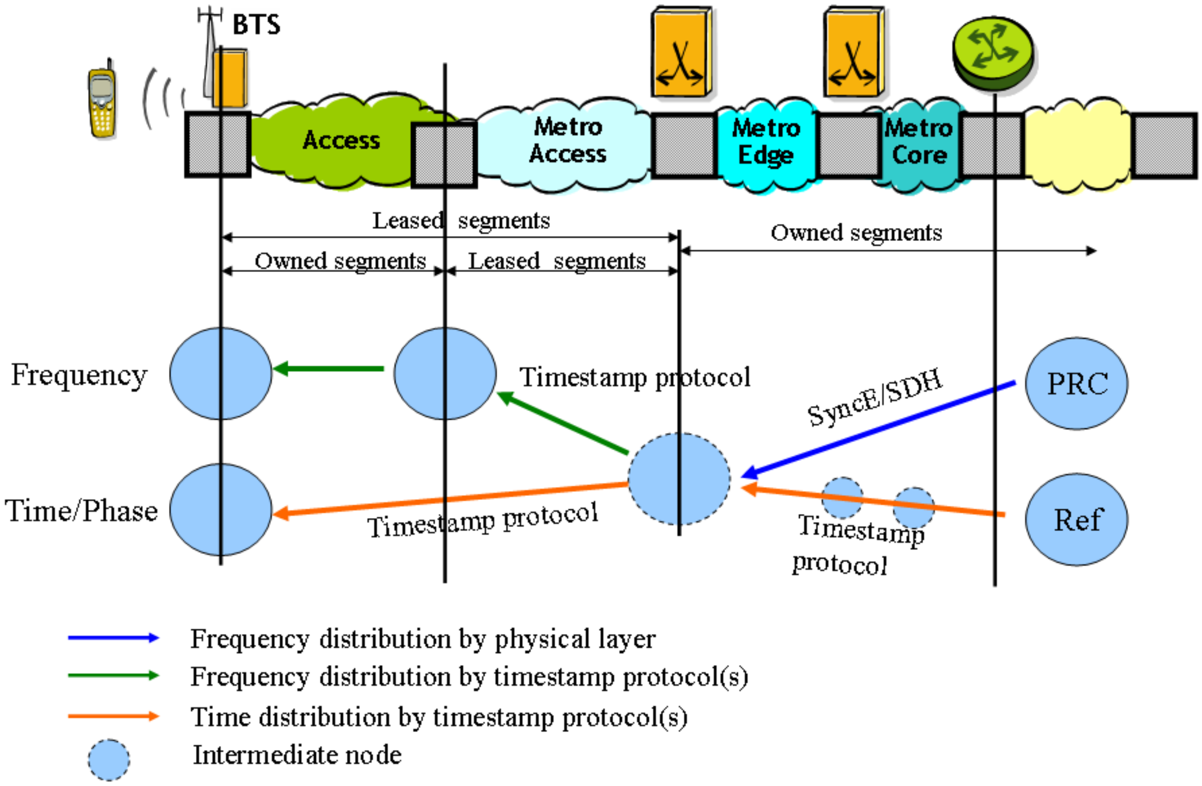


Figure 4 – MO not owning all backhaul segments

This scenario covers numerous cases where the MO does not own all backhaul segments or does not have any backhaul network at all. Within leased parts of the backhaul network, the MO has to make use of timestamp protocols in order to distribute both time and frequency. The MO has no visibility on the potentially available on-path support that can be implemented along the leased segments. End-to-end approach should be adopted in this case.

5.4 Operator deployment focuses

This chapter presents high level deployment priorities as foreseen/viewed by France Telecom. The majority of the proposals in this document can be applied to a large number of deployment scenarios. However, a particular focus will be put on the following choices when it comes to performance comparison, and especially to simulations and/or test results which will be thoroughly exposed in the future D24:

1. <u>For frequency-only distribution:</u>

1.1 Physical layer based methods (e.g. SyncE): this is the preferred solution for distributing frequency. Physical methods are standardized and largely known, thus will not be investigated within Tiger II research project.

1.2 Packet-based methods:

1.2.1. Packet-based methods without any hardware support (e.g. BC/TC) from the network: it is an interesting case especially for inter-carrier frequency distribution cases (although physical layer based solutions may apply also to this case, e.g. if a SyncE service is offered as part of the leased line). Also, this case is relevant for supporting gradual network migration towards physical layer based technologies as a targeted architecture.

1.2.2. Packet-based methods with partial hardware support (e.g. BC/TC) from the network: regarded by FT as less important than other approaches. Indeed, if some hardware support is required from the network for frequency distribution, then Sync-E is seen as the preferred solution.

1.2.3. Packet-based methods with full network hardware support (e.g. BC/TC): regarded by FT as less important than other approaches. Indeed, if some hardware support is required from the network for frequency distribution, then Sync-E is seen as the preferred solution.

Summary regarding frequency-only distribution: regarding the deployment of new hardware support, physical-based methods such as Sync-E are considered as the preferred approach to guarantee the performance delivered to the end application. In other cases, the use of packet based methods in an end-to-end approach is preferred.

2. F<u>or both frequency and TOD distribution</u>:

2.1 PTPV2 (alternatively NTP) without Sync-E and without network hardware support (e.g. BC/TC): this is considered by FT to be a challenging case if the purpose is to meet the stringent LTE phase/time requirement with µs accuracy objective.

2.2 PTPV2 (alternatively NTP) without Sync-E and with partial network hardware support (e.g. BC/TC): same remark as 2.1. This is considered as a challenging deployment.

2.3 PTPV2 (alternatively NTP) without Sync-E and with full network hardware support (e.g. BC/TC): this is identified as possibly an interesting case where prolonged holdover is not an issue.

2.4 PTPV2 (alternatively NTP) with Sync-E and without network hardware support (e.g. BC/TC): same remark as 2.1. This is considered as a challenging deployment.

2.5 PTPV2 (alternatively NTP) with Sync-E and with partial network hardware support (e.g. BC/TC): same remark as 2.1. This is considered as a challenging deployment.

2.6 PTPV2 (alternatively NTP) with Sync-E and with full network hardware support (e.g. BC/TC): this is identified by FT as probably the best technical approach that would certainly allow to guarantee both required TOD and frequency performance under normal operation conditions as well as holdover.

2.7 Link-by-link TOD distribution using ESMC messages: this case is identified to be an interesting solution for networks where Sync-E is already deployed as it will save the need to use PTP in addition.

Summary regarding frequency and TOD distribution: according to France Telecom**,** physical layer frequency distribution should be used to support time distribution (to guarantee time holdover performance). A fully deployed hardware support is also seen as a targeted architecture in order to meet stringent LTE phase/time requirement with µs time accuracy objective. TOD distribution can either be accomplished end-to-end using PTPV2 (or NTP) with full network hardware support or link-by-link using the ESMC messages (if Sync-E is already deployed).

6 Improving Timing-over-Packet (ToP)

In the context of frequency distribution, timestamp packet protocols, such as PTPV2 (and NTP), are very dependent on Packet Delay Variation (PDV) in term of performance as previously highlighted in chapter 4.3.2. After a general discussion on possible existing mechanisms at different levels of the synchronization topology (master, network elements, and slave) to dampen PDV effects, this chapter will propose a new distributed filtering mechanism which demonstrates the interworking benefits by considering all previously cited levels. Improvements should also be brought into PTP protocol/domain in order to ensure inter-working with SyncE in term of synchronization status (e.g. SSM).

Even though more focused on frequency distribution, it is to be noted that some discussions and new concepts in this chapter are rather generic and can also be applied to the distribution of time or phase as well.

6.1 PDV management discussions

Although the PDV is essentially induced by the network, it is possible to dampen its effect at different levels of the hierarchical synchronization topology [5], meaning at the master level, the network level and the slave level.

6.1.1 Master approach – emission profile

At the master level, the emission profile is the major feature to investigate in order to dampen PDV effects on ToP performance. Within the emission profile, different parameters can be tuned. The following are some examples:

- Timing packet rate: this is one of the main parameters enabling to manage PDV effects. Indeed, a higher number of samples (timing packets) per time unit allow for improving the frequency accuracy at the Slave level. Figure 5, based on equations developed in [6], illustrates in a first order approach this purpose, with a common 5 mHz Phase Locked Loop (PLL) filtering stage.

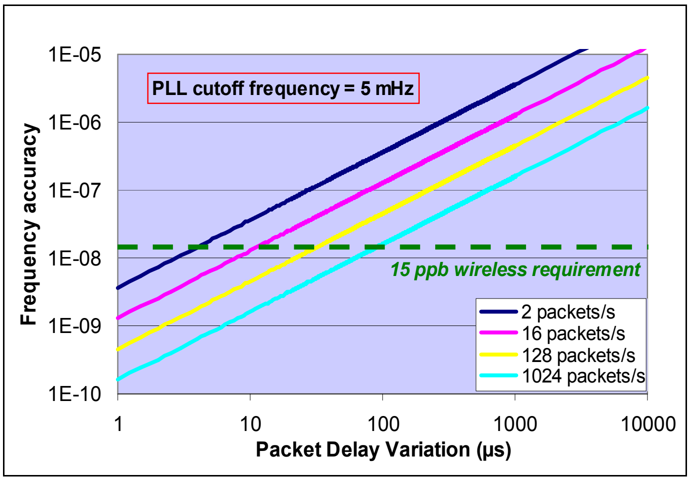


Figure 5 – Frequency accuracy according to the timing packet rate for a 5mHz PLL filter

For instant, a factor-10 improvement in term of frequency accuracy can be observed with an increase in emission rate from 2 messages/s to 128 messages/s (note that IEEE 1588V2 specifies a maximum message rate of 128 messages/s). Targeting an often-mentioned 15 ppb frequency accuracy at the network level for wireless requirements (e.g. WCDMA, LTE), a 100µs PDV value appears to be a recommended limit.

- Randomization: as it comes to syntonization, one often makes use of a periodic packet stream (i.e. one-way periodic Sync messages). In this case, it is demonstrated in [7] that packet streams with similar transmission frequencies compete against each other leading to additional PDVs. This phenomenon is often referred to as a "beating effect". In order to solve such an issue, a simple but powerful method consists in randomizing the transmission instant of timing packets around the nominal periodic value.

6.1.2 Network element approach

PTPV2 introduces different concepts which allow for reducing PDV at the network element level. These concepts are the Hardware Timestamping (HTS), the Boundary Clock (BC) and the Transparent Clock (TC). Those are discussed below:

- Hardware Timestamping (HTS): its principle consists in generating the message timestamp as close as possible to the transmission medium (physical layer) in order to work-around PDV due to buffering delays and processing delay variation induced by upper protocol layers. This is illustrated by Figure 6 below. The timestamp point is often implemented at the Media Independent Interface (MII) which is located between the MAC layer and the PHY layer.

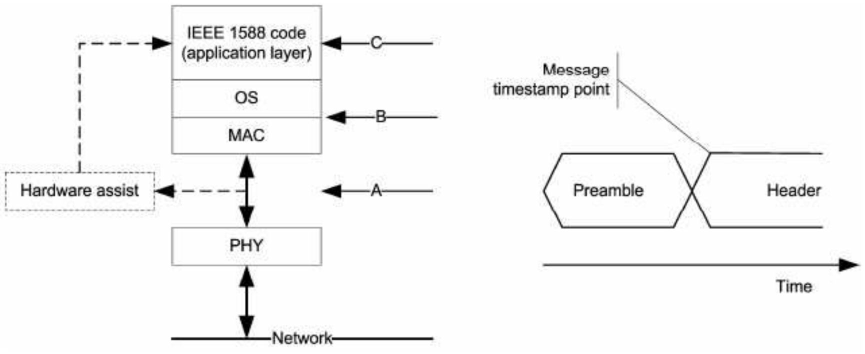
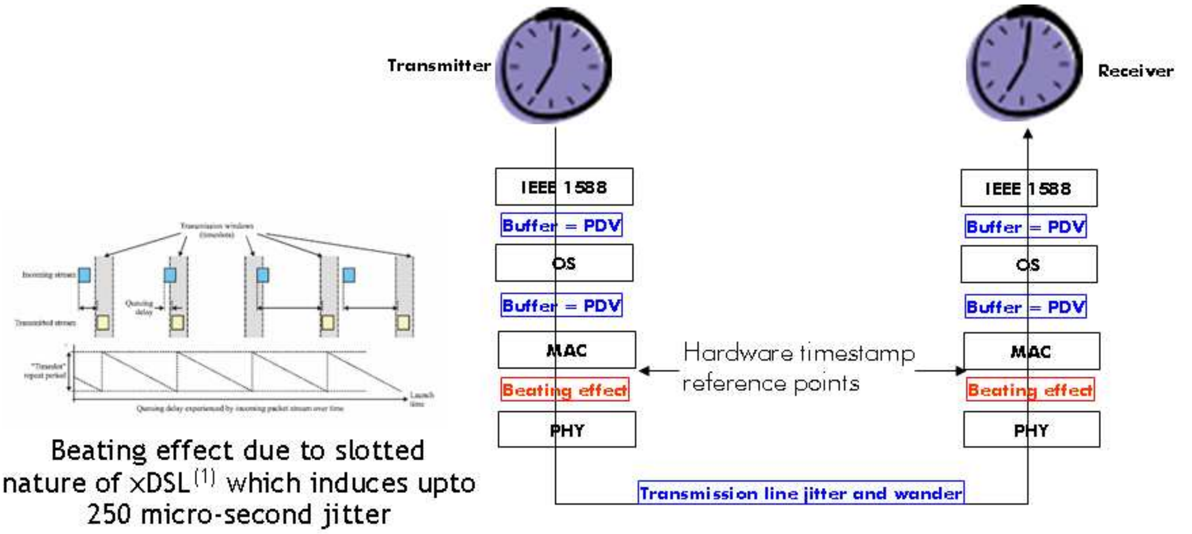


Figure 6 – IEEE 1588V2 Standard timestamp generation model

The limit of the HTS seems to be reached when transport technologies introduce important jitters at the physical layer. These technologies are, for instant, xDSL or point-to-point microwave technologies which are cost-effective technologies largely used for the mobile backhauling. Figure 7 illustrates the case of xDSL where the time-slotted structure of the physical layer introduces a "beating effect" to the timestamp packets with a magnitude that can goes up to 250 µs. Thus, jitters introduced by the xDSL physical layer (e.g. under the MII interface) introduces an uncertainty of 250 µs to the timestamp generation. In order to overcome this uncertainty, it is necessary to move the hardware timestamp point very nearer to the transmission medium (e.g. xDSL sample level). The main difficulty is how to make a link between the physical signal (e.g. a given xDSL sample), taken as the timestamp significative instant, and the timestamp packet entity as there is no obvious direct connection between both entities.



