NTNU
Norwegian University of
Science and Technology

An Innovative Synchronization Technique for OpMiGua-based Mobile Backhauls

The IEEE 1588v2 HPTS Scheme

Francesco Puleio

Master of Science in Communication Technology
Submission date: July 2010
Supervisor: Steinar Bjørnstad, ITEM

Norwegian University of Science and Technology
Department of Telematics

Problem Description

The Optical Migration Capable Networks with Service Guarantees (OpMiGua) concept (http://www.opmigua.com) has the main objective of combining the best properties from both circuit and packet switched networks into a hybrid solution. While the OpMiGua previously has been studied for large powerful transport networks with many wavelength channels, the main objective of this project is to apply the OpMiGua hybrid principle in mobile backhauls, that is a network containing only a few, or only a single channel.
The focus of this project will be on the synchronization among Base Stations when the migration to the all-packet solution is completed. The hybrid properties of OpMiGua can be an important added value when proposing a new synchronization scheme.

Assignment given: 27. January 2010
Supervisor: Steinar Bjørnstad, ITEM

To my precious family
Mimmo, Rosa, Toto
and welcome to Marta

Abstract

Legacy mobile backhauls are based on Time Division Multiplexing. Due to the current evolving of mobile traffic, a major change to a packet-based network is seen as inevitable. This means that the TDM signal cannot be used as source of synchronization anymore. The packet-layer based approach of the IEEE 1588v2 protocol has experienced a successful diffusion, while the physical-layer based solution of Synchronous Ethernet is still under development.
A new packet-based synchronization technique is presented along this thesis. It is based on applying the OpMiGua HPTS scheme over a mobile backhaul that is structured into a cluster-type topology. The proposed technique has been given the name of IEEE 1588v2 HPTS scheme. It consists of exchanging timestamps according to the synchronization algorithm standardized into the IEEE 1588v2 protocol but exploiting the switching capabilities of the OpMiGua HPTS node.
Differently from the IEEE 1588v2 protocol, our scheme foresees the timestamps to be preprended to the train of time-slots traveling throughout each ring (i.e. the fundamental element of the cluster-type network topology). The presence of time-slots is due to the adoption of the OpMiGua HPTS scheme. Thanks to the hybrid switching capabilities of the OpMiGua HPTS node, it is assured a fixed end-to-end delay to the timestamps.
Work done consists of proposing a structure for the OpMiGua HPTS node, allowing to forward and duplicate for processing the timestamps in the same time, and the format of all the messages foreseen into the new scheme. A header format is presented as well as the framing for the messages containing the timestamps. This is applied over the cluster-type topology since the latter is identified as a more suitable physical layer configuration compared to the tree structure of legacy backhauls.
The proposed technique is able to improve the accuracy achievable, since the timestamps are made independent from the traffic load into the nodes. Also a saving in terms of bandwidth consumption is provided.

Preface

This thesis is submitted as the final work for the degree of Master of Science in Communication Technology (CommTech) that has been taken by the author at the Norwegian University of Science and Technology (NTNU, Norges Teknisk-Naturvitenskapelige Universitet). The specialization Network and Quality of Service has been chosen as part of the double-degree exchange program TIME (Top Industrial Managers of Europe). The report is based on the work conducted by the author in the period January-July 2010 on a project assignment assigned by the Department of Telematics, under the supervision of the associated professor Steinar Bjørnstad.

I wish to thank my professor Steinar Bjørnstad for the valuable feedbacks and the constant and helpful support. I will treasure the opportunity I received by working on such an interesting project as OpMiGua is. I also take the opportunity to wish him the best luck with his newly founded company TransPacket.

A sincere and heartfelt thanks to all the people that made these two years of study the experience of my life. I would like to mention Chris, Stefano, Alessandra and Supertramp for their moral support during the writing of this thesis.

Francesco Puleio
July 2nd, 2010

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction

In the last decade mobile backhauls have undergone many major changes and more are still to go [1-3]. The growth in the number of mobile users is not the only factor driving those changes; it has to be combined with the big upgrade taken by broadband services over the air. The hunger of bandwidth from a single mobile subscriber is forecast to increase almost 500 times-fold in the present 10 years (2005 - 2010) [4] . The mobile traffic is going to experience a dramatic growth that has not been experienced before, neither from the fixed data traffic. At the base of this incredible expected growth there is the shift of the traffic over-the-air from voice-centric to data-centric [5]. The bulk of the expected traffic in 2015 will be made up of Data and Video, followed by P2P and only at the end Audio [4].
As a consequence, high transport capacity requirements are put on the network backhauling the mobile traffic. On the other hand, due to the Time Division Multiplexing nature of the existing backhauls, the revenues do not scale linearly with the increase in traffic [2]. A major switch from TDM-based to packet-based backhauls is required in order to optimize the cost efficiency. Thanks to these changes, the backhauling networks will be able to satisfy the requirements of Next Generation Mobile Networks (NGMNs) [6]. Among the transport protocols, Carrier Ethernet is being broadly accepted by service providers as the most suitable solution thanks to its cost-effectiveness [7].
In this rapidly evolving scenario of mobile backhauls, the hybrid network proposed by the OpMiGua project can bring the important added value of providing both packet-switch and circuit-switch Quality of Service to the traffic passing through its nodes [8-9]. OpMiGua is primary thought for optical networks, but it finds its application also in the electrical domain. The Hybrid Packet/Time Slotted Circuit Switched Scheme (HPTS) is particularly suitable for mobile backhauls, that are networks containing only a few, or only a single channel (optical or electrical). It is based on dividing the circuit-switched path into time-slots, allowing different circuits to share the same channel by time-slot allocation [10]. By means of the HPTS scheme, the OpMiGua concept is able to provide finer granularity. Which, together with its hybrid circuit-switching capabilities, makes it a really interesting solution in mobile backhauls where audio and video traffic has very strict boundaries upon end-to-end delay and jitter.
The possibility of providing end-to-end circuit-switched QoS to the packets might also considered an added value for a peculiar requirement of mobile backhauls: synchronization among the Base Stations (BSs). In the evolution from TDM-based to packet-switched technology into the backhauls, an important drawback is that packet-switched networks do not carry the timing signal. In PDH or SONET/SDH networks, the TDM signal is used as a source of synchronization itself [11]. Switching to an all-packet network, this source of synchronization is gone thus alternative methods have to be used [12]. A substitute that is a good candidate for providing synchronization over a packet-switch network is represented by the IEEE 1588v2 protocol (aka PTP, Precision Time Protocol) [13]. Standardized in 2002 and with the release of a second version in 2008 that addresses the telecommunication profile, it organizes the clocks in a Master-Slave hierarchy and states the

exchange of timestamps between each master and slave in order to achieve a target accuracy of sub-microseconds.
We are going to present a new synchronization technique which exploits in the same time the switching capabilities of an OpMiGua node with the principles stated in the PTP. This technique is going to be introduced over a cluster-type topology, a specific physical configuration that brings improvements in relation with reliability, link cost parameter and handling of diversity handover. Adopting the OpMiGua HPTS scheme into a cluster-type network topology means having a train of time-slots travelling throughout each ring, the fundamental element of such a layer configuration. This specific feature is the cornerstone of our proposal.

1.2 Motivations

The main motivation behind this work is a deep interest for the innovative OpMiGua project, seen by the writer as a final topic that fits perfectly with his field of study. Part of a double degree project which allows to achieve two different master degrees in two different European universities, the author based his studies on optical networking. He combines studies upon optical techniques at the Politecnico di Milano while deepening on the networking and Quality of Service provisioning at NTNU. The thesis applies the OpMiGua project over a branch of telecommunications of relevant importance at the present time such as the mobile backhauls.

1.3 Research method

The research method consisted of two parts:

- narrowing down in order to identify a specific application of OpMiGua into the vast topic of mobile backhauls
- introduce the technique clearly and make it easily understandable

In the first part the reader is introduced into the challenging problem of synchronization among Base Stations in mobile backhauls. Starting from the big picture of the major changes that the backhauls are undergoing to in this decade, the main elements needed for our proposal are one by one presented:

- why the switch to all-packet network is inevitable, making the synchronization a challenging aspect of the backhauls
- which are the fundamentals of the OpMiGua project and its particular application that is the HPTS scheme, why OpMiGua can provide an added value to mobile backhauls
- which are the solutions already existing, which one can be sensibly improved with the introduction of the OpMiGua concept.
- since installation of OpMiGua has to be considered a long-term investment, the spotlight is put on a specific network configuration which brings important enhancements for future mobile networks

In the second part our proposal is illustrated. Also in this part an approach step by step has been chosen. First a general description of the technique is provided, thus we enter into the details. In this part where our *innovative* work is shown, the knowledge previously provided is of fundamental importance. After a part of the proposal is illustrated, it always follows an example. The latter presents the proposal step by step in order to make its comprehension clearer to the reader.

1.4 Report Outline

The thesis is organized according to the following structure:

Chapter
↳ Section
    ↳ Subsection

The final repost is structured in Chapters, which in turn are divided in Sections, which in turn can present Subsections.