(1) ITU-T G.8261/Y.1361 - Timing and synchronization aspects in packet networks – April 2008

Figure 7 – Hardware Timestamping limits ?

- Boundary Clock (BC): the BC does not forward PTP messages related to synchronization, signaling and the establishment of the master-slave hierarchy. It allows for segmenting a large synchronization network into small areas (within a PTP domain) where PDVs can be engineered and/or controlled within specified bounds. BCs serve as means to recover the reference time (or frequency) as accurately as possible before distributing the latter into next (per-PDV) areas of the synchronization hierarchy.

  Some issues have been observed with certain BC products in that they represent an accumulated phase error which grows rapidly with the number of cascaded BCs within a synchronization path. Figure 8 provides an illustration of such an issue which limits the scalability of BC deployment [8].

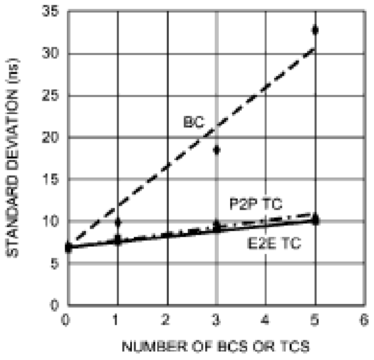


Figure 8 – BC accumulated error vs TC accumulated error

- Transparent Clock (TC): End-to-End Transparent Clocks (E2E TC) allows for correcting transit delays of the network elements whereas Peer-to-Peer Transparent Clocks (P2P TC) allow for compensating both transit delays and link delays. The first one allows to record, within the PTP message "correction field", its (cumulative) resident time within each traversed NE and allows for the Slave clock to compensate PDVs produced by queuing and processing processes. The second one measures not only the NE resident time but also link delays. Both information allows the Slave to completely correct uncertainties due to PDV and thus to reach accurate time/frequency references at the Slave level.

  Whilst TCs represent a very effective solution to fight out PDVs, they represent some non negligible challenges for network operators considering a full deployment. The first one is the cost related to the implementation of such devices within the NEs along all PTPV2 signalization paths. The second important one is rather a technical challenge linked to the fact that TCs represent an infringement to the layer separation principle. Indeed, for precision purpose, TCs require timing monitoring operations (e.g. hardware time stamping) being realized at the physical layer while reporting the monitoring results (e.g. packet resident time) at the PTPV2 packet layer. Moreover, TC normal operations make an infringement to tunneling principle which consists in never modifying transported data and related checksums.

  Nevertheless, those constraints have to be re-visited in the particular scenario of "green field" deployments targeting the distribution of both accurate time and frequency references.

Before the introduction of the above PTPV2 concepts, other mechanisms have been performed in order to overcome packet jitter effects. Some of these mechanisms are discussed below:

- Quality-of-Service (QoS) approach: it consists in differentiating timing signalization flows from other packet flows using QoS-build-in information within the transport header. This allows for prioritizing synchronization packets with regards to data packets, and avoiding a direct competition of the latters against the formers. QoS technique usually relies on parallel buffers which filling depends on scheduling policies.

  This technique allows to reduce the PDV induced by queuing effect for a given packet flow, especially for high traffic load conditions, but without suppressing this component completely [9].

  Indeed, the method does not fully prevent low priority data packets from competing with high priority timing packets. Even with separated queuing buffers, QoS technique cannot avoid competition at the physical medium when it comes to the serialization of packets onto the transmission line. For instant, the timing packet waiting times, induced by already departing Best Effort packets (from another queue) onto the transmission line (physical layer level), can greatly vary depending on the data packet length and its relative arrival time at the physical output medium. This phenomenon is often referred to as the (data) "jumbo packet" phenomenon. Another phenomenon that QoS technique cannot solve is the competition between timing packet flows within a same QoS priority. In case of periodic timing flows having the similar transmission frequencies, such a competition produces the "beating effect" [7].

  Chapter 6.3 will present more detailed discussions on QoS in PDV management.

- Delay equalization: this is a well-known technique especially implemented within ATM switches for the interconnection of different SDH/SONET transmission lines. The method consists in adding a time label in order to track the node delay experienced by ATM cells. At the output port, this makes it possible to “equalize”, via a buffer, node resident times of ATM cells originating from the dismantling of the same (SDH/SONET) input payload. Delay equalization is quite powerful to fight out packet/cell jitters related to both processing and queuing, as different packets/cells pertaining to the same flow undergo the same node delay.
  However, in this specific scenario (retiming at the bit level), delay equalization requires the distribution of a precise clock across all the NE ports. This pre-requisite supposes a high accurate, thus high cost, local clock especially if the latter is left into a free-run mode.

6.1.3 Slave approach – filtering mechanisms

Different filtering mechanisms (a.k.a. ranging mechanisms) are available at the Slave clock level. Dealing with syntonization concerns, two traditional mechanisms are typically used. These latter are the Moving Average (MA) and the Exponentially Weighted Moving Average (EWMA) mechanisms. Both consist in estimating the average inter-arrival time of timing packets in order to derive the nominal frequency. Both methods assume that the timing packet delay distribution as stationary and that these delays are independent and identically distributed. Unfortunately, this is not the case in real network. For instant, it has been demonstrated that EWMA method becomes rapidly inefficient as the correlation between successive packets increases [10,11].
Thus, new paradigms in packet filtering presently focus on packet experiencing minimum (respectively maximum) delays [12,13]. New related metrics, such as MinTDEV [14], etc, have been defined for this purpose. A performance comparison between these mechanisms has been started and provided through [15]. But, no overall consensus has been found and thus no decision has been taken within standardization organization in terms of recommendation.

6.2 New distributed ranging mechanism

As discussed in the previous Chapter 6.1, there are numerous manners to manage PDVs within telecom networks. A limited set of powerful approaches have been presented with their strengths and weaknesses. Stating that all improvements performed at the Slave level to manage PDV could not meet all deployment scenarios and application requirements, network-based solutions have to be considered with careful studies on deployment cost. In this context, a new distributed mechanism is proposed here. This new mechanism particularly enables to increase timing packet (e.g.PTPV2) rate - and thus to improve ToP performance for a given PDV profile - with a limited impact on the available network bandwidth.

6.2.1 Principle description

The concept relies on the impact of the timing packet rate and the dynamic nature of the Packet Switched Network (PSN) where network traffic load can evolve rapidly. This new approach consists in a distributed selection/filtering of relevant (low delay) timing packets along the synchronization path thanks to a given synchronization requirement at the receiver level. The idea particularly enables to increase the timing message rate with a limited impact on the consumed bandwidth. In a simple configuration, each filter (including the receiver) has the ability to trigger an increase of the timing source message rate according to a provisionable timing message rate threshold. Figure 9 below depicts the main principles of this approach.

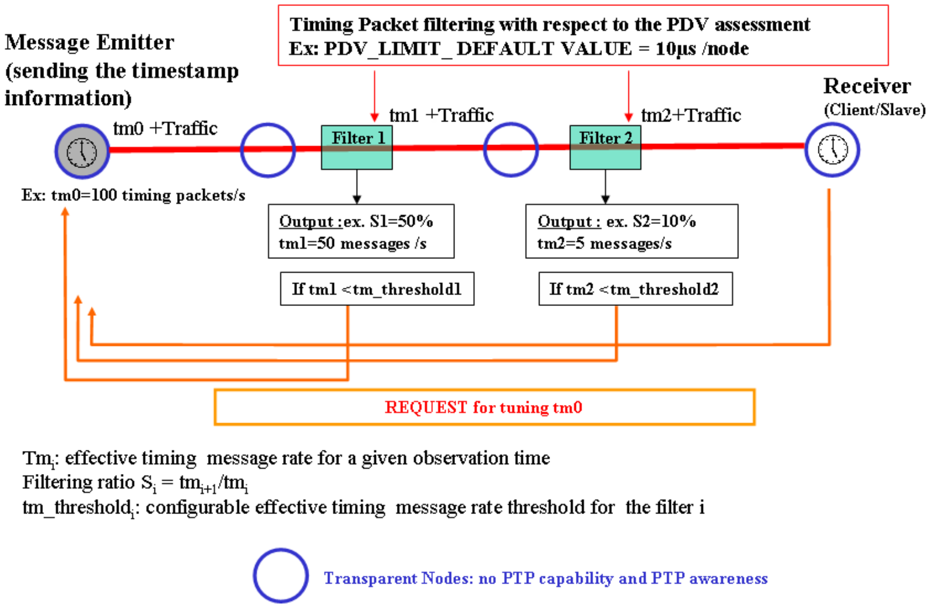


Figure 9 – Distributed ranging mechanism

In such a representation, “transparent nodes” are intermediate nodes which do not implement any timestamp protocol (or ToP such as PTPV2) features. Filter modules have the ability to recognize timing (e.g. PTPV2) packets and to proceed to the ranging process.

For each aforementioned filter module, a ranging process is applied in order to select packets demonstrating low PDV (e.g. based on packets experiencing minimum delays). If the number of these selected packets is lower than a certain provisioned threshold (e.g. tm1 for filter 1), the filter module requests the Master for increasing the timing packet emission rate. In the opposite way, a second threshold can be provisioned so that the filter module can ask the Master for decreasing the timing packet emission rate when the number of selected packets is found to be above this second threshold. This method guarantees that the Slave receives sufficient relevant packets during its filtering process in order discipline its local oscillator.

In case of PTPV2 deployment, one possible way to change the Master timing packet emission rate is to use a management message with a PORT_DATA_SET management Type Length Value (TLV) data field (this latter includes other TLVs defining different message emission frequency values, such as logSyncInterval). Alternatively, filter modules can make use of the REQUEST_UNICAST_TRANSMISSION TLV to modify the Master timing packet emission rate when unicast profile is deployed.

In order to optimize the consumed bandwidth by downstream Network Elements (NEs), non-selected synchronization packets having their PDV range over a pre-defined bound can be dropped and thus not forwarded towards the Slave.

Facing heavy traffic load conditions, the presented approach enables to tune the individual timing packet filtering ratio in order to ensure the best synchronization performance in a degraded state, for which the synchronization requirement at the Slave level is not reachable. This “less-bad state” approach could be relevant is some congestion scenarios for which the synchronization topology reconfiguration time plays a significant role.

From an implementation perspective, the timing packet filtering functionality could be integrated either into a NE, or into an external module placed onto the path between two network nodes.

It is to be noted that these distributed filters can take advantage of an external syntonization for a precise monitoring of the selection process but similarly to the TC functionality, this does not appear critical (some ppm frequency accuracy should be sufficient to cover stringent synchronization requirements).

6.2.2 Proposal benefits

In order to illustrate the benefits of the distributed ranging mechanism, two representative PDV examples are taken (PDV here is defined with regards to the minimum delay) with related Probability Density Functions (PDF) as depicted in Figure 10. These PDF examples, reflecting high and low traffic load conditions as defined by the ITU-T G.8261 recommendation [8], arbitrary follow a Gamma density function already discussed within the ITU-T [16]. For low and high traffic loads, the computed standard deviations are respectively 7.1 µs and 20 µs with regards to Figure 10 examples).