Chapter 2

Overview of mobile backhauls

The rise in data traffic carried upon wireless networks over the last years has put a lot of attention on the mobile backhaul networks. If on one side mobile operators witnessed significant increase in connection speeds over the air (both on the uplink and downlink) brought by newer cellular technologies and a lot of improvements over the wired core network as well, on the other side still many changes have to happen about the mobile backhaul networks. There is not a unified definition for Mobile Backhaul, but it is universally recognized as the portion of the network that connects the Base Stations (BSs) and their air interface to the Base Station Controllers (BSCs), which in turn are connected to the mobile core network. The cell site may either consist of a single BS or a collection of BSs that are aggregated. The generic overview is illustrated in Figure 2-1.

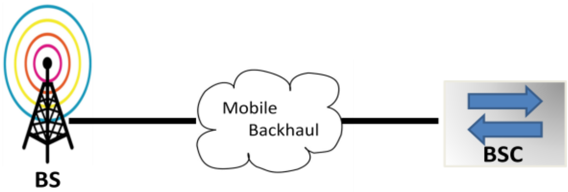


Figure 2-1. Mobile backhaul as the portion of the network that connects the Base Station (BS) and its air interfaces to the Base Station Controller (BSC)

In this chapter an overview over mobile backhauls is provided. In Section 2.1 the enormous growth in traffic over the air is presented, due to a switch of traffic from voice-centric to data-centric. This calls for major changes since with the T1/E1 lines of legacy backhauls the revenues do not scale linearly with the increases in traffic. In Section 2.2 is presented how mobile backhauls evolved so far in order to answer to the new hunger for bandwidth, with the final goal to switch to a packet-only architecture. In Section 2.3 an overview over Carrier Ethernet is provided, the extension to Ethernet which is being successfully adopted by service providers. In Section 2.4 an hint of the requirements of NGMN (Next Generation Mobile Network) backhauls is provided.

2.1 From voice-centric to data-centric

What is leading the changes in the mobile backhauls is the shift of the services required over the air from voice-centric to data-centric; the increased availability of mobile data devices such as 3G handsets, laptop cards, the iPhone, the BlackBerry and other PDAs (Personal Digital Assistants) are driving huge increases in IP traffic out at the cell tower. As illustrated by the latest traffic mobile data traffic forecast from Cisco [4], the amount of traffic exchange of a single mobile subscriber in 2015 could very conceivably be 450 times what it was 10 years earlier in 2005 (Figure 2-2).

Figure 2-2. A single mobile subscriber is forecast to increase its bandwidth consumption from 30 Mbytes in 2005 to 14,275 Mbytes in 2015 (450 times more)

As shown in Figure 2-3, how these new high-end handsets and laptop cards impacts the legacy mobile network in term of required bandwidth is dramatic: a single laptop can generate as much traffic as 450 basic-feature voice-based phones, and the fully-successful product iPhone creates as much traffic as 30 basic-feature phones.

Figure 2-3. Full-successful smartphones (i.e. Iphone and similar) can generate as much traffic as 30 basic-feature voice-based phones, new high-end handsets and laptop cards as much as 450 basic-feature phones

Accordingly with the mass-diffusion of these advanced devices, also highly resource consumptive application are going to be more and more present in the mobile cells in order to be served. Always according to [4], the application responsible for the majority of the growth between 2008 and 2013 is going to be the video. If in 2013 mobile data traffic is expected to grow to more than 2 exabytes per month, over 1.4 of those are due to mobile video traffic, i.e. almost 64 % of the total traffic (Figure 2-4).

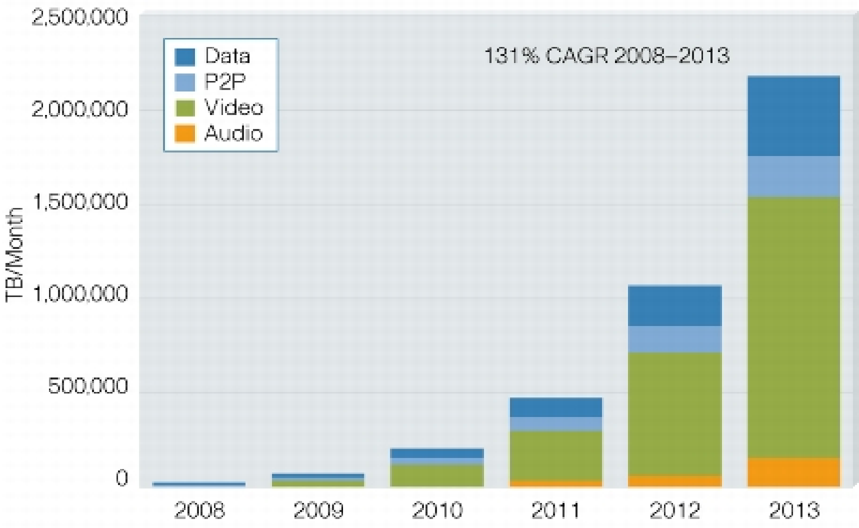


Figure 2-4. Forecast of mobile traffic growth from Cisco until year 2013. Mobile video alone will represent 64% of the traffic.

In Figure 2-4 can also be observed, independently from the application components, the exponential growth of traffic over the air with a CAGR (Compound Annual Growth Rate) of 131 percent between 2008 and 2013. In order to underline the extraordinary growing rate, it can be considered as a basis for comparison that the passage from 1 petabyte per month to 1 exabyte per month is going to take half of the time (from 2008 to 2012) it took fixed data traffic to do so (14 years).
A detailed breakout of the Cisco Global Mobile Data Traffic Forecast 2008-2013 is provided in Table 2-1.

IP Traffic 2008-2012							
	2008	2009	2010	2011	2012	2013	**CAGR 2008-2013**
By Application (TB per month)							
Audio	3612	7996	16930	35486	74503	154968	**1.12**
Video	13062	38681	107714	274820	650310	1390548	**1.54**
P2P	6714	15851	33784	69856	134224	220829	**1.01**
Data	9680	22547	48984	102054	217282	417847	**1.12**

Table 2-1 Detailed breakout of mobile traffic growth until year 2013 from Cisco.

In the light of this expected explosion of wireless traffic, it appears obvious that mobile backhaul networks have to catch up with all these changes and thus tend to counterbalance the enhancements on their two sides of the network (the core mobile network at one end, the cellular technologies at the BS site at the other end) in order to avoid a bottleneck effect between two highly capacitive edges. To achieve this goal, a significant change from TDM-based legacy backahuls is required as a response of the shift from voice-centric to data-centric traffic; the new scenario is as illustrated in Figure 2-5, to translate the exponential rise in traffic into revenues opportunity wireless operators have to move from traditional T1/E1 private or leased lines assessment.

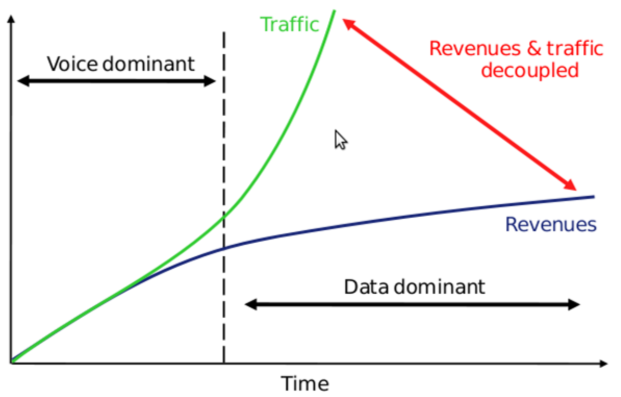


Figure 2-5. The evolving scenario along the time of mobile backhauls; an exponential growth of traffic does not translate in a related exponential growth of revenues. Into a data domain, T1/E1 lines are not a suitable solution anymore because to an exponential growth of traffic they answer to an exponential growth of costs as well.

While with voice traffic the amount of bandwidth provided to the customer is linearly dependent with the revenues, entering into the field of data-dominant traffic causes a decoupling of offered bandwidth and revenues: revenues per megabyte continues to fall due to increasing competition (mobility is a key driver for a subscriber satisfaction nowadays) and the introduction of flat-rate pricing. New network technologies are urgently required by the mobile operators to remain competitive.

2.2 Mobile Backhaul Evolution

As mentioned above, what is driving towards the Next Generation Mobile Network (NGMN) is the incapability of current backhaul assets to provide a cost-effective solution to the hunger of bandwidth of the newest data-centric services over the air. Currently most mobile operators provide 2G/3G-based voice-centric services, but new solutions are needed in order to keep the pace with the bandwidth requirements from the customers.
The current 2G and 3G services use TDM and ATM technologies respectively; the TDM traffic generated from 2G BTSs (Base Transceiver Stations) and ATM traffic from 3G Node-Bs (BTS equivalent) are multiplexed and transported into TDM backhauls. Today most cell towers are only connected by a few T1 or E1 private lines which are not sufficient to meet these new demands; an important factor to bear in mind which motivates for a drastic change into the radio access networks is that the cost of a T1-/E1-based backhaul increases with the bandwidth, just adding more T1/E1 connections at the cell tower might be a solution but it is deeply ineffective from an economical point of view. As of now, 8 to 12 T1s/E1s service a BTS.
Hence, before focusing on the evolution of mobile backhauls, it has to be provided an overview of the mobile technologies evolving over the air. A roadmap of service availability based on mobile access technologies is showed in Figure 2-6.