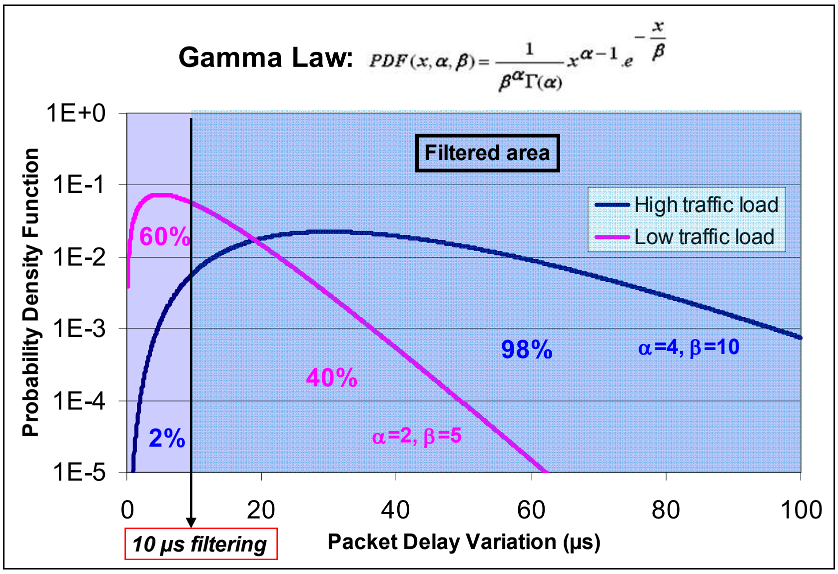


Figure 10 – PDV PDF for high and low traffic load conditions

An arbitrarily configured PDV threshold filtering of 10µs per filter module is considered. This scenario particularly means that, without any filter module, timing packet demonstrating a PDV over this 10 µs reference value have a very poor probability to be retained by the first filtering stage of the end receiver packet clock. The following discussions take the first filter module as an example but one can reiterate the arguments through successive filter modules downstream.

Within this configuration, a low traffic load implies a low PDV range and a better probability (e.g. 60%) to have an arriving timing message selected by the first filter module. In this case, the number of messages selected should be sufficient (above the provisioned threshold) and thus no feedback mechanism towards the sender (e.g. master clock) is needed in order to increase the emission rate. In this scenario, one can observe a 40% timing-message bandwidth saving at the downstream link next to the first filter module.

At high traffic load condition, the probability to have a timing packet selected by the first module is strongly decreased (e.g. down to 2%). Thus the selected packets are not to be of sufficient number with regards to the module provisioned threshold. As described in the previous chapter, one possible policy in this case consists in loosing synchronization constraints (e.g. degraded state) in order to avoid traffic load increase. Alternatively it can be decided to trigger the feedback mechanism for an increase of the timing message transmission rate. This second process has to be carefully calibrated according to the congestion state margin. Indeed, such a mechanism can significantly deteriorate the node state with a final performance improvement lower than expected (e.g. linked to packet loss). Congestion control and/or rerouting are consequently other techniques to be considered for solving such an issue. Nevertheless, they have to be well engineered by network operators with regards to:

- the priority allocated to the synchronization traffic;
- the policies which drive topology changes (avoiding congestion points), as they usually involve signal transients impacting the synchronization performance.

6.3 QoS-based methods and related performance

This study is on-going and will be thoroughly described within D24.

6.4 ToP as an extension for physical-layer frequency distribution

6.4.1 Deployment context

In term of frequency-only distribution, SyncE is a straightforward solution that meets the stringent frequency requirements of wireless networks. Derived from SDH/SONET technology, SyncE takes advantage of the experience of the former which has demonstrated very good performances within very large deployed networks. However, due to the principle of distributing frequency link-by-link, SyncE requires that all equipments in a synchronization chain to be 100% compatible with the SyncE standards as one incompatible element disrupts the chain. Many events can contribute to the risk of this synchronization chain disruption such as interoperability between SyncE equipments and mis-configuration (e.g. configuration of the ESMC).

In the context of migration to SyncE, certain parts of the network will take more time to migrate due to cost constraints. Leased lines can also be used to link different parts of the backhaul network together (e.g. MO does not own all backhaul segments). In this latter case, physical-based frequency distribution scheme is broken and ToP could be used to extend this distribution across leased lines.

Within this context, ToP is used:

- to interconnect SyncE clouds together, or
- to extend the SyncE cloud to reach the final applications (e.g. base station).

This is illustrated by Figure 11 below.

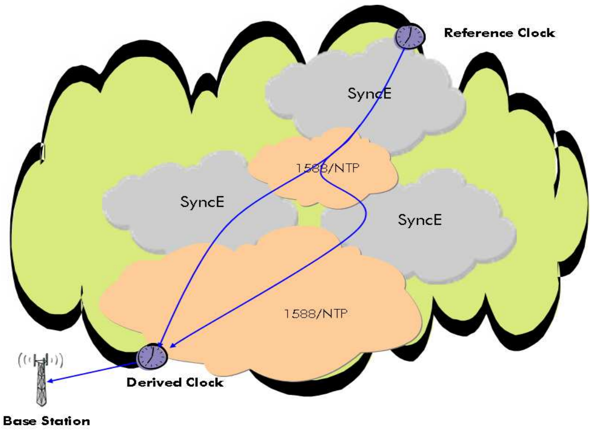


Figure 11 – SyncE inter-connection and extension by ToP

An identified requirement here is to ensure the continuity of traceability and loop avoidance mechanisms across the ToP domains.

6.4.2 Proposals

The following discussions will focus on 1588V2 protocol, but they can also be applied to NTP as well.

The first proposal meeting the aforementioned requirement is depicted in the Figure 12. This latter relies on the configuration of a 'SSM tunnel' carrying the Status Synchronization Message (SSM) information through the 1588V2 domain, ensuring E2E SSM connectivity between two SyncE domains. In such a configuration, intermediate nodes in the 1588V2 domain carry transparently the SSM information (SSM information is not processed). The tunnel establishment should take into account the underlying (with regards to 1588V2) transport technology (e.g., VLAN for Ethernet, etc).

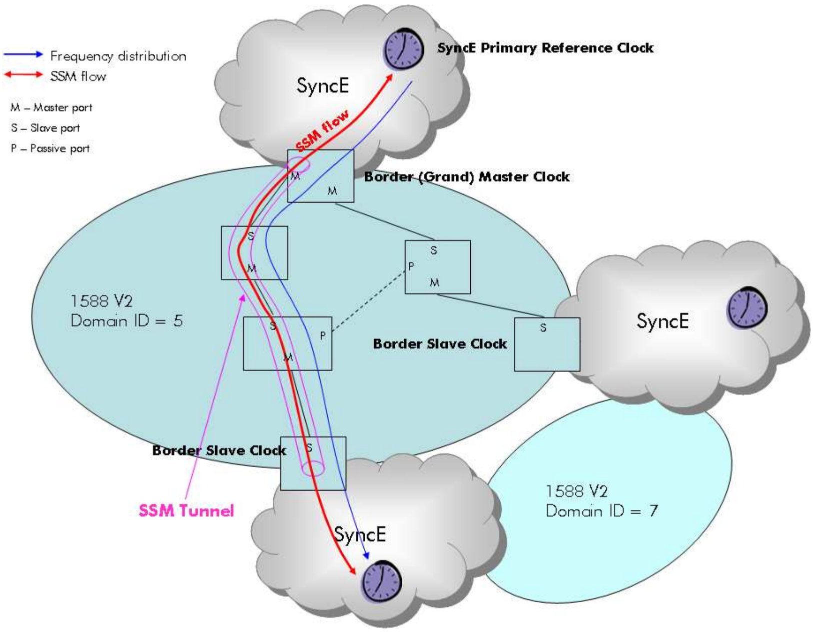


Figure 12 – SSM Tunnel across ToP domain

To enable a stronger integration between 1588V2 and SyncE, an interaction between the SSM carried by the ESMC and 1588V2 signaling within the 1588V2 domain is required, so that the 1588V2 domain behaves like a virtual SyncE node. This means for instance that the intermediate 1588V2 nodes are able to switch the SSM Quality Level to a DNU value in the case of failure detection. One solution for such integration is to encapsulate the SSM value within the 1588V2 signaling. The wording of ‘SSM encapsulation’ is used in this case to clearly differentiate this approach from the previous ‘SSM tunnel’ one. Within the encapsulation context, SSM information is not conveyed transparently across 1588V2 domain. Instead, it can be read, modified or completed by the intermediate 1588V2 entities.

The encapsulated SSMs naturally take the same route as the related synchronization distribution packet flow inside the 1588V2 domain, allowing intermediate 1588V2 nodes for monitoring and reporting any case of the synchronization signal exceeding the “synchronization noise tolerance” (jitter and wander budget). The Quality Level (QL) received by the SyncE endpoints enables to trigger the selection of the best synchronization signal following the standard SyncE behavior. Figure 13 allows for illustrating such a concept.

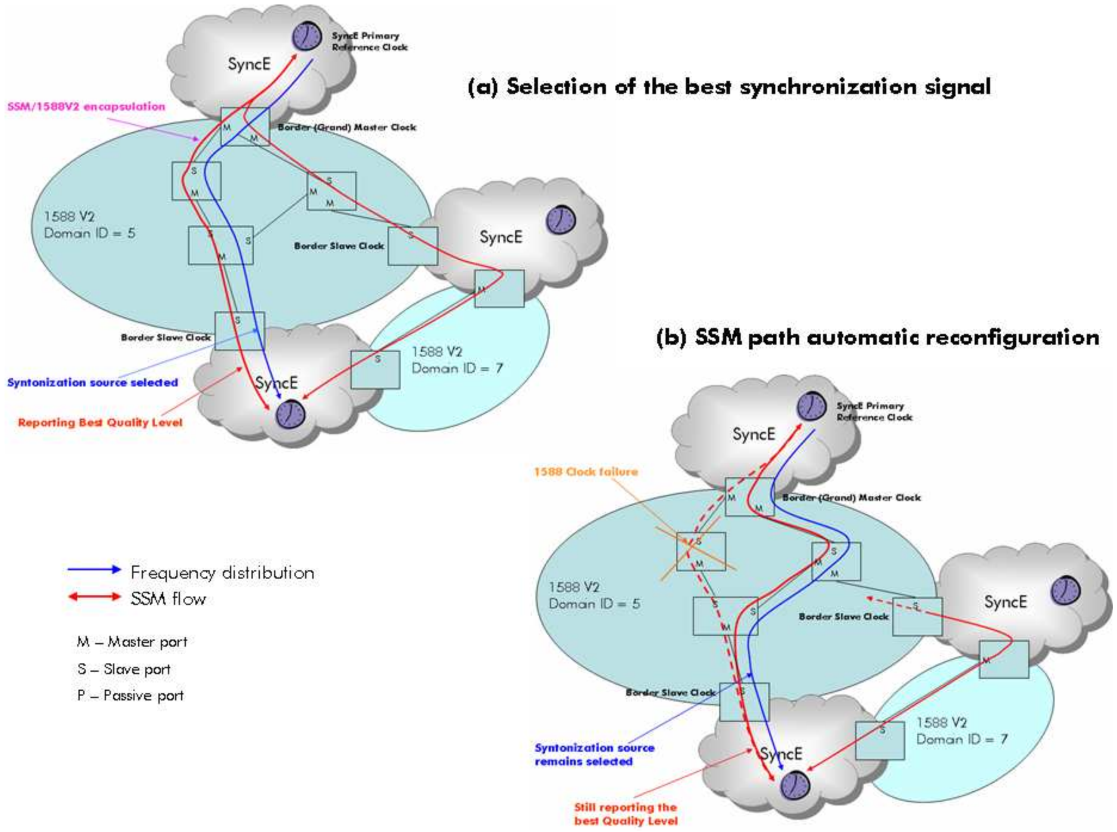


Figure 13 – SSM encapsulation with automatic reconfiguration

Finally, the encapsulated SSMs allow for taking the full advantage of the self-healing capability of the 1588V2 domain (e.g. default Best Master Clock Algorithm or BMCA), as illustrated in Figure 13b. At the detection of a failure on an intermediate node, the clock hierarchy is automatically reconfigured thanks to dedicated algorithms (e.g. default BMCA). The encapsulated SSMs conveyed by 1588V2 messages take the same new path.

7 Time or phase distribution with network support

Within this document the terminology “network support”, widely used within the synchronization community will refer to either of the followings:

- Link-by-link termination and generation (support at the physical layer or very close to it)
- Any external synchronization information source like Sync-E or TDM,
- Any other special mechanisms, embedded in the packet network, that help to distribute the synchronization information, e.g. QoS, 1588V2 Transparent Clocks, etc

Two kinds of network support will be considered within Tiger II project: IEEE 1588 supports (e.g. BC, TC) and SyncE as a syntonization support. This section will particularly focus on architecture and deployment considerations while making use of such network support.