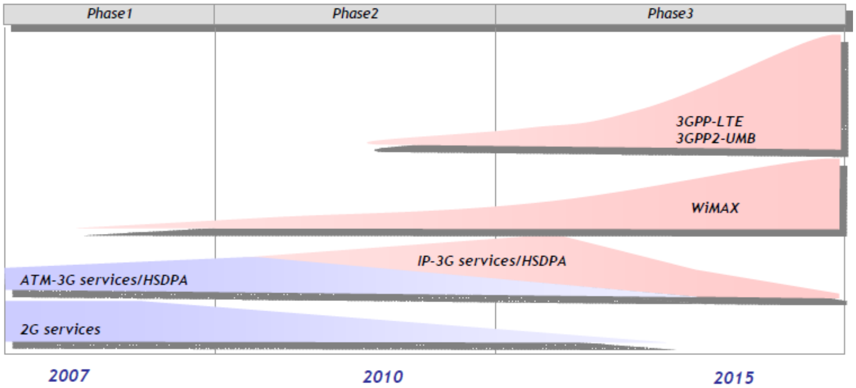


Figure 2-6. Roadmap of service availability. Three phases can be recognized: a first phase where 2G is dominant and 3G already available, a second phase where 2G is on a clear decline and 3G is growing, finally a third phase where fourth generation is rolled out and becoming dominant, 3G observes a downturn and 2G in not available anymore.

The 2G services are already in customer use, but the number of subscribers is forecasted to take a downward turn as the other services are made available. The 2G services, however, must continue to be supported for years until all of 2G subscribers shift to 3G or later services. The 3G services have already been available and gradually deployed on a steady basis. The broadband services, such as IP-based 3G and fourth generation services (WiMAX, 3GPP-LTE and 3GPP2-UMB) will increase and become dominant in the near future. While the WiMAX services are also available for the deployment, the 3GPP-LTE and 3GPP2-UMB are expected to be deployed starting in 2010. As can be observed again from Figure 2-6, the roadmap can be divided into 3 phases. In phase 1, current 2G/3G services are provided, in phase 2 newly IP-based 3g/HDSPA services are available; finally, in phase 3, 3GPP-LTE and 3GPP2-UMB services are growing.
Consequently, also the mobile backhaul are undergoing through an evolution which has the final goal to abandon TDM-based implementations and to move towards a packet-only architecture. With respect to the roadmap of service availability illustrated in Figure 2-6, also the architecture evolution will evolve through 3 phases with the ultimate goal of a single packet-base platform (Figure 2-7).

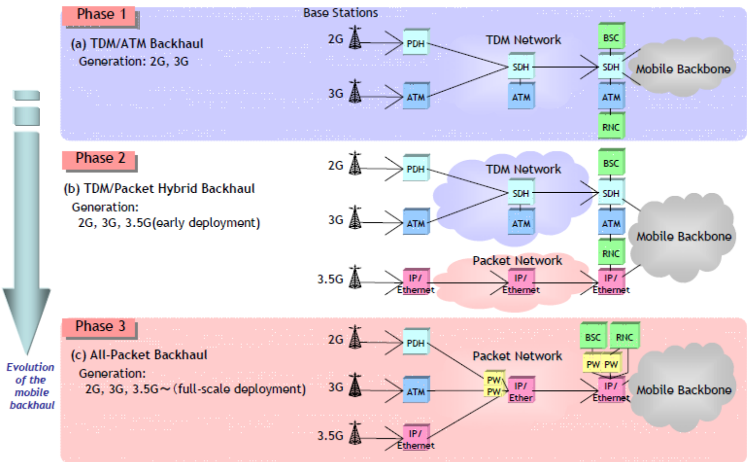


Figure 2-7. As the services offered over the air evolves from 2G to 4G, the mobile backhauls undergo an evolution which can be divided into 3 phases, related to the roadmap of Figure 2-6.

2.3 Carrier Ethernet as the future of Mobile Backhaul

In the following document a solution to apply into the new challenging scenario of mobile backhauls described above will be introduced, with particular focus on utilization of Ethernet, or more specifically on its extension Carrier Ethernet, as unique transmission protocol within the network.
Carrier Ethernet has been defined by MEF (Metro Ethernet Forum, a global industry alliance) as a "ubiquitous, standardized, carrier-class Service and Network defined by five attributes that distinguish it from familiar LAN based Ethernet: standardized services, scalability, reliability, quality of service and service management" [14] (Figure 2-8).

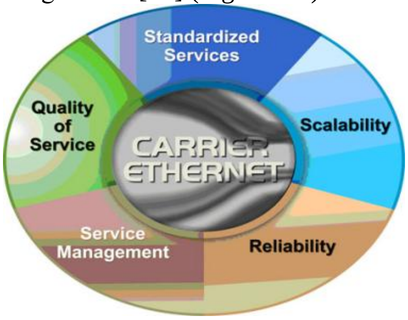


Figure 2-8. Five attributes distinguishing Carrier Ethernet from LAN based Ethernet: standardized services, scalability, reliability, quality of service and service management.

Carrier Ethernet is being broadly accepted as most suitable solution by service providers since it is more cost-effective than any other networking technology able to satisfy the restraining requirements of NGNM backhauls. We now hereafter dwell on the MEF 22 specification [7] to introduce to the reader the scenario in which our proposed solution is going to be inserted. Especially two aspects of the specification has to be born in mind along this document:

– it provides a common terminology that allows to abstract from referring to a particular mobile technology
– it provides use cases which present all the possible deployment scenario using Ethernet services

2.3.1 Common terminology

The reference model, which we will refer to in the sequent part of the document, is illustrated in Figure 2-9.

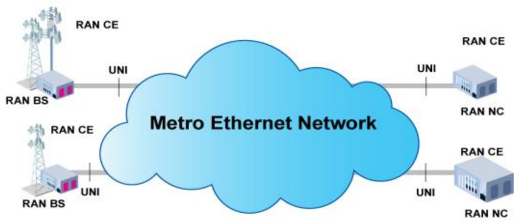


Figure 2-9. The reference model with the common terminology provided by Metro Ethernet Forum. The mobile backhaul is a Metro Ethernet Network (MEN) which connects the two RAN Customer Edges (RAN CEs): RAN Base Station site and RAN Network Controller. The User Network Interface (UNI) delimits the customer's area of responsibility from the provider's.

The mobile backhaul is a single Metro Ethernet Network (MEN) that connects the mobile network nodes, referred herein as RAN Customer Edge (RAN CE). On the base station side we refer to the RAN Base Station site, on the network controller side to RAN Network Controller site instead. A RAN NC may be a single network controller or a site composed of several network controllers; in the same way, a RAN BS can also be a single base station or a collection of several base stations.
The demarcation point, the User Network Interface (UNI), is the hand off point between the customer and the service provide; identifying where the responsibility of the customer and service provider begins and ends. A UNI is composed of two functional blocks: the UNI-C, which corresponds to the functions performed by the RAN site, and the UNI-N which specifies the functions performed by the MEN.
More complex scenario involving multiple MEN domains are out of the scope for this document.

2.3.2 Use Cases

In the following use cases that are going to be explained there is the foundation for all possible deployment scenario. Two different main use cases can be detected, that can be in turn divided into two sub-cases:

– RAN BS and RAN NC both equipped with Ethernet interfaces.
– RAN BS and RAN NC both with non-Ethernet based interfaces; the usage of a Generic Inter-working Function (GIWF) is needed.

In both use cases might be present or not a legacy network.

Use Cases 1a and 1b.
These are the cases where the RAN BS and the RAN NC can not be directly connected to a UNI because they have a non-Ethernet based interfaces, such as ATM or TDM. The RAN CEs then need to first connect to a Generic Inter-working Function (GIWF), which in turn is connected to a UNI.
In Figure 2-10 is illustrated the case of the presence of a legacy network in addition to the MEN; hence a split access scenario where there are two parallel networks that transport different types of mobile traffic. This might be the case where the service provider does not feel ready to abandon its legacy network, maybe because of a recent investment in its rollout or just not confident in handing over all the traffic on the MEN, thus low priority highly bandwidth-consumptive traffic is offloaded from the legacy network to the MEN in order to scale after network demand.

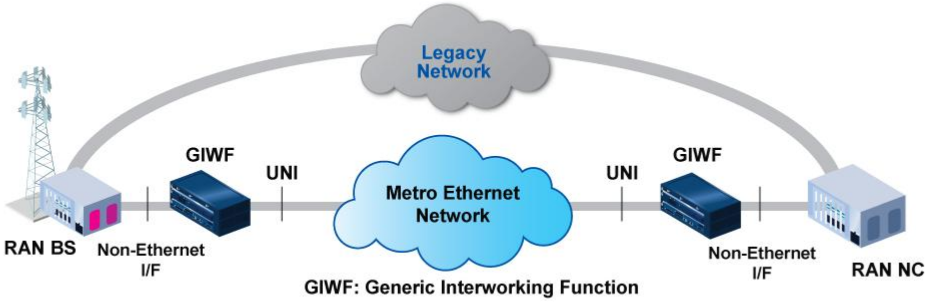


Figure 2-10. Use case 1a. A legacy network is present and the RAN CEs are not-Ethernet interface enabled; a GIWF (Generic Inter-working Function) is needed. On the MEN the low-priority highly bandwidth-consumptive traffic is offloaded.