As a first step, IEEE1588V2 TC and BC support will be considered alone, and then their collaboration with SyncE support will be investigated in a second step. Finally, as a third option, a complete Sync-E frequency and TOD distribution will be discussed.

7.1 IEEE 1588V2 support: Boundary and Transparent clocks

7.1.1 Boundary and Transparent clock overview

Apart from defined Ordinary Clocks (Master and Slave clocks), IEEE 1588V2 standard provides two types of network support, Transparent Clocks (TCs) and Boundary Clocks (BCs).

As briefly depicted in 6.1.2, the role of the BC is to maintain the timescale of the domain, and it contains multiple PTP ports. All output ports are synchronized with the local clock, which is locked to an input port reference (depending on the Best Master Clock (BMC) algorithm discussed in the next section.

The main functions of a 1588v2 BC are:

- to delineate the 1588v2 domains by blocking the transmission of the main part of the 1588 messages,
- to provide “regeneration points,” thus eliminating the large latency fluctuations typically generated by routers and similar devices,
- to distribute time in a point/port to multi-point/port mode manner, and
- to ensure the synchronization of IEEE 1588v2 clocks across heterogeneous networks (e.g., mapping functions between protocols of different layers).

As complementary tool allowing for fighting out network dependencies, TCs are also proposed in the IEEE 1588V2 standard. Functionalities of a TC depend on its type:

- an End-to-End Transparent Clock (E2E TC) is dedicated to measuring the network element delay experienced by PTP messages while traversing a network element (e.g., a router).
- a Peer-to-Peer Transparent Clock (P2P TC) provides link propagation delay measurement in addition to the measurement of the network element transit time.

Important remark: note that in case of a full P2P TC deployment (all network elements are supported by a P2P TC), the real-time correction of network dependencies allows for using a forward timing flow from the Master to the Slave for a time distribution purpose (i.e. one-way signaling instead of two-ways signaling). Moreover for such a configuration, multicast mode is a realistic option regarding inherent packet replication capabilities of transport networks (such as Ethernet switches).

As specified within the related standard, TCs and BCs can be syntonized by external means. This capability will be particularly investigated in section 7.2 , focusing on SyncE as the external syntonization means.

Clock Hierarchy:

The IEEE 1588v2 standard organizes clocks into a master-slave hierarchy by running a distributed algorithm called the Best Master Clock algorithm. An instance of this algorithm is run at each clock, which enables identification of the highest quality clock by comparing the clock features, called a “dataset”. Unlikely to TCs, BCs are “active elements” within the Master-Slave hierarchy and consequently actively participate into the Best Master Clock algorithm.

The highest ranking clock, the so-called “grandmaster clock”, synchronizes all other slave clocks. If the grandmaster clock is removed from the synchronization network, or if its characteristics change in such a way that it is no longer the best clock available, another clock hierarchy (with another grandmaster) is computed by the BMC. Thus, the BMC algorithm provides a self-healing mechanism and a building block for a simplified management of synchronization.

Usually, this functionality is deactivated since it does not work very well in many specific telecom deployments, for instance, when the BMC has to “decide” which of two master ports with direct primary reference source traceability should be the grandmaster port. Other issues can occur during re-configuration transients, such as when a master port in failure (and thus in the holdover mode) is switched into the slave state but it is still needed as a master, for instance, while feeding the synchronization of a CES tunnel.

As depicted in the 1588V2 specification, this algorithm is also not robust enough regarding transmission loops of synchronization packets. If activated, the use of a robust network protocol that avoids communication loops, such as the IEEE 802.1D spanning tree protocol, is recommended to support the BMC algorithm.

7.1.2 Deployment considerations: no or full IEEE 1588V2 hardware support?

While talking about Transparent and Boundary clock supports, the first question to deal with is related to the support coverage (from a network with no BC/TC support to a full supported network with TC or BC capabilities on each nodes). Focusing on the distribution of a time reference, **is it recommended to have a full Transparent clock/Boundary clock deployment?**

One way for addressing such question consists in investigating the distribution of time through incognizant nodes (nodes without any PTPV2 capability) from a scalability perspective. Targeting a

given time accuracy requirement at the slave level, the incognizant node number tolerance has to be investigated.
As discussed in section 4.3.2, time distribution error is half of the delay asymmetry between Master-to-Slave and Slave-to-Master message communication paths. Thus targeting 1µs time accuracy, the averaged delay asymmetry during the Slave observation time window has to be below 2µs.

This is a very low value, especially while considering the ITU-T G.823 synchronization mask demonstrating a 2µs wander, or the time to transmit a packet with a 1548 bytes length at 1 Gbit/s (12µs).

While talking about packet delay variation and delay asymmetry, 3 components have to be considered:

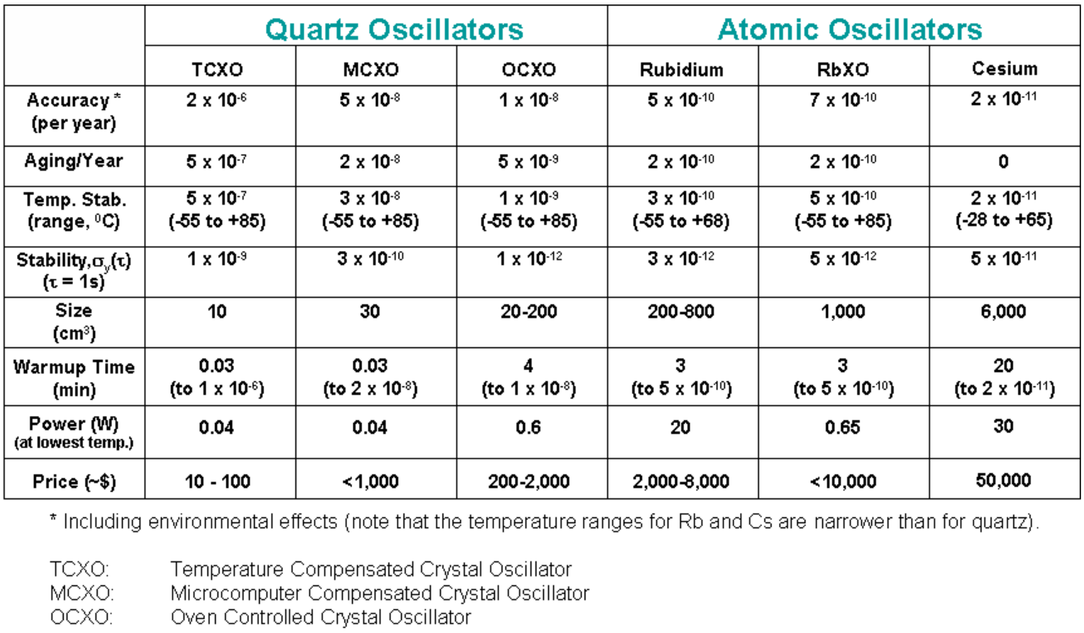
- A threshold delay: it means that over long observation time (much higher that the slave observation time, Master-to-Slave or Slave-to-Master packet delays demonstrate a clear and stable floor. This is quite true for most scenarios. For such a component, the related delay asymmetry is not filterable and cautions have to be taken to manage or to provision it.
- A delay following a low frequency evolution law: it means during the (ranging) observation time the minimum packet delay follow a low frequency variation law. This variation – that could be met for some beating effects or for some environmental conditions- is typically very hard to filter by packet filtering algorithms at the slave level,
- A statistical delay: in this case packet delays follow an uniform statistical distribution for which each packet is an independent sample (no correlation between packet delays). Fastest packets can be filtered from the rest of the traffic in order to reach the minimum threshold delay (first component) if floor delay variation is not an issue (second component).

Packet filtering and Slave oscillator considerations:

Regarding this former discussion, it clearly appears that packet filtering algorithms deployed at the Slave level are of major importance. These latter inherently depend on the oscillator features.

Table 2 provides a comparison between different oscillators commonly used in the industry, based on most used characteristics such as frequency accuracy, frequency stability, ageing, etc and an order of costs [17].

Remark: it is to be noted that actual prices are not publicly advertised. Thus, the below numbers do not fully reflect the real prices on the market but their number orders allow for establishing a priori valid conclusions.



	Quartz Oscillators			Atomic Oscillators		
	TCXO	MCXO	OCXO	Rubidium	RbXO	Cesium
Accuracy * (per year)	2 x 10^{-6}	5 x 10^{-8}	1 x 10^{-8}	5 x 10^{-10}	7 x 10^{-10}	2 x 10^{-11}
Aging/Year	5 x 10^{-7}	2 x 10^{-8}	5 x 10^{-9}	2 x 10^{-10}	2 x 10^{-10}	0
Temp. Stab. (range, ^{0}C)	5 x 10^{-7} (-55 to +85)	3 x 10^{-8} (-55 to +85)	1 x 10^{-9} (-55 to +85)	3 x 10^{-10} (-55 to +68)	5 x 10^{-10} (-55 to +85)	2 x 10^{-11} (-28 to +65)
Stability, $\sigma_y(\tau)$ (τ = 1s)	1 x 10^{-9}	3 x 10^{-10}	1 x 10^{-12}	3 x 10^{-12}	5 x 10^{-12}	5 x 10^{-11}
Size (cm^3)	10	30	20-200	200-800	1,000	6,000
Warmup Time (min)	0.03 (to 1 x 10^{-6})	0.03 (to 2 x 10^{-8})	4 (to 1 x 10^{-8})	3 (to 5 x 10^{-10})	3 (to 5 x 10^{-10})	20 (to 2 x 10^{-11})
Power (W) (at lowest temp.)	0.04	0.04	0.6	20	0.65	30
Price (~$)	10 - 100	<1,000	200-2,000	2,000-8,000	<10,000	50,000

* Including environmental effects (note that the temperature ranges for Rb and Cs are narrower than for quartz).

TCXO:	Temperature Compensated Crystal Oscillator
MCXO:	Microcomputer Compensated Crystal Oscillator
OCXO:	Oven Controlled Crystal Oscillator

Table 2 – Oscillator comparison

This table particularly leads to several remarks related to implementation and deployment trends:

- Filtering algorithms supported by low cost oscillators usually work with observation times in the order of (some) 100 s. It particularly means that addressing TDD requirement with a 50 ppb frequency accuracy is not sufficient for achieving a 1µs time accuracy. Indeed, after 100s the maximum phase error related to the oscillator drift reaches 5µs! Thus, the 50 ppb value has specifically to be re-considered in this case for (short) holdover events. In normal operations (locked mode), frequency accuracy at the slave level should be far below this value, typically around and even below 1ppb.

- Regarding intrinsic features (frequency stability), a low-cost TCXO appears as a relevant candidate for a full TC support. On the contrary a more expensive and stable OCXO would be at least required for an end-to-end approach while considering the packet filtering stage has to cope with any kind of PDV scenario, comprising beating effect (periodic variation of the floor delay) occurring within a significant time scale (e. g. in the 10mn period range). A quick comparison indicates a ~20 cost ratio between a TCXO and an OCXO. This especially means that from an oscillator cost perspective, a full supported Master-Slave path with 20 TCXO based transparent clocks has the same cost as an end-to-end OCXO (OCXO at the slave level) approach, without any guarantee of results targeting a 1 µs time accuracy at the slave level. Indeed, packet filtering algorithm performance strongly depends on the traffic conditions that could not be deterministic during a long time.

In case of a green field scenario where infrastructure deployment costs are neutral (this has to be part of the investment in any way), these remarks advocate for a clear preference on a full BC or TC support for distribution of a time reference. Within a migration scenario, the unpredictable nature of the traffic condition and the related behavior of the Slave can also lead to the same choice.

Accordingly, the following section focuses on deployment considerations implied by such a choice.

Figure 14 typically depicts what could be a PSN demonstrating a full IEEE1588V2 support along synchronization paths. The main issue raised here is how to efficiently deploy BC and TC resources thanks to operator requirements and considered network scenarios.