In Figure 2-11the legacy network is not present anymore, it has been completely substituted by the Metro Ethernet Network and all the traffic from the RAN CE is transported over the MEN using Ethernet services via the GIWF.

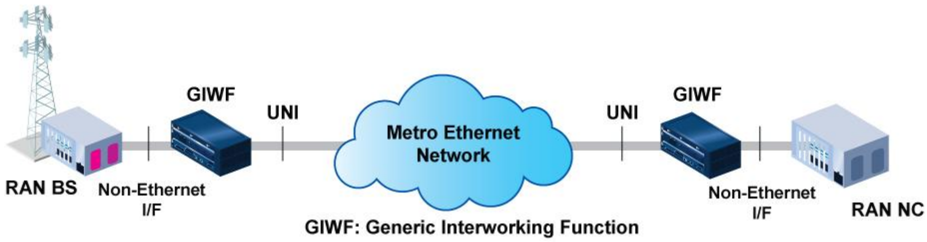


Figure 2-11. Use case 1b. The MEN carries all the traffic between the two RAN CEs, the legacy network has been entirely substituted; since they are not-Ethernet interface enabled, usage of GIWFs is needed.

Use cases 2a and 2b

Use cases 2a and 2b regard the RAN BS's and RAN NCs both equipped with Ethernet interfaces; RAN CE equipment can thus be connected directly to the UNI without the need of a intermediate GIWF. Use case 2a, illustrated in Figure 2-12, is similar to the previous use case 1a where the low priority highly bandwidth-consumptive traffic is offloaded into the MEN form the legacy network. The main focus here by the Metro Ethernet Forum is to specify the interworking (real-time traffic bearing, synchronization, …) only on the MEN, such operations via the legacy network is out of scope.

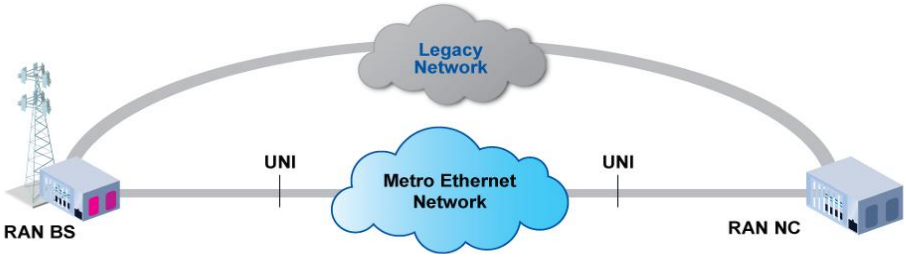


Figure 2-12. Use case 2a. A legacy network is present and the RAN CEs are Ethernet interface enabled; no GIWFs are needed. On the MEN the low-priority highly bandwidth-consumptive traffic is offloaded

When the legacy network has been completely removed and substituted (Figure 2-13), all the traffic and all the operations are committed to the Metro Ethernet Network, able to connect the RAN CEs without the intervention of GIWF.

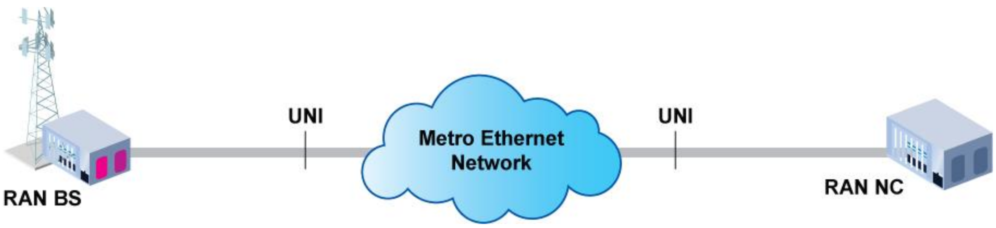


Figure 2-13. The MEN carries all the traffic between the two Ethernet enabled RAN CEs, the legacy network has been entirely substituted.

2.4 Next Generation Mobile Network backhaul requirements

Thus it will be committed to Carrier Ethernet the task of transporting the traffic within the backhaul; the main reason behind the choice of Carrier Ethernet by mobile operators is its cost-effectiveness and bandwidth-efficiency, but it cannot be forgotten that in backhauls detailed and strict requirements have to be satisfied.
A complete list of requirements that have to be matched by the backhaul can be found in [6] by the NGNM Alliance, an initiative by a group of leading mobile operators to provide a vision for technology evolution beyond 3G for the competitive delivery of broadband wireless services to increase further end-customer benefits.
Hereafter are listed the crucial points that have to be satisfied by the OpMiGua based mobile backhauls:

- R1: The NGMN Backhaul solution MUST allow connecting each e-NB to one or several aGW's (i.e. S1 interface in 3GPP LTE standard) for multi-homing purpose (e.g. S1-flex in 3GPP LTE standard) or multi- operator RAN Sharing reason.
  Typically up to 6 operators and 16 S1 interfaces per operator MAY be envisioned per e-NB.
  Typically up to 16000 S1 interfaces per operator MAY be envisioned per aGW.

- R2: The NGMN Backhaul solution MUST allow connecting each e-NB to one or several e-NB's (i.e. X2 interface in 3GPP LTE standard). This list of inter-e-NB connections MUST be operator configurable .

- R4: The e-NB/a GW Transport Module MUST map the radio QoS Class Identifier (QCI) to transport QoS markings (L2 an d/or L3 according to operator design choice)

- R12: The NGMN Backhaul solution MUST provide Peak/Average Bandwidth in a flexible and granular way. The NGMN Backhaul service bandwidth profiles, consisting of peak and committed information rates, SHOULD be configurable in increments of 2 Mbps between rates of 2-30 Mbps and increments of 10 Mbps up to 100 Mbps, and increments of 100 Mbps beyond 100 Mbps .

- R13: The NGMN Backhaul solution MAY provide up to 450/150 Mbps Downstream/Upstream

- R14: The NGMN Backhaul solution MUST provide at least 150/50 Mbps Downstream/Upstream Minimum Access Bandwidth

- R18: The NGMN Backhaul solution SHOULD be able to transfer all Multicast and Broadcast flows

- R36: The NGMN Backhaul solution MUST support clock distribution to the e-NB for frequency synchronization and SHOULD support phase/time alignment

This requirement R36 is going to be the platform of this thesis. It has to be noticed has the word 'MUST' is used.

Chapter 3

The OpMiGua Hybrid Packet/Time Slotted Circuit Switched Scheme (HPTS)

Whilst in the previous chapter we introduced mobile backhauls along with their evolution until the present time and the foreseen changes they will undergo to in the immediate future, in this chapter a suitable solution to be applied into mobile backhauls is presented. That consists of the OpMiGua concept (Optical packet-switched Migration-capable network with Guaranteed service), an hybrid network capable to provide on the same channel both circuit switching and packet switching Quality of Service (QoS). In the first place the OpMiGua project will be briefly introduced to the reader (Section 3.1), but the main focus is going to be on its time slotted scheme HPTS (Hybrid Packet/Time Slotted). The HPTS scheme (Chapter 2.2) is able to provide higher granularity and flexibility by adopting time-slots for circuit establishing. In Section 2.3 we provide a closer look to why the OpMiGua HPTS scheme offers itself as a suitable solution for mobile bakchauls

3.1 The OpMiGua Concept

The OpMiGua model [8] is a hybrid optical network that combines the best of packet switching (granularity, flexibility) with the best of circuit switching (dedicated path, no contention, constant low delay). The traffic is divided into two classes [9]:

– a circuit-switched Guaranteed Service Traffic (GST) class. Packets of the GST class follow lightpaths in the WRON (Wavelength Routed Optical Network) routing solution and possess a circuit-switched quality of service (QoS) ;

– a packet-switched Statistically Multiplexed (SM) service class; packets belonging to this class are processed at every node.

The SM packets are inserted between GST bursts in order to avoid under-utilization of available bandwidth. A WRON network is in fact subjected to coarse granularity; the granularity of a circuit coincides with the capacity of a single wavelength. This is counter-acted by filling gaps between two subsequent GST packets or bursts with packets belonging to the SM class. A schematic illustration of a three-node OpMiGua hybrid network is provide in Figure 3-1.

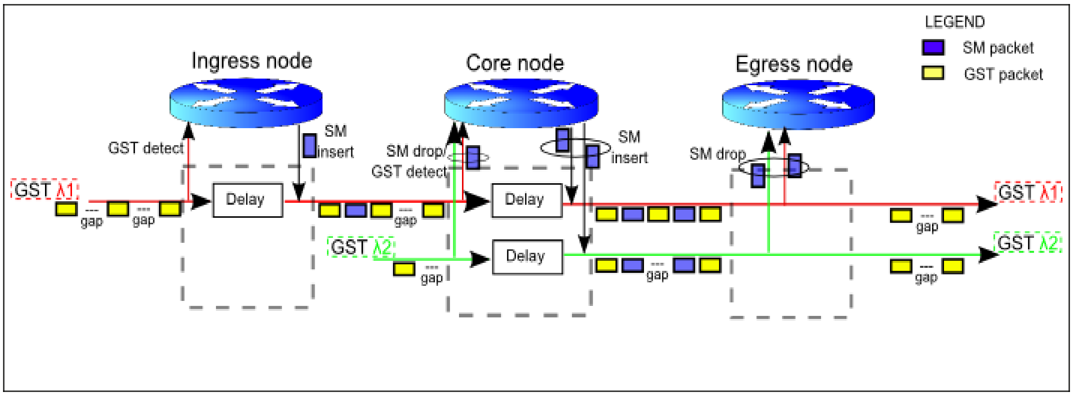


Figure 3-1. An OpMiGua hybrid network consisting of three nodes: ingress, core and egress node. Wavelength λ_1(red) transports GST packets from the ingress node to the egress node, wavelength λ_2 (green) from the core node to the egress node. SM packets are inserted into the gaps between GST packets, disregarding of the wavelength. They are extracted and processed on an hop-by-hop base.

For transmission and separation of the two classes the Polarization and Time Division Multiplexing (PolTDM) scheme is used in OpMiGua [15], as illustrated in Figure 3-2. Thus the State of Polarization (SoP) is used to distinguish the two classes (GST packets are transmitted in one SoP, while SM packets are transmitted on the orthogonal SoP) and the two different SoPs are not transmitted simultaneously but interleaved in time. Comparing PolTDM with traditional polarization multiplexing technique (PolMUX), the former avoids coherent crosstalk between the two service classes and reduces the effects of polarization dependent loss (PDL) and polarization-mode dispersion (PMD).

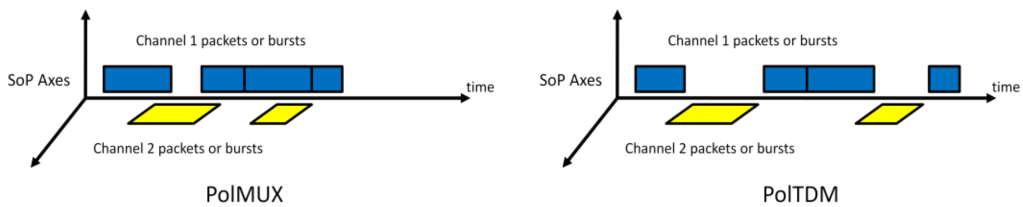


Figure 3-2. Traditional polarization multiplexing (PolMUX, on the left) and Polarization and Time Division Multiplexing (PolTDM, on the right) in comparison. Whilst in PolMUX technique traffic is present simultaneously on Channel 1 and Channel 2, with PolTDM method only one packet at a time is present on both channels. In PolTDM technique, the two channels represents two different classes of service.

For a more complete treatment of the OpMiGua project the reader should refer to [8] and [16].

3.2 The HPTS scheme

If the OpMiGua hybrid network might represent an important further step in the drastic changes brought by optical techniques into the telecommunications, it finds its limitations when a few, or a single, channels (i.e. wavelengths) are available; this is the typical use case in mobile backhauls. In standard OpMiGua circuits are wavelength-paths following the WRON scheme; as a consequence the capacity of a link is dictated by the number of GST circuits traversing the link and the granularity of the wavelengths. In a scenario where the number of wavelength is constrained, OpMiGua standard (unslotted) scheme is not a suitable solution anymore.

For this purpose, the Hybrid Packet/Time Slotted Circuit Switched Scheme (HPTS) has been proposed in order to enable higher granularity and higher number of circuits upon the same number of wavelengths by employing time-slots [10]. The basic idea is as follows: to divide the GST wavelength paths into time-slots, allowing different circuits to share the same wavelength by time-slot allocation. It gives the benefits of a higher granularity (determined by the time-slot and not by the wavelength capacity), higher benefits and, as said above, it allows to setup an higher number of circuits upon the same wavelength. Those benefits translate into a lower grade of under-utilized resources. The main drawbacks is a higher complexity of the node compared with the original OpMiGua concept: the costs involved with a higher grade of complexity have to be counter-balanced by saving in capacity. When this condition is satisfied, OpMiGua HPTS scheme represents a viable approach.
The HPTS Scheme has its best exploitation in the optical domain, where the State of Polarization can be used as label to differentiate GST (circuit switching quality) and SM packets (statistically multiplexed) but it is a suitable solution also in the electrical domain; in the latter case, service class differentiation has to be provided within the framing technique.
By dividing the GST wavelength-path into time-slots, it is possible to assign every time-slot to a different circuit; in this way a GST packet is forwarded not only based on the wavelength, but also on the circuit the time-slot has been assigned to. Circuit quality is still guaranteed since the packet is not processed in intermediate nodes, but the number of circuits that can be set up upon the same wavelength is considerably higher. Also flexibility in circuit capacity is provided, committing to the control plane the time-slots assignment to the different circuits. As a result, GST packets are scheduled in time-slots allocated for the ingress node and have the priority to fill the whole capacity of the time-slot; SM packets are instead allowed to be inserted into unused space (capacity) in a time-slot allocated to a GST path or into unallocated time-slots. SM packets are processed at all intermediate nodes and forwarded according to the packet header.
Hereafter an example of three-node OpMiGua HPTS network is showed (Figure 3-3)

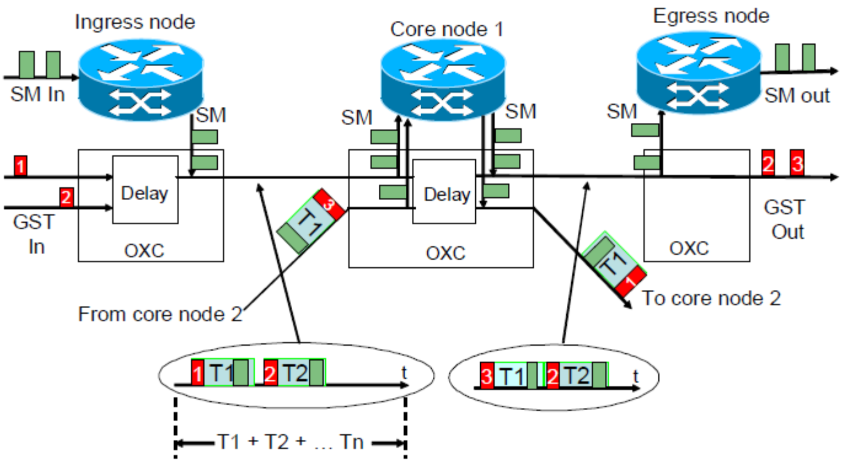


Figure 3-3. Example of a three-node OpMiGua HPTS network: ingress, core and egress nodes.

It can be seen how the GST packet (red) labeled as 2 is inserted into the time-slot T2 and is not extracted until its destination node (i.e. the egress node). On the other hand the GST packet labeled as 1 has the core node as its address. It is thus inserted in time-slot T1 and extracted at the destination node. Its time slot is reused by the GST packet 3, addressed to the core network. The SM packets (green) make use of the unutilized resources (empty gaps in the time-slots)

3.3 OpMiGua HPTS scheme & mobile backhauls

The OpMiGua HPTS scheme has the main benefit to provide a higher granularity to the GST circuit paths: whilst the original OpMiGua concept has a bandwidth granularity of a wavelength, introduction of time-slots allows a finer granularity. This enhancement represents an important added value when the resources available are in scarce quantity, which is the very case of mobile backhauls. As shown in Chapter 2, the mobile backhaul is put under pressure from the hunger for bandwidth of high-end handsets and laptop cards; while available bandwidth is not an issue in the fiber-based core networks and, in the same time, air interfaces have considerably increased their spectral efficiency, mobile backhauls run the risk of creating a bottleneck effect. Even where they make usage of wirelines (whether copper or fibers) instead of point-to-point microwave and millimeter-wave links, a few or a single channel (i.e. wavelength) are available. In this scenario the HPTS scheme comes to good rescue: it allows to adopt the OpMiGua concept without the need of a plurality of channels. Through its hybrid approach, both packet-switched and circuit-switched QoS can be provided while operating on the same channel; the capacity of providing circuit-switched QoS can be a fundamental added value for some mobile applications that are particularly constraining in terms of end-to-end delay, jitter and packet loss rate due to packet contention. As stated in Section 1.1, video will represent the biggest slice of traffic carried over mobile backhaul; this very application implies wide bandwidth consumption and strict limits for end-to-end delay, as well as quality degradation due to variable jitter and loss of packets. Being able to provide a circuit-switched quality to this family of applications and in the same time allowing lower-priority packets to be inserted and treated as a packet-switched service class translates into the possibility for mobile operators to offer quality-guaranteed services to their customer base with efficient resources utilization.