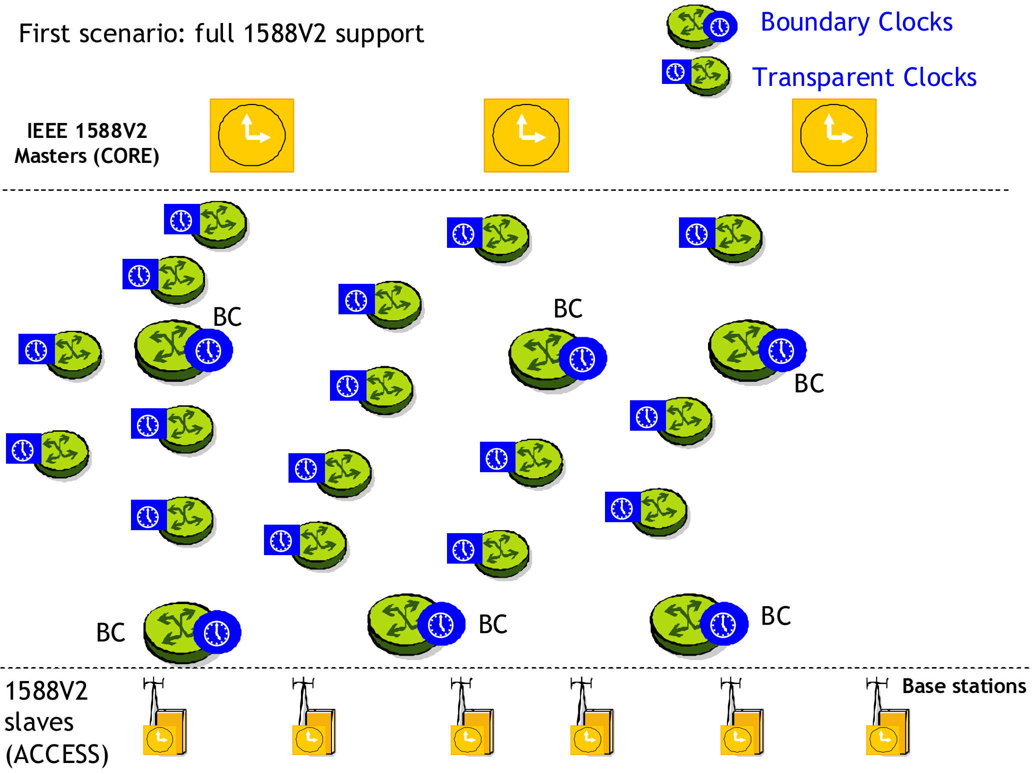


Figure 14 – Full deployed TC and BC support

In order to address such an issue, TC and BC (implementations) have to be compared. Both supports are quite – but not totally - different in principle; BCs require a locking system whereas TCs require delay measurement capabilities.

Pros and cons of each support are provided in Table 3 hereafter.

Boundary Clock	**Transparent Clock**
Pros	**Pros**
Analogy with robust SyncE	High cascadability
Technology translation, gateway role	A low-cost clock (ppm) is sufficient
Limit the synchronization message flooding	Very simple alternate BMCA
	Transparency to route changes for P2PTC deployment
Relax Master load	Relax Master load for P2PTC
	Very simple management
Cons	**Cons**
Limited cascadability: PLL noise, frequency drift, etc	E2E TC more sensitive to route changes, depending on the communication path distance and the message rate
Cost issue regarding deployment costs	Message flooding for large scale network when multicast is used
Complex BMCA	Protection costs related to transparency features: local TC redundancy, fast TC failure detection, etc
Unicast flows are transparent to BCs. This implies an update of all unicast tables at neighboring clocks at new BC introduction (resp. BC suppression)	Limiting the use of various lower-layers telecom transport layer (encapsulating the PTP packets). Mandates the use of Deep Packet Inspection (DPI) in some cases.

Table 3 – Pros and cons of TCs versus BCs

Regarding all capabilities and issues of each support, a single conclusion is arising: BC and TC are complementary hardware support. Indeed, BCs demonstrate some capabilities not addressed by TCs such as interoperability between domains relying on different technologies (for instance technology translation between 1588V2/UDP/IP and 1588V2/Ethernet), whereas TCs present high scalability potentials and a ease deployment (not part of the BMCA) at the expense of implementation complexity depending on the considered network scenarios (over Ethernet or over IP)

Focusing on Ethernet-based networks, the TC implementation issue is relaxed. This opens the door for high level rules taking the best of each approach in a standard way.

A valid scheme would be for instance to use BCs only when they are really necessary and P2P TC between, in order to

- meet interworking/interoperability requirements
- relax the grandmaster load
- save bandwidth
- relax BC clock filtering requirements within the chain (transients reduction)
- simplify as much as possible Protection schemes (limit the BMCA complexity)

7.2 IEEE 1588V2 with SyncE support

7.2.1 Transparent Clock with SyncE support

The aim of Transparent Clocks is to fight out the Packet Delay Variation by measuring 1588V2 packet residence time of traversed network elements and to advertise this (cumulated) correction to the Slave. In order to clarify the benefits of a syntonization support, some basic calculations are given bellow.

Considering a worst residence time of about 10 ms, usually revealing traffic congestion conditions, then the measurement error of this residence time by a local low-cost oscillator **demonstrating a 1-10ppm frequency accuracy, is about 100ns** (10ppm x 10ms). Without any statistical consideration (error averaging), this particularly means that a 1 µs time accuracy at the slave level could be theoretically achieved while considering both low cost clocks and a high number of (over-loaded) cascaded nodes (between 10-100 nodes in this case). Thus, at the first order, syntonization requirements for TCs are inherently dependent to scalability requirements.

Remark: this theoretical result has to be balanced regarding hardware implementation constrains that can significantly degrade the computed capabilities. As an illustration, Figure 15 depicts the theoretical time accuracy limit for a given number of cascaded nodes while considering different frequency accuracy values.

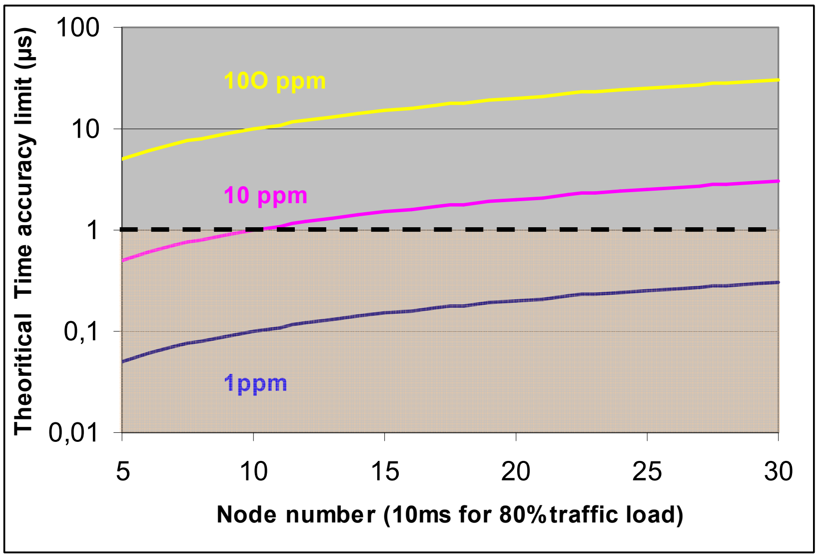


Figure 15 – Achievable time accuracy for a linear chain of TCs with different oscillator frequency accuracy in free-run

Considering the use of low-cost oscillators (10ppm) and some implementation margins, then the typical one micro-second accuracy requirement becomes quite challenging after 10 cascaded nodes.

For a high number of cascaded nodes, some deployment considerations have to be made. Indeed, from a synchronization management perspective, it could be more beneficial to deploy only IEEE1588V2 with accurate – and cost acceptable - TCs and then to derive frequency than to deploy low-cost TCs with a Synchronous Ethernet support that will require to be managed. In other terms, both CAPEX and OPEX have to be considered and not only CAPEX.

Remark: for most operators, time accuracy target is about 1 micro-second for a wide network with a scalability requirement between 10 and 30 nodes for the most stringent ones (e. g. China Mobile), under a traffic load of 80%. For an equipment supplier such as Alcatel-Lucent which typically targets several markets with a same (family) product, this particularly means that SyncE and TC features are implemented together within a Network Element.

7.2.2 Boundary Clocks with SyncE support

As demonstrated in [18], while considering a IEEE1588V2 –based time distribution principal, SyncE allows for providing a welcome clock stability which strongly enhances the performance of packet filtering algorithms (regarding the longer convergence time as discussed previously). This really makes a difference while considering sub-micro-second requirements at the slave level.

As already discussed in section 6.1.2, one of the BC issues is related to cascadability limitations. Without any syntonization support, 10 nodes cascadability is already hard to obtain thanks to 2 reported root causes [19], [20]. The first cause is related to the fact that PTP messages exchanges between different pairs of successive nodes within a synchronization path are not synchronized. The second one is explained regarding the noise generation and noise transfer of different Phase-Locked Loops (PLL).

One way for efficiently solving this scalability issue is to take advantage of a SyncE support which provides a continuous flow of physical transition information that allows for efficiently fighting out aforementioned root causes

Thanks to this SyncE support, excellent scalability results have been reported [21]. Indeed, cascading 30 nodes while still maintaining a time accuracy at the slave level below 300 ns has been reported as achievable.

7.3 Protocol inter-working proposals for the distribution of Time

7.3.1 Deployment context

Focusing on the distribution of time and according to previous discussions 2 areas of protocol interaction clearly appears: the first one is related to IEEE 1588V2 with a SyncE support. The second one is related to IEEE1588V2 and IETF NTP. Both aspects will be covered in the following sections through the proposal of interworking solutions that leverage the overall synchronization performance in terms of time accuracy, protection and optimal use of all synchronization resources.

7.3.2 Proposal

The following scenario (Figure 16) describes two inter-twined domains, a SyncE domain and a PTPV2 one, for which a SyncE syntonization support is provided to some particular IEEE 1588V2 BCs

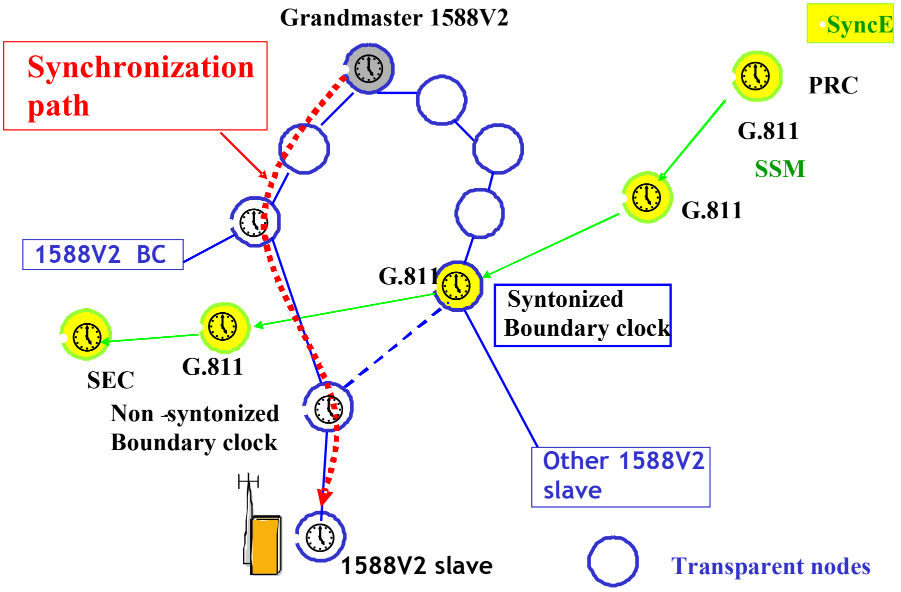


Figure 16 – Time distribution scheme ignoring any syntonization support

Referring to the default Best Master Clock algorithm behavior as depicted in the IEEE 1588V2 standard, if a local SyncE support is not previously (manually) provisioned within the PTPV2 management, then this latter simply ignores this functionality for a static configuration of the synchronization topology. The SyncE support discovery is thus a first requirement for avoiding some manual operations. This first issue could be solved by considering a collaboration scheme between SyncE and PTPV2 management entities. Ideally, the SyncE management (as a functional block) announces the SyncE capability to the PTPV2 management which then updates the PTPV2 node dataset driving the Best Master Clock algorithm (the PTPV2 topology could be statically configured as well without any BMC considerations).

The following solution, relying on an efficient protocol signaling interaction – that could be called as a “synchronization control plane” approach - proposes a simpler way for managing the synchronization network by relaxing the management plane functionalities.

This solution is based on a local translation feature as depicted hereafter. The SSM QL (Quality Level) information is locally integrated within the IEEE1588V2 Announce message in such a way that the BMC can use this available information in order to automatically build a relevant PTPV2 topology. The relevant time distribution topology is thus computed by the Best Master Clock algorithm by taking into account the existing SyncE synchronization support without any action from the synchronization management. This mode of operation is depicted by Figure 17.