Chapter 4

Synchronization over a packet network: IEEE 1588v2 Precision Time Protocol

With the switch from TDM-based to all-packet mobile backhauls, the issue of synchronization among RAN BSs becomes challenging. The TDM signal in fact can be used as a source of synchronization itself. Packet-based networks are asynchronous, alternative techniques have to be adopted. Among those, the IEEE 1588v2 has experienced a good success. This protocol, aka PTP (Precision Time Protocol) is presented in this chapter.

In Section 4.1 the issue of synchronization is presented. Two different approaches in order to substitute the TDM signal are presented: physical-layer and packet-layer. The choice of a packet-layer approach is thus motivated. In Section 4.2 the most successful among the packet-layer based solutions is introduced: the IEEE 1588v2 protocol, aka PTP. The main improvements with the release of version 2 are also listed. In section 4.3 the types of messages exploited by the protocol are presented. The specific issue of utilizing an hardware for the creation of timestamps is dealt with in Section 4.4. How the synchronization process is carried out is exposed step-by-step in Section 4.5. In Section 4.6 is presented how PTP messages are mapped over UDP/IP/Ethernet or over Ethernet. The format of the PTP messages is illustrated in Section 4.7. In Section 4.8 it is dealt with the issues affecting the timing performances (i.e. the accuracy) of the protocol over packet networks. The specific case of an Ethernet network is introduced in Section 4.9. The consequences of the timing impairments upon the accuracy of the protocol is the focus of Section 4.10.

In this chapter many detail of the IEEE 1588v2 protocol are presented. Those are fundamental because are the same principles our synchronization technique is based on. The results achieved with our proposal have to be compared with those presented in this chapter.

4.1 The issue of synchronization over an asynchronous networks: from PDH and SONET/SDH to Ethernet

In typical 2G/3G networks, RAN BSs can derive synchronization from the conventional PDH or SONET/SDH leased lines used to backhaul the traffic; the TDM signal is used as a source of synchronization itself. The TDM input clock is traceable back to the wireline carrier’s Primary Reference Clock (PRC). Over the long term this makes an extremely accurate reference, better than 1 part per 10^{11}. Against possible frequency variations over the short term, a Phased Locked Loop (PLL) can be used for filtering and provision of a stable clock reference [11].

For PDH and SONET/SDH networks, limits on the magnitude of jitter and wander at network interfaces are set in the ITU-T recommendations G.823 (digital networks based on the 2048 kbit/sec hierarchy, i.e. E circuits) [17], G.824 (digital networks based on the 1544 kbit/sec hierarchy, i.e. T circuits) [18] and G.825 (digital networks based on the synchronous digital hierarchy, i.e. SDH)

[19]. The following values have to be respected:

- o GSM, WCDMA and CDMA2000 require frequency accuracy of 0.05 ppm (parts per million, 1 ppm results in $\frac{1\,\mu sec}{1\,sec}$) at the air interface
- o CDMA2000 requires time synchronization at ±3 µsec level (±10 µsec worst case)
- o WCDMA TDD mode requires 2.5 µsec time accuracy between neighboring Base Stations.

Since Ethernet is an asynchronous transport technology which does not natively provide any synchronization services on its own, an alternative way to carry the synchronization information along the network has to be provided in order to overcome the challenge. Key guidelines have been provided by ITU-T in April 2006 with the recommendation ITU-T G.8261 [12] leading to two different approaches: either physical-layer or packet-layer based.

The first approach consists of embedding a clock distribution signal in the line code, the timing (frequency) is distributed at the physical layer; this synchronization method, similar to the one currently used in SONET/SDH networks, is physical-layer dependent. Since, as mentioned in Section 2.3, Carrier Ethernet is being broadly accepted as most suitable solution by service providers, it is worth to mention the method Synchronous Ethernet. Defined in the recommendation ITU-T G.2862 [20] and at the present time considered an emerging technology, Synchronous Ethernet is extremely accurate but with the main drawback of being required in each node in the network for the end-to-end synchronization. It provides for a frequency offset range of 4.6 ppm by means of an Ethernet Synchronization Status Messaging (ESMC) channel; similar to SONET/SDH Synchronization Status Messaging (SSM) bytes that allow nodes to deliver their synchronization status to downstream nodes [21]. As specified in [12], [20] and [21] Synchronous Ethernet equipment must be able to operate with SONET/SDH existing installation and must be able to perform both sync and non-sync operations modes. In non-sync operational mode Synchronous Ethernet is identical to IEEE 802.3.

The second approach instead is based on distributing timing (frequency and/or time) in the packet layer using a dedicated IP/Ethernet packet stream. In order to synchronize the clocks of the two end nodes into the network, two methods can be made used of:

- timestamp exchanging protocols (IEEE 1588v2 Precision Time Protocol or IETF Network Time Protocol): the Base Station can reconstruct original time and frequency information by using the timestamps of the received packets and sophisticated algorithms which elaborate the received information.
- packet inter-arrival times: by usage of the time gaps between the reception of two following packets, Adaptive Clock Recovery (ACR) methods can reconstruct the original frequency information.

The main advantage of the packet-layer based approach over the physical-layer based, is that the former is physical-layer independent and is flexible to interoperate with a wide variety of access transport layers (e.g. fiber, copper, microwave,…). The relevant drawback is to be vulnerable to packet network impairment, such as end-to-end delay, delay variation and frame/packet loss; an accurate network engineering is important for the effectiveness of the packet-layer base method.

Apart from the consideration of Synchronous Ethernet being still under development, in a scenario like the mobile backhauls it looks more convenient and reasonable to focus on the packet-based approach.

Mainly two aspects work in favour of the packet-based approach: its feature of being physical-layer independent and the capability of distributing multiple synchronization signals. The former is an important added value in the backhauling network since a plurality of link typologies might be present: optical fibers, copper wires, microwave bridges, optical wireless. The latter instead provides for separate synchronization processes to mobile service providers that operate on the same RAN but each of them wants to refer to its own reference clocks. This is also a likely situation when dealing with mobile backhauls.

Among the packet-layer based approaches, the one that received most of the attention and experienced the highest popularity among network architects, planner and engineers is the IEEE 1588v2 protocol, also called PTP. We thus adopt this as platform for timing distribution into the mobile backhaul. Hereafter the fundamentals of the PTP are presented; they have to be born in mind for our proposal which is going to be shown from Chapter 6.

4.2 IEEE-1588v2 Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control System

Published as IEEE 1588-2002 on November 8th, 2002 [22], and followed by the release of IEEE 1588-2008 [13] (also known as IEEE 1588v2) some years later, this successful synchronization protocol was introduced in order to "fill a niche not well served by either of the two already existing protocol, NTP (Network Time Protocol) and GPS (Global Position System)" [23]. IEEE 1588 is addressed to local systems requiring very high accuracies beyond those attainable using NTP; it is also designed for application that cannot bear the cost of a GPS receiver at each node, of for which GPS signals are inaccessible. The IEEE 1588 is also called with the acronym PTP, Precision Time Protocol, a term defined into the same standard (Clause 3.18) and that will be used throughout this document; it was used during the development of the technology prior to standardization and carried over into the text of the standard by the responsible working group. A comparison upon the main features of IEEE 1588v2, NTP and GPS-based timing protocols is provided in Table 4-1.

	IEEE 1588v2	NTP	GPS
Spatial extent	A few subnets	Wide Area	Wide Area
Communications	Network	Internet	Satellite
Target accuracy	Sub-microseconds	Few milliseconds	Sub-microsecond
Style	Master/Slave	Peer ensemble	Client/Server
Resources	Small network message and computation footprint	Moderate network and computation footprint	Moderate computation footprint
Latency correction	Yes	Yes	Yes
Administration	Self organizing	Configured	N/A
Hardware?	For highest accuracy	No	RF receiver and processor
Update interval	~2 seconds, variable in version 2	Varies, nominally seconds	~1 second

Table 4-1. Comparison between IEEE 1588v2 and other two widely used timing protocols, i.e. NTP and GPS-based. The main features of the three protocols for their functioning are considered.

With regard to the target accuracies of the three different protocols, the values are reported in Table 4-2.

	IEEE 1588v2	NTP	GPS
Target accuracy	Sub-microseconds	10 milliseconds-200 microseconds	10 nanoseconds -100 nanoseconds

Table 4-2. Target accuracies of IEEE 1588v2, NTP and GPS

In the packet-layer based approach of IEEE 1588v2, the accuracy of the timing algorithm is correlated with the performances of the network that the packets are carried by. Impairments introduced by the network (i.e. packet loss, bit errors, packet delay, …) affect the accuracy with different weights, as it will be seen in Section 4.8; hence a unambiguous value cannot be provided, but a target accuracy of sub-microseconds is stated into the protocol (Clauses 1.1 and 1.2). Over a suitable, well-designed network, by means of PTP a time accuracy better than 1 microsecond and a frequency accuracy of better than 10pbb are achievable [11].
The Network Time Protocol, as said above, cannot satisfy the accuracy required by cellar mobile networks; over the public Internet is common an accuracy of 10 milliseconds, while under ideal conditions an accuracy of 200 microseconds can be reached. As last, the GPS target accuracy: it is determined by the ICD-GPS-200 (Interface Control Document of the Global Position System), Clause 3.3.1.5 at the value of ±100 nanoseconds [25]. This value has not been modified through the following revisions, but unofficially an accuracy of 10 nanoseconds is the common target for vendors.