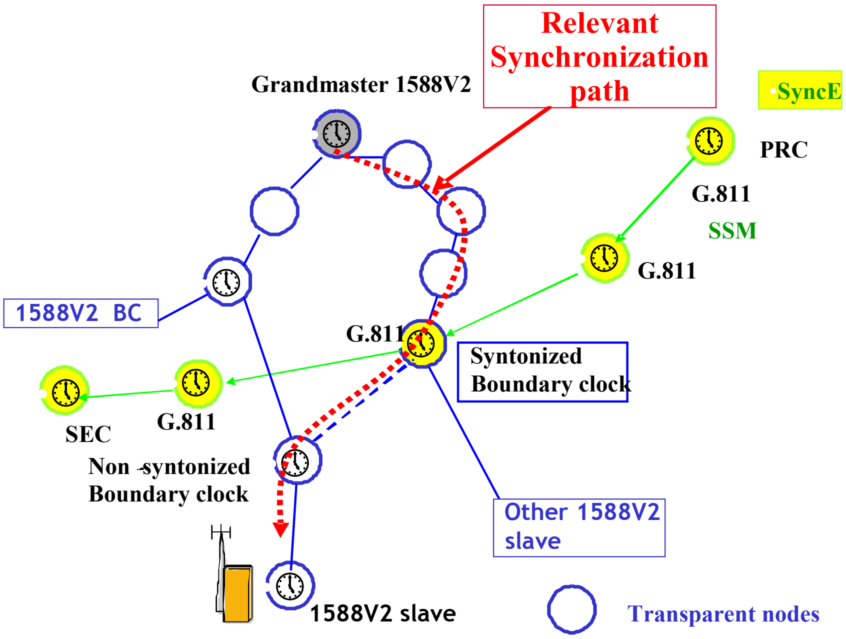


Figure 17 – Automatic computation of the PTPV2 topology thanks to an improved BMC taking into account a local SyncE support

This new methodology demonstrates all its benefits while considering failure events at the SyncE level. Indeed, a traditional management-oriented methodology may impact reconfiguration times in case of failure events regarding the remote nature of management equipments. Clearly, a change of the SyncE topology may impact the time distribution topology which has to be quickly re-configured regarding holdover constraints and phase transient monitoring. The new solution, relying on a local behavior allows for addressing efficiently these issues.

Fast protection benefits of the new solution are depicted in Figure 18. A PTPV2 path is established by the BMC which takes the best path amongst 2 paths demonstrating a local SyncE support. When Quality Level (QL) degradation occurs at the SyncE level, the BMC automatically and quickly reconfigures the topology by taking into account this QL degradation. The new PTPV2 synchronization topology is advertised to PTPV2 manager afterwards.

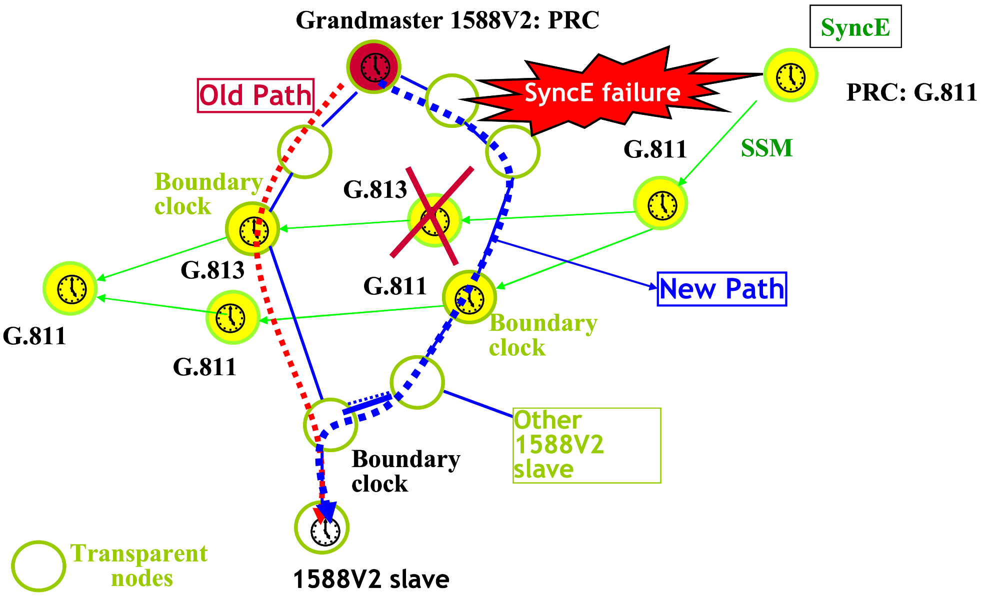


Figure 18 – Fast reconfiguration scheme facing a failure event at the SyncE level

Thus, the BMC takes into account the Quality Level degradation and automatically re-configures the time synchronization topology accordingly to the new On-Path-Support availability.

7.4 Leveraging deployed PTPV2 network hardware support for NTP

7.4.1 Context description

PTP has been preferred by the majority of telecom actors with regards to NTP as per the different concepts introduced within the release 2, especially the Hardware Timestamping (HTS) and the Transparent Clock (TC) capabilities. Moreover, the ITU-T is adapting PTPV2 into telecom environments through PTP telecom profiles which will enable higher interoperability between different vendor PTPV2 products and will ease out a broad deployment of PTPV2 within telecom networks accordingly.

Meanwhile, thanks to its long experience within the domain of time distribution (since 1985 with the first release – RFC 958), millions of existing NTP equipments, especially within the end-user and corporate environments have been deployed. This particularly means that PTPV2 and NTP have at least to co-exist.

In term of time synchronization networks, one possible scenario can be illustrated by Figure 19 below.

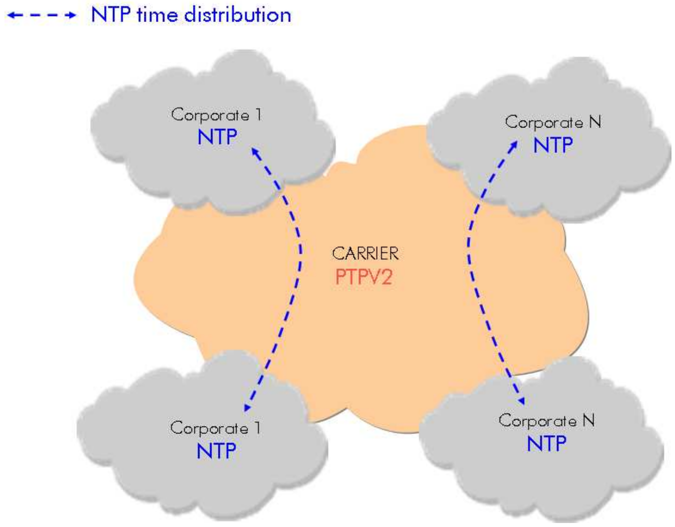


Figure 19 – PTPV2 and NTP deployment scenario

Parallel to the adaptation of PTPV2 into the telecom environment by the ITU-T, the IETF NTPV4 working group is being finalizing a “Carrier-Class NTP”, addressing the hardware timestamping. This latter work allows for improving NTP performance with regards to PTPV2, opening the door for certain telecom actors to leverage their existing NTP equipments for precise time distribution. However, the concept of TC can hardly be implemented on NTP without an equivalent to the PTPV2 correction field in the NTP packet header.

The resulting performance gap in terms of achievable time accuracy between the two protocols is a barrier for an optimal sharing of synchronization resources that would yield to cost savings. Indeed, having the same level of performance between NTP(V4) and PTPV2 will allow for using the deployed NTP elements for stringent applications as well, yielding to a reduction of the overall OPEX/CAPEX related to synchronization [22].

7.4.2 Proposals

The first proposal consists in taking advantage of PTPV2 transparent clocks for supporting the time distribution between two NTP nodes. This segmented approach does not imply any inter-working between these two protocols. It consists in connecting the time domains (NTP/1588V2/NTP) with relevant synchronization gateway embedding a dual protocol stack: a first gateway with the dual functionality NTP client + PTPV2 master and another gateway with the dual functionality PTPV2 slave + NTP server. Figure 20 allows for depicting such a proposal.

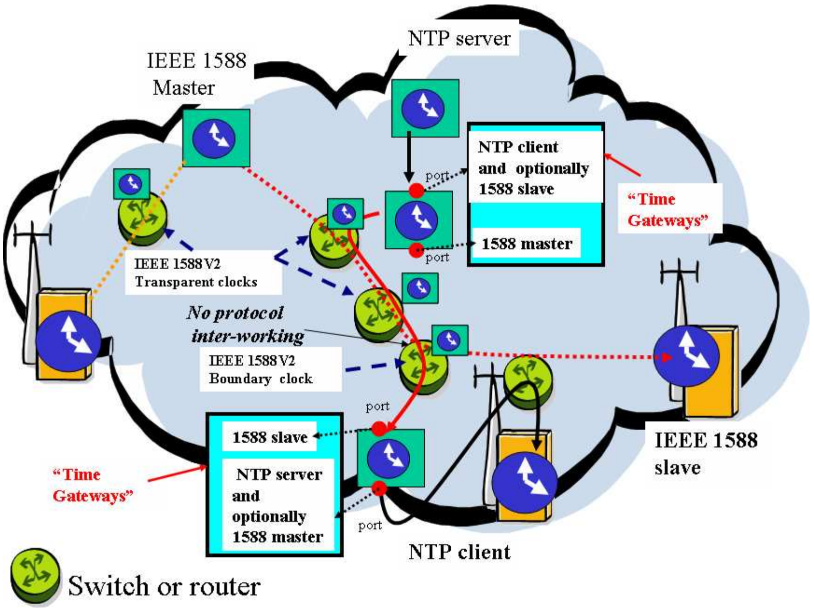


Figure 20 – NTP/1588V2 gateway concept

This solution is not good enough as time distribution performance is dependent on the deployment of these aforementioned gateways. In addition, as these latter have to ensure the connection between NTP and 1588V2 domains, their internal clocks cannot be cheap as requiring a high stability. Furthermore, this manual configuration at the gateway level is not easily applicable in a Service Level Agreement (SLA) context (e.g. inter-domain, inter-carrier). But above all, the major drawback of this method is its limitation for balancing the 'synchronization traffic load' between NTP servers and 1588V2 time reference sources. This limitation impacts synchronization deployment costs and protection schemes.

The second proposal consists in using the 1588V2 signaling for carrying the NTP one. This latter could be performed by taking advantage of the flexible 1588V2 structure through a Type Length Value (TLV) field and by ensuring a protocol inter-working at the intersection of time domains, as depicted by Figure 21.

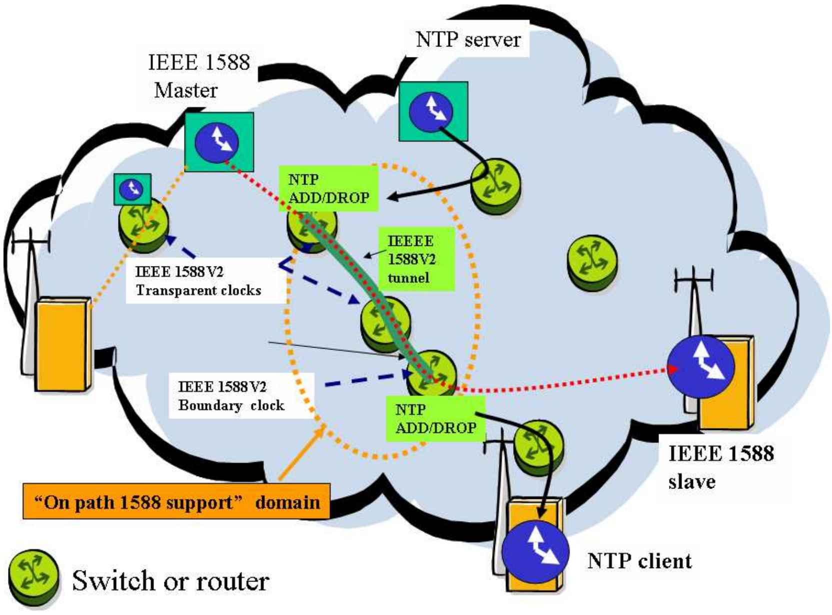


Figure 21 – NTP ADD/DROP concept

The proposed solution is based on 2 complementary building blocks:
- The first one relies on NTP ADD/DROP elements bordering 1588V2 transparent clocks domains guaranteeing the coherence and continuity between NTP and 1588V2 (particularly for timestamps and correction fields).
- The second building block is an auto-discovery/advertising mechanism dedicated to inform a management or a control plane of the position of these ADD/DROP elements to ease the (dynamic) establishment of the best paths between a NTP server and a NTP client.

NTP ADD/DROP element clocks have local requirements for the correction of (queuing) times and thus should be far cheaper than 'gateway clocks' (which stability has to be ensured during the two-way signaling). This enables a dense deployment of these formers, yielding to a high flexibility in terms of deployment (especially for sharing expensive NTP servers and 1588V2 masters), providing an optimal allocation of all synchronization resources.

7.5 Link-to-link TOD distribution using the ESMC messaging channel

7.5.1 Context description

As more and more traffic is being delivered in Ethernet format, carriers are realizing the advantages to converging on a pure Ethernet infrastructure, but are not able to do so with Ethernet as presently defined. New ITU-T Recommendations, such as Y.1731 for Ethernet OAM and G.8031 for Ethernet protection switching, provide features essential for operational maintenance of Ethernet networks. Recently proposed mechanisms for configuring point-to-point connections in Ethernet networks close another gap between Ethernet and SONET/SDH networks.