4.2.1 PTP version 2

As stated in the very first Clause of the IEEE 1588-2002 standard, all IEEE standards are subject to mandatory reviews. The standard may be modified at any time following the procedures defined by the IEEE, and must be reviewed every 5 years. This is due to the fact that standards have to evolve to meet the needs of the market and changes in technology.
As mentioned above, the IEEE 1588 standard was ratified in 2002. It was originally designed for the industrial automation and test and measurement industry; in those fields has been in use for several years. After few years, in 2005, a new project was started in order to revise the standard: better performances wanted to be achieved both to be used in its original intended application space and to allow the protocol to be suitable for telecom applications. The version 2 of the protocol saw the light of the day in 2008 with its release under the name IEEE 1588-2008 (aka IEEE 1588v2). The main enhancements from version 1 to version 2 that are relevant for the telecom applications are:

- Unicast in addition to multicast support. In the first release of PTP, only support for multicast message exchanges was specified. However, in a telecom environment the support for unicast messages is required in order to reach finer tuning and optimize network performances. Unicast minimizes the bandwidth consumption and the processing power required at the client site.
- Shorter message formats. Instead of a unique 165-octet long message containing both information about the clock source and the timestamp as defined in version 1, in the second release the two parts are split in two separate messages. An announce message is foreseen to contain the details about the source clock while the synchronization timestamps are carried by smaller 44 octet PDUs.
- Specified transport over more network layer protocols. Transport of synchronization PDUs over more network is specified in version 2 of PTP, e.g. Ethernet, UDP/IPv4, UDP/IPv6. It gives operators the possibility to better manage how to allocate resources according to network transport layer availability and packet type.
- Higher update rates allowed. Based on the IEEE 1588-2002 version, PTP masters and clients send synchronization messages every 30 seconds by default. Into a telecom environment, much higher messages rates are required: into the version 2 an update rate up to 64 times higher is defined. Version 2 allows also to adjust those rates as needed according to bandwidth availability and timing performances at the clock sites.
- Best Master Clock (BMC) algorithm. This algorithm is defined into the version 2 and has to be supported from all the PTP clients. It allows the PTP clients to auto-configure themselves in case they lose their timing reference. The network is scanned and the quality

of the timestamps received taken into account, through the BMC algorithm the best replacement as timing reference is identified.

4.3 The Message types in IEEE 1588v2

As said above, IEEE 1588v2 is based upon a message exchange between the master and the slave clock. Seven types of messages are defined and grouped into two categories, i.e. event messages and general messages, as shown in Table 4-3.

Event messages	**General messages**
Sync	Announce
Delay_Req	Follow_Up
	Delay_Resp
	Management
	Signaling

Table 4-3. Message types defined into IEEE 1588v2

Event messages trigger the generation of a timestamp at both the sending and the receiving clock; general messages are used for management and control purposes or for passive transport of timestamps, i.e. without triggering the mechanism for timestamp generation.
By means of part of the messages illustrated into Table 4-3, the communication between the two ends is carried out as illustrated in Figure 4-1.

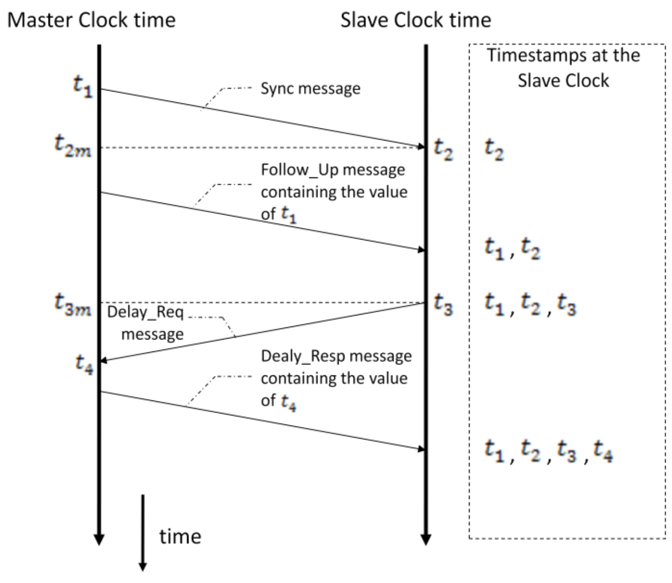


Figure 4-1. Message exchange between the Master Clock and the Slave Clock in order to synchronize the Slave clock on the same time beaten by the Master Clock.

Hereafter a closer look to the 7 types of messages is provided: what they trigger at the timestamp points, which values they carry and how these values contribute into the computation for the synchronization of the Slave Clock to the Master Clock. The time values refer to those illustrated in Figure 4-1.

Sync Message. Issued by clocks in the Master state, it contains an estimate of the sending time ($\sim t_1$); the timestamp is an estimated value because calculated far from the physical layer (e.g. at the application layer). The important feature of this message is its receipt time t_2 from the slave clock.

Follow_Up message. Issued by clocks in the Master state and always associated with the preceding Sync message, it contains the precise sending time t_1 as measured as close as possible to the physical layer of the network. This precise value of t_1 is used in computations rather than the estimated sending time contained in the Sync message.

Delay_Req message. Issued by clocks in the Slave state, it is sent towards the master clock; the sending time t_3 is measured and saved by the slave. At the master side, the receipt time t_4 is noted.

Delay_Resp message. Issued by clocks in the Master state and always associated with a preceding Delay_Req message from a specific clock (no broadcasted), it contains the receipt time t_4 of the associated Delay_Req message.

Management messages. Used for management purposes, they provide external visibility to the data sets maintained within each clock as well as a mechanism to modify those data sets. They can also be exploited to force changes and modifications in states and times of clocks.

Announce messages. There are used to indicate the capabilities of a clock to the other clocks belonging to the same subdomain; the concept of subdomain will be further explained in the following. The information carried from an announce message are used from the BMC algorithm for the master-slave hierarchy establishment.

Signaling messages. Used for signaling and control purposes over the network, their utilization is vendor dependent.

4.4 On the presence of hardware for precise timestamping

As mentioned in the previous section, event messages generate a timestamp when they are both sent or received: it can be done in hardware, ISR (Interrupt Service Routine) or kernel, or at application level. High precision can only be achieved if timestamping is performed in hardware, close to the physical layer, so that the addition of a possible jitter from the software is eliminated. A pulse is generated when the IEEE 1588 message is passing to/from the physical layer, and this event triggers the copying of a counter value (i.e. the clock timestamp) to a register.

We here assume the utilization of an hardware component for precise timestamping; although this hardware is beyond the scope of this document, the assumption of its adoption is important to be aware of which chain of messages we are going to deal with. In fact performing the timestamping with an hardware add-in implies the presence of the Follow_Up message and thus a 4-message chain. The Follow_Up message is not mandatory but a on-the-fly message modification would request a more complicated structure.

The structure we are going to utilize to explain the 4-message chain is thus as illustrated in Figure 4-2, with the presence of the timestamp point included.

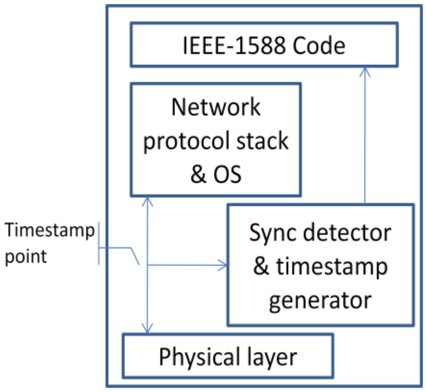


Figure 4-2. Block structure of a node with unique regard to the clock synchronization. A Timestamp point is present close to the physical layer for an extremely precise timestamping; the synchronization process is started form the network protocol stack and carried out with the support of the IEEE-1588 Code.

4.5 The synchronization process

In the normal execution of the IEEE 1588v2 protocol, the synchronization process consists of two phases:

- In phase 1 the Master-Slave hierarchy among the clock is established.
- In phase 2 each slave synchronizes to its master by an exchange of the 4-message chain illustrated in Figure 4-1.

We hereafter illustrate the two phases in detail and in conclusion is reported the timing algorithm which makes usage of the exchanged timestamps.

4.5.1 Establishment of the Master-Slave hierarchy

Before starting the message exchange for clock synchronization, a hierarchical master-slave structure has to be established among the clocks. On top of the hierarchy is located the Grandmaster Clock, which determines the time base for the whole system; the time in each clock within the system has to be as close as possible to the one beaten by the Grandmaster clock. Its first slave is the Slave of the Grandmaster clock and, in turn, master of its slave; from master to slave, the whole hierarchy can be climbed down level of hierarchy by level of hierarchy. The clocks organization can be briefly summarized as illustrated in Figure 4-3.