The final difference between conventional TDM-based networks and Ethernet is that the former also transport frequency information, needed for some applications, while Ethernet does not. A comprehensive solution for that problem has been achieved when the ITU-T has developed Synchronous Ethernet that locks the timing of the Ethernet physical layer, much the same way that SONET/SDH does. This enables simple recovery of the physical layer timing, which can be used directly by locking downstream networks and applications, or indirectly by using the physical layer timing as a common reference.

Nevertheless, today end applications (e.g. 3G and WiMAX cellular base-stations) are becoming more and more dependent on the distribution of a precise TOD or phase reference in addition to accurate and stable frequency reference. This ruling is even truer when it comes to future applications such as LTE cellular base stations or within Single Frequency Networks (SFN) such as the Terrestrial Digital Video Broadcasting (DVB-T) standard.

Existing packet based time distribution protocol such as IEEE 1588v2 (PTP) and the Network Time Protocol (NTP) have been markedly improving over the recent years and have been shown to be able to deliver the required level of performance in some network scenarios. Nevertheless, being ‘upper layers’ type of time distribution protocols (in the general case) they suffer from ‘upper layer’ type of network interferences such as Packet Delay Variation (PDV) and network re-route delay changes. At the end of the day, these methods cannot be trusted to always perform as expected under changing network scenarios.

A known fact is that in order to be able to assure a certain level of performance under (almost) all network conditions the information has to be carried in the physical layer (or very close to it). This can be accomplished either by applying precise HW timestamping very close to the physical layer to support the upper layer synchronization distribution protocol (e.g. TCs and link-by-link BCs in the case of PTP) or by applying the synchronization distribution directly at the physical layer (or very close to it). An FT contribution to Q13/SG15 of the ITU-T [23] purposed exactly that: a link-by-link phase distribution scheme based on link layer OAM messages.

The apparatus purposed in this document presents some modifications and improvement to the scheme presented in [23]. Specifically, the Ethernet Synchronization Messaging Channel (ESMC), currently used to carry Ethernet Synchronization Status Messages (ESSM) across Synchronous

Ethernet capable links, is suggested as a container to carry the time information between neighboring nodes.

7.5.2 The ESMC

The Ethernet SSM is an ITU-T defined [24] Ethernet slow protocol that was defined in order to distribute Quality Level (QL) information across Synchronous Ethernet capable Ethernet links. Synchronous Ethernet SSM or ESSM distributes the same SDH/SONET 4-bits QL words. In SDH/SONET, these 4-bits are carried in a fixed location in the frame header. In Synchronous Ethernet, on the other hand, these 4-bit words are distributed using a dedicated unidirectional IEEE 802.3 Organization Specific Slow Protocol (OSSP) termed Ethernet Synchronization Messaging Channel (ESMC).

In order to comply with clock selection procedures given in ITU-T Recommendation G.781 (e.g. maximal time to generate a clock source switch over when the current selected source QL changes) two ESMC message types have been defined. *Information* 1 second 'heart beat' messages are used to continuously inform the downlink node of the current QL value. Additionally, in order to minimize short-term interferences and large phase excursions during transition to holdover or clock source switch-over, an *event* message is immediately send towards the downlink element to signal it on the changed QL. To protect against possible failure, the lack of the messages is considered to be a failure condition. The protocol behaviour is such that the SSM value is set to DNU if no SSM messages are received after a five second period.

ESMC frames bear an ITU-T organizationally unique identifier (OUI) 00-19-A7 and a slow protocol subtype 0x0A that distinguish it from other OSSPs. The SSM quality level is carried in a type length value (TLV) field, which is contained within the ESMC PDU. Two types of ESMC PDU frames are defined and are distinguished by the C flag. These are the ESMC information PDU and the ESMC event PDU mentioned above.

The ESMC PDU format is shown in Table 4. The QL TLV is shown in Table 5. The same QL TLV is used for both the information and event messages. To allow for potential hardware implementations, the SSM TLV is always sent as the first TLV in the data/padding field. This means that the QL indication always remains fixed in the PDU. Any padding must occur after the SSM TLV.

Octet number	Size/bits	Field
1-6	6 octets	Destination Address =01-80-C2-00-00-02 (hex)
7-12	6 octets	Source Address
13-14	2 octets	Slow Protocol Ethertype = 88-09 (hex)
15	1 octets	Slow Protocol Subtype =0A (hex)
16-18	3 octets	ITU-OUI = 00-19-A7 (hex)
19-20	2 octets	ITU Subtype
21	bits 7:4(note 1)	Version

	bit 3	Event flag
	bits 2:0(note 2)	Reserved
22-24	3 octets	Reserved
25-1532	36-1490 octets	Data and Padding (See point J)
Last 4	4 octets	FCS

Note 1: Bit 7 is the most significant bit of Byte 21. Bit7 to bit 4 (bits 7:4) represent the four bit version number for the ESMC.

Note 2: the three least significant bits (bits 2:0) are reserved.

Table 4 – ESMC PDU format (Table 11-3/G.8264)

ESMC PDUs have the following fields in the order specified above:

a) Destination address (DA): This is the IEEE-defined slow protocol multicast address. The format is defined in Annex 43B of [IEEE 802.3].

b) Source address (SA): The source address is the MAC address associated with the port through which the ESMC PDU is transmitted.

c) Slow protocol Ethertype: ESMC PDUs must be type encoded and carry the slow protocol type field value.

d) Slow protocol subtype: Assigned by the IEEE and fixed with a value of 0x0A.

e) ITU OUI: Organizational unique identifier assigned by the IEEE registration authority.

f) The ITU subtype is assigned by ITU-T. The value of 00-01 applies to all usage defined in this Recommendation.

g) Version: The four-bit field indicates the version of ITU-T OSSP frame format. This field shall contain the value 0x1 to claim compliance with version 1 of this protocol.

h) Event flag: This bit distinguishes the critical, time-sensitive behaviour of the ESMC event PDU from the ESMC Information PDU. A value of 1 indicates an event PDU, a value of 0 indicates an information PDU.

NOTE 1 – The behaviour of the event PDU is similar to the critical event defined for Ethernet OAM in clause 57 of [IEEE 802.3]. Event messages need to meet processing times defined in [ITU-T G.781].

i) Reserved for future standardization (27 bits). These fields are set to all zero at the transmitter and are ignored by the receiver.

j) Data and padding: This field contains data and necessary padding to achieve the minimum frame size of 64 bytes. The PDU must be an integral number of bytes (octets). Padding characters are not defined and are ignored by receivers.

NOTE 2 – The recommended maximum size for the ESMC PDU is 128 bytes as per Annex 43B of [IEEE 802.3]. However, PDU sizes greater than 128 bytes may be permitted.

k) FCS: Four-byte frame check sequence as defined in clause 4 of [IEEE 802.3].

Octet number	Size/bits	Field
1	8 bits	Type:0x01
2-3	16 bits	Length: 0x0004
4	bits 7:4	0 (unused)
	bits 3:0	SSM code
Note 1: Bit 7 of Octet 4is the most significant bit. The least significant nibble, bit 3 to bit 0 (bits 3:0) contain the four bit SSM code.		

Table 5 – QL TLV format (Table 11-4/G.8264)

According to [24], additional TLVs comprising future extensions to the ESMC messages are possible. Such extensions might include 'trace-route' like mechanisms to better mitigate timing loops. Another possible extension to the ESMC messages is the support for phase/TOD distribution over Ethernet links.

7.5.3 Proposals

The proposal comprises 3 major parts.

The first part of the proposal is to enhance the current ESMC messages format with an additional TLV that will carry TOD information across an Ethernet link.
It is therefore a fact that unidirectional ESMC messages carrying QL information are distributed across both directions of any Sync-E capable line[1]. Hence, Sync-E capable Ethernet networks already possess a very bandwidth efficient communication channel between neighbouring nodes. As already discussed in the motivation part of this document, extending this messaging channel with a complete TOD distribution scheme seems, therefore, something of great value.
In order to make it happen and allow ESMC messages to convey TOD information two additional information entities needs to be added to the current standard ESMC messages format:

(1) Timestamps (TS) that will carry the TOD information from one side of the link to the other. As will be presented shortly, two timestamp fields will be added to each ESMC message using the TLV format presented in Table 5 above.
(2) A Sequence number (SN) that will mark the ESMC messages according to their chronological transmission order. As will be presented shortly, the sequence number will be used to correlate a timestamp to the specific ESMC message it belongs to.

One possible way of incorporating these changes in the current standard ESMC message format is given in Table 6 and Table 7. The changes comprise an additional 24-bit Sequence Number introduced into currently reserved 3 octets (22 to 24) in the message header. This will be a standard running sequence number that will reset to 0 once reaching its maximal value (2^{23}-1). The two timestamps will

---

[1] According to [24] there might be Synchronous Ethernet links with partial support. In other words, where the physical layer clock as well as the QL distribution are being distributed in one direction only or where ESMC messages are not being transmitted at all. Nevertheless, these cases can be regarded as exceptional and are only relevant for last mile Ethernet links.

be introduced into a new dedicated TOD TLV that will be placed immediately after the currently standard QL TLV (starting at octet number 29). Each of these timestamps will have the standard PTP 80-bit Timestamp format [25] to allow for better interworking with PTP as will be later on explained and demonstrated (alternatively these timestamps could have been designed to bear the standard NTP format of 64-bits. Nevertheless, as PTP seems to be getting more traction from telecom service providers, especially in the cellular backhaul arena, and a lot of discussions and work had been taking place concerning better PTP/Sync-E interaction, it seems only natural to adopt the PTP Timestamp format).

The first Timestamp (octets 4-13) records the precise time (according to the local node time counter) at which the *last* periodic ESMC message was received in the specific node, and is used to feedback the other end of the link with that information. Following (octets 14-16), the sequence number of the specific received periodic message that the timestamp refers to, is also sent to allow the receiver at the other end to correlate between transmission and reception time events on this specific direction of the link.

The second Timestamp (octets 17-26) records the precise time (according to the local node time counter) at which the *last* ESMC message was transmitted by the specific node (alternatively this Timestamp could have been made to comprise the precise time at which the *current* transmitted ESMC message left the node. However, this would impose on-the-fly HW timestamping mechanism at each participating node. Such a mechanism might increase the overall cost of the scheme, but more important, will unnecessarily degrade the accuracy of the timestamps by some amount).

As will be explained in the next clause, each link end (node), independently sending and receiving ESMC messages, will have all the information it needs to recover TOD information from the other end. This concept is very much in the spirit of standard Sync-E frequency distribution where (except for some specific Ethernet interfaces) the frequency information can, potentially, go both directions and correct distribution direction is determined by the overlay clock selection procedure.

Octet number	Size/bits	Field
1-6	6 octets	Destination Address =01-80-C2-00-00-02 (hex)
7-12	6 octets	Source Address
13-14	2 octets	Slow Protocol Ethertype = 88-09 (hex)
15	1 octets	Slow Protocol Subtype =0A (hex)
16-18	3 octets	ITU-OUI = 00-19-A7 (hex)
19-20	2 octets	ITU Subtype
21	bits 7:4(note 1)	Version
	bit 3	Event flag
	bits 2:0(note 2)	Reserved
22-24	3 octets	24-bit Sequence Number (SN)
25-1532	36-1490 octets	Data and Padding (See point J)
Last 4	4 octets	FCS

Table 6 – TOD enhanced ESMC PDU format (proposed changes are highlighted)

Octet number	Size/bits	Field
1	8 bits	Type:0x02
2-3	16 bits	Length: 0x0017
4-13	80-bit	Timestamp of last received periodic ESMC message
14-16	24-bit	Sequence Number of last received periodic ESMC message
17-26	80-bit	Timestamp of last transmitted periodic ESMC message

Table 7 – TOD TLV format (proposed changes are highlighted)

It should be emphasized that the above TOD extensions need only be applied to *periodic* (Event flag = '0') ESMC messages with the average rate of 1 PPS. *Event* driven ESMC messages (Event flag = '1') need not incorporate these modifications as they will not contribute anything to the TOD distribution scheme. Hence, as far as TOD distribution is concerned, ESMC messages with Event flag = '1' should be ignored.

The second part of the proposal is to define a new apparatus exploiting the new proposed ESMC messages TOD TLV extension to distributed TOD information across the Ethernet link.

Lets us begin by assuming two neighboring Sync-E capable nodes **X** and **Y** that are, in the general case, part of a Sync-E frequency distribution chain according to [7], [26] and [24]. Each of these nodes will transmit standard QL ESMC messages to the other Naturally, depending on the decided timing information distribution direction, one node will transmit QL='DNU' to the other side. As will be discussed in the next clause, this QL information could also come very handy for the TOD distribution process, but for now we will assume that the frequency distribution and the TOD distribution process are completely orthogonal.

By extending the ESMC messages format as explained in the previous clause, each node will now be transmitting the following TOD information to its neighbor:

(1) Precise local time of last received periodic ESMC message marked by: $TS_R(k\text{-}1)$.
(2) Precise local time of last transmitted periodic ESMC message marked by: $TS_T(n\text{-}1)$,

where k and n are the different periodic ESMC messages indexes on both directions.

Additionally, at each node, the timestamp of the precise local time at which the last periodic ESMC message was transmitted ($TS_T(k\text{-}1)$) as well as the timestamp of the precise local time at which the last periodic ESMC frame was received ($TS_R(n\text{-}1)$) are recorded and saved. A graphical representation of the above text is given in Figure 22.

At each node, every 1 second after the completion of a successful ESMC message reception, the exactly same following four timestamps will be available: $TS_T(k\text{-}1)$, $TS_R(k\text{-}1)$, $TS_T(n\text{-}1)$ and $TS_R(n\text{-}1)$. Using these four timestamps each node could now calculate an estimation of the time error between himself and its neighbor using the following, very known, time error estimation formula:

$$\widetilde{timeerror}(k-1) = \frac{1}{2}[(TS_R(k-1) - TS_T(k-1)) - (TS_R(n-1) - TS_T(n-1))]$$

The calculation time error estimation could, at this point, be used to time-lock each node to the other. The only left issue to be resolved is the correct direction (based on some correctness criteria as will be discussed in the clause) of the TOD information flow. In other words, which node is going to play the role of the *Master* and which one will be *Slave,* for this specific link (not setting an unambiguous master-slave relationship for the specific link might results in a “time-loop” situation where, in a very similar manner to timing-loops in frequency distribution, both ends will simultaneously try to maintain lock to the opposite, resulting in unstable, unpredicted behavior).

Clearly, the above suggested apparatus introduces a 1-second delay (at least) in the time error estimation procedure, and therefore, also in the TOD recovery process. At first glance, this fact might be seen as problematic (1 second in the time error estimation procedure means that the time error could be reduced up to that residual time error, stemming from the time error development over that 1-second calculation delay). Nevertheless, taking into account that the aforementioned residual time error is negligibly small if the time counters on both nodes are timed by the Sync-E physical layer recovered clock, this does not appear to be such a bug sacrifice at the benefit of avoiding the timestamping accuracy degradation that accompanies any on-the-fly timestamping mechanism.

Many time (clock) recovery apparatuses can be envisaged to perform the time lock procedure based on the calculated time error estimations. This issue is currently left for further study.

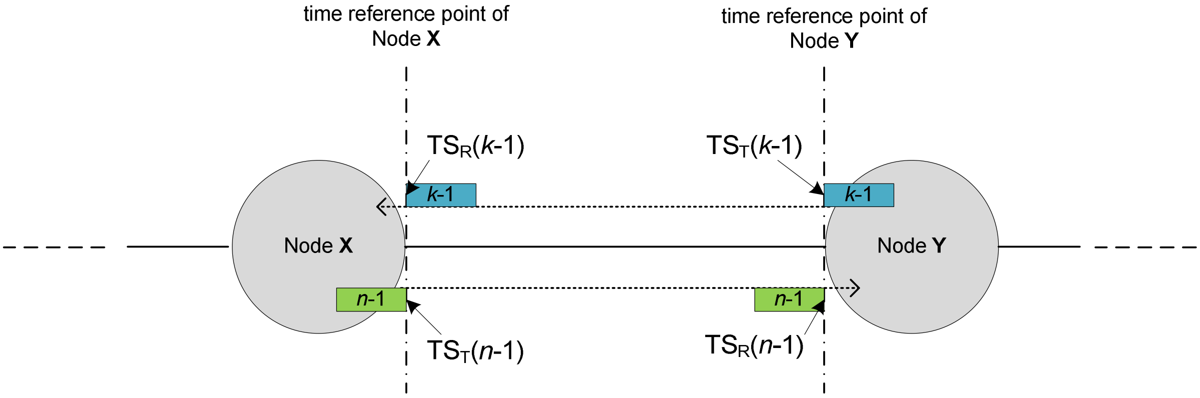


Figure 22 – TOD ESMC messages exchange

The third part of the proposal is to define the TOD selection process that will be used in each node to select the best TOD information source among possible multiple sources. This TOD selection process is analogues to the frequency selection process defined in [27].

Simplifying the matter a little bit, TOD distribution can be broke-down into the following elements:

(1) A frequency distribution scheme that maintains the local time counters along the distribution node in the correct pace.

(2) An additional occasional phase correction information that align the phase of the clocks along the distribution chain, removing any developed bias.

In that view, the performance of a link-by-link TOD distribution scheme is very much impacted, in the general case, from the same factors that impact the performance of a link-by-link frequency distribution scheme. In other words, similar to the frequency distribution case, the specific free-run quality of the local clocks along the distribution chain plays an important role. On the other hand, as this is a link-by-link TOD distribution scheme, other attributes such as path asymmetry and path PDV are of less importance. In other words, TOD distribution chains that comprise stable and accurate local oscillators have better chance of delivering good TOD distribution performance.

Another reinforcement of that approach is the fact that the other capabilities of the conventional frequency selection process such as the hold-off and wait-to-restore functions as well as the various user configurable parameters (priority tables, force/manual switch commands etc.) seems to be also desired functionalities for the TOD distribution case.

It therefore seems logical that the same standard clock selection process already established for the act of frequency distribution could be largely reused for TOD distribution.

Two strategies of incorporating the fundamentals of frequency clock selection into TOD clock selection can be envisaged:

(1) Using the exact same clock selection process for both frequency and TOD. This basically means staying with the existing clock selection process, using it to select both frequency and TOD. Working in such a strategy ensures that the same reference clock will always be selected for both frequency and TOD.

(2) Using a similar but orthogonal clock selection process for TOD. The existing clock selection process will be modified to better suit TOD source selection (for example, the specific values of the hold-off and wait-to-restore timers might change or a different user priority table might be used). The new TOD selection process will run in parallel to the frequency selection process, which might result in different reference clocks selected for frequency and TOD. This obviously means different network paths for frequency and TOD.

The first strategy is very simple and straightforward. No change (almost) to existing clock management agents in live products will be needed. The frequency and TOD distribution will always go together hand by hand following the same network path, a fact, that will simply overall network management. Nevertheless, such a scheme offers little flexibility for the user to force its own TOD distribution preferences, which might be different from his frequency distribution preferences, into the selection process.

The second strategy, on the other hand, gives the user the freedom to design and configure its TOD selection preferences independently from his frequency distribution preferences. Nevertheless, it doubles the required management efforts.

7.5.4 Deliverables D24

A MATLAB simulation environment will be produced in order to better evaluate the suitability and achievable performance of the outlined purposed apparatus through this document. The simulation environment will be comprised of a TOD/frequency Sync-E distribution chain with a configurable number of nodes. The following features will be tested:

(1) End-to-end TOD distribution performance for various number of intermediate nodes (with physical layer clock distribution support).
(2) Distribution chain reorganization performance due to fault conditions.
(3) Suitability of the current frequency clock selection scheme for TOD applications
(4) Suitability of an independent TOD clock distribution scheme.

Other features may also be tested based on internal TIGER II discussions. The specific details are for further study.

8 Conclusion

After discussing on the main synchronization protocol constraints and dependencies, this document provides innovative ideas and concepts allowing for the improvement of both frequency and time distribution over Packet Switched Networks (PSNs). Many of these proposals advocate for a strong interworking between synchronization protocols for optimizing synchronization resources deployed across the network regarding targeted requirements. Accordingly, some proposals point out the benefits of a converged synchronization management system [28] efficiently supporting such interworking schemes.

From this protocol inter-working perspective, link-by-link time distribution methodologies, relying on a physical-layer frequency distribution topology - SyncE - , have been presented and discussed. In any case it appears that a trade-off has still to be found between deployment cost - CAPEX, OPEX - and achievable performance, according to each operator requirements regarding targeted network scenarios.

Considering a full controlled network, QoS approach has been mentioned as a powerful methodology for improving the synchronization performance. This aspect will be particularly investigated within the TIGER II project.

By the way, considering drastic time distribution requirements, that could be hardened regarding recent wireless orientation (e.g. MIMO, MBMS), SynchE appears as a welcome and precious support for timestamp protocols regarding scalability and holdover issues.

Within this scope and apart from standardized approaches, a new SynchE-based time distribution solution is proposed. This latter, relatively simple in principles, particularly allows for simplifying the synchronization management, while raising new issues in terms of interoperability and ability to address all network scenarios.

On the whole, regarding the introduction of innovative ideas and concepts, further investigations will be required based on France Telecom "preferred approaches". In this context, some platform test and simulation results will be presented as part of the final D24 document.

Within the latter context, it is to be noted that packet clock models have not yet been defined by the ITU-T and thus related performance metrics have not yet been specified. It is therefore not an easy task to discuss on performance gain introduced by different aforementioned proposals as any number is likely to be implementation specific. However, some specific use-cases will be presented in order to narrow down the deployment environment and to allow for making some performance gain assessment.

9 References

[1] ITU-T G.8261, Appendix IV.2.3.

[2] K. O'Donoghue - Synchronization in Packet networks – NTPv4 and IEEE 1588 - International Telecom Sync Forum (ITSF) 2007, November 13th 2007.

[3] IETF RFC5481 - Packet Delay Variation Applicability Statement – March 2009

[4] ITU-T SG15/Q13 WD35 - Effect of routers' architecture on timing distribution - San Jose, March 2009

[5] D.T. Bui, A. Dupas, M. Le Pallec – Packet Delay Variation Management. International IEEE Symposium on Precision Clock Synchronization for Measurement, Control and Communication (ISPCS), Brescia 2009.

[6] K. Shenoi, Symmetricom - Sync for Support of Legacy Services – ITSF London, October 2005.

[7] ITU-T G.8261/Y.1361 - Timing and synchronization aspects in packet networks – April 2008.

[8] IEEE 1588 Boundary Clock and Transparent Clock Implementation using the DP83640 AN-1838 – National Instrument 2008.

[9] ITU-T SG15/Q13 WD38 - Additional PDV measurements on Ethernet platform, Edinburgh, June 2009.

[10] J. Aweya, D.Y. Montuno, M. Ouellette, K. Felske – Clock recovery based on packet inter-arrival time averaging. Computer Communications 29. 2006.

[11] J. Aweya, D.Y. Montuno, M. Ouellette, K. Felske – Analysis of a clock-recovery technique for circuit emulation services over packet networks. International Journal of Communication Systems. 2008.

[12] S.B. Moon, P. Skelly, D. Towsley – Estimation and Removal of Clock Skew from Network Delay Measurements. Proc. IEEE Infocom, March 1999.

[13] L. Zhang, Z. Liu, C.H. Xia – Clock Synchronization Algorithms for Network Measurements, Proc IEEE Infocom, June 2002.

[14] ITU-T SG15/Q13 C363 - Definition of Minimum TDEV (minTDEV) – Geneva, June 2007 .

[15] ITU-T SG15/Q13 WD36 - Packet selection methods for performance metrics- San Jose, March 2009.

[16] ITU-T SG15/Q13 C327 - Modelling and simulation of long-tailed packet delay variation (PDV) over a multi-switch network, May 2007.

[17] John R. Vig - Crystal Resonators and Oscillators for Frequency Control and Timing Applications, US Army Communications-Electronics Research, Development & Engineering Center Fort Monmouth, NJ, USA , January 2007.

[18] Yaakov (J) Stein, Alon Geva, Gaby Zigelboim- Delivering Better Time-of-Day Using Synchronous Ethernet and 1588, ITSF 2007.

[19] Sihai Wang, Jaehun Cho, and Geoffrey M. Garner - Improvements to Boundary Clock Based Time Synchronization through Cascaded Switches – Samsung Electronics.

[20] IEEE 1588 Boundary Clock and Transparent Clock Implementation using the DP83640 AN-1838 – National Instrument 2008.

[21] ITU-T SG15Q13, C599, Huawei,-ZTE ITU-T SG15Q13, WD29, - Test Results of IEEE1588v2 hop-by-hop time synchronization using Boundary Clocks. , December 2009.

[22] M. Le Pallec, D.T. Bui - Synchronization Protocol inter-working - ITSF 2009 Roma, November 2009.

[23] Contribution C449 to SG15 by France Telecom, 'Proposal of transmission of a phase information using OAM messages', Geneva, May 2007.
[24] ITU-T G.8264 – Distribution of timing packets through packet networks – October 2008.
[25] IEEE Std 1588-2008, IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems, July 2008.
[26] ITU-T G.8262 - Timing characteristics of synchronous Ethernet equipment slave clock (EEC) – August 2007.
[27] ITU-T G.781 – Synchronous Layer Functions – September 2008.
[28] M. Le Pallec, D.T. Bui, F. Dorgeuille and N. Le Sauze – Time and Frequency Distribution Over Packet Switched Networks. Bell Labs Technical Journal 14(2), 131-154.