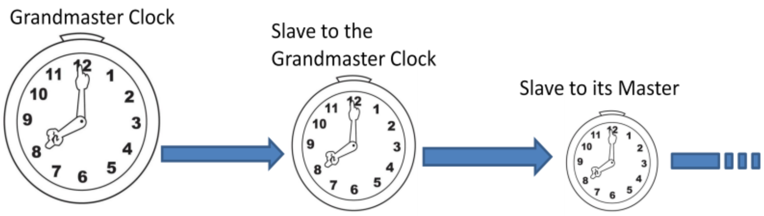


Figure 4-3. Clocks' master-slave organization. The Grandmaster clock determines the time base for the system; the slave of the Grandmaster Clock is in turn master of its slave. The master-slave organization defines a parent-child hierarchy of master-slave clocks.

In a network which utilizes IEEE 1588v2 for clock synchronization, the master-slave hierarchy is self organized according to clock characteristics and network topology: these information are contained into the Announce messages and processed by the other clocks through the "Best Master Clock" algorithm. The latter is defined into the body of IEEE 1588v2 and run by all clocks:

- A clock at startup to determine it should enter into the network as a master rather than as a slave.
- A master clock may receive Announce messages from other clocks ('foreign masters') who are currently in a master position and run the BMC algorithm to determine whether it should remain master or yield to a foreign master.
- Each non-master clock to determine whether it should become a master or which is the best clock to elect as its master.

4.5.2 The 4-message exchange

In Figure 4-4 the steps at both master and slave clocks are summarized.

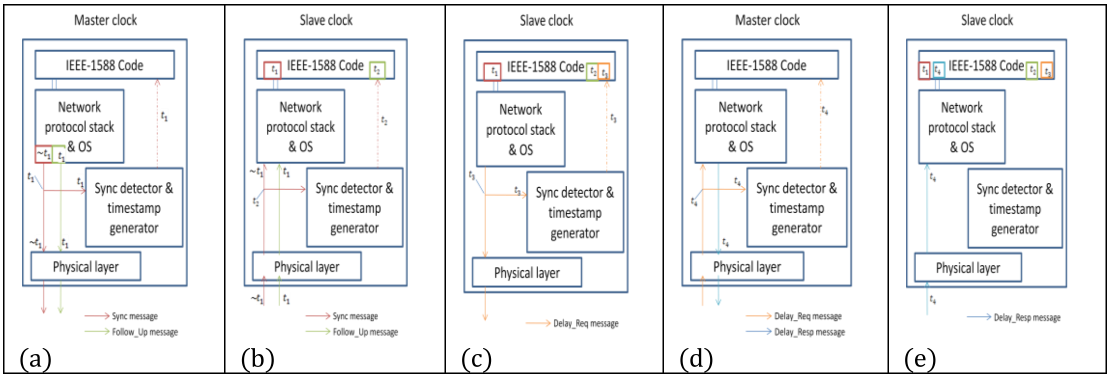


Figure 4-4. The message exchange with the timing information exchanged and stored at the slave IEEE-1588 Code for offset calculation between the Master Clock and the Slave Clock.

The Master clocks starts the synchronization process sending a Sync Message (Figure 4-4.a) with the estimated (since calculated 'far' from the physical layer) sending time $\sim t_1$. When going through the timestamp point, the Sync message triggers the sync detector which samples the exact sending time t_1, hands it over to the IEEE 1588v2 Code first and in turn to the network protocol stack; its duty of the latter to insert it in a Follow_Up message and send it to the Slave Clock. The Slave Clock receives the Sync message first (Figure 4-4.b); when it goes through the timestamp point, the exact receiving time t_2 is sampled and stored into the IEEE 1588v2 Code. The sending time $t_1$contained in the Follow_Up message is stored as well. The slave clock itself then sends a Delay_Req message (Figure 4-4.c) to the Master Clock; no particular timing information are contained into the message, but it triggers the sync detector which stores the exact sending time t_3 into the IEEE 1588v2 Code. The Delay_Req message will in turn trigger the sync detector at the Master Clock's site (Figure 4-4.d); the exact receiving time t_4 is sampled at the timestamp point and inserted into a Delay_Resp message to be sent to the Slave Clock. The time t_4 is finally inserted into the IEEE-1588 Code at the Slave Clock's site in order to carry out the offset calculation (Figure 4-4.e).

4.5.3 The final Master-Slave offset computation

For sake of completion, how the synchronization is carried out through the message exchange is now presented. It has to be noticed that through the IEEE 1588v2 protocol the offset between the master clock and the slave clock can be computed; but how the slave node adjust its local clock from this offset is not specified into the PTP.
The quantities that the computation refers to are illustrated in Figure 4-1.

- The measured difference between the time a Sync message is received at the slave and the time it was sent by the master:
$$ms_{difference} = t_2 - t_1$$
- The measured difference between the time a Delay_Req message is received at the master and the time it was sent by the slave:
$$sm_{difference} = t_4 - t_3$$
- The actual offset between the time observed on the master and slave clocks:
$$offset = t_{slave} - t_{master}$$
- The actual propagation time for a Sync message traveling between the master and slave clocks:
$$ms_{delay} = t_{2m} - t_1$$
- The actual propagation time for a Delay_Req message traveling between the slave and master clocks:
$$sm_{delay} = t_4 - t_{3m}$$

The following relationships are derived from first principles:
$$\begin{cases} ms_{difference} = offset + ms_{delay} \\ sm_{difference} = -offset + sm_{delay} \end{cases}$$
Rearranging the two equations above:
$$\begin{cases} offset = \dfrac{(ms_{difference} - sm_{difference}) - (ms_{delay} - sm_{delay})}{2} \\ ms_{delay} + sm_{delay} = ms_{difference} + sm_{difference} \end{cases}$$
We obtain as a result two equations involving two measured quantities (i.e. $ms_{difference}$ and $sm_{difference}$) and three unknowns (i.e. $offset, ms_{delay}$ and sm_{delay}); it is clearly impossible to solve the obtained system without either independent measurements or assumptions. It is a common problem shared by all synchronization protocols with timing information carried over channels with unknown propagations delay. The assumptions is represented by the following equation:
$$sm_{delay} = ms_{delay} = delay_{one-way}$$
which means that the two propagation times are assumed to be equal. If the error caused by this assumption is significant relative to the required synchronization accuracy, then some sort of calibration correction must be applied. Thanks to the choice of the cluster-type typology, this assumption is going to be consistent in all its aspects within the ring.

4.6 Transport of PTP messages

As mentioned in Subsection 4.2.1, the specification for transport of PTP messages over more network layer protocols is listed among the enhancements brought by the version 2 of the protocol. It is of our particular interest to see how a PTP message is transported over a Ethernet-based network, as Carrier Ethernet is being broadly accepted as most suitable solution by service providers.
Over the Ethernet, a PTP message is commonly carried as UDP/IP packet and hence mapped over the Ethernet or alternatively mapped straight into the IEEE 802.3 frame. The IP protocol can be both

version 4 and version 6. In case of PTP over UDP over IP, the IEEE 1588v2 messages is mapped as illustrated in Figure 4-5. The UDP datagram is identified as a PTP message by the UDP port number. It is also possible to map the PTP message directly into the IEEE 802.3 as shown in Figure 4-6. In the latter case, it is the type field within the Ethernet frame set to 0x88F7 which identifies the PTP message as such.

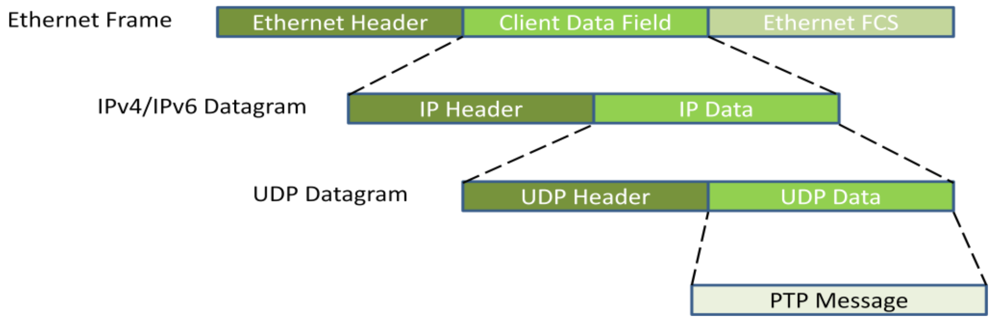


Figure 4-5. Mapping of a PTP message over UDP over IP over Ethernet

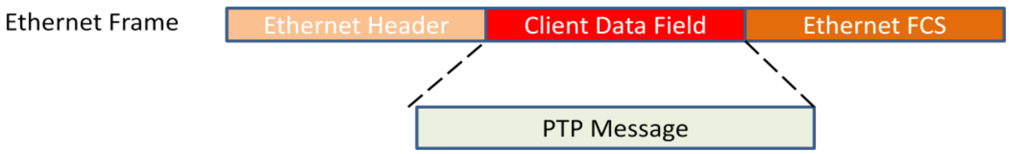


Figure 4-6. Mapping of a PTP message over IEEE 802.3/Ethernet

4.7 PTP message formats

For sake of completion and in order to have a clear idea of the bandwidth consumption required for traditional PTP packets, a brief hint on the message formats established by IEEE 1588v2 is provided below. For a detailed description of the fields the reader can refer to [13]. As shown in Figure 4-7, the PTP message consists of a header, a body and an optional suffix. The header has a fixed length of 34 octets while the size of the body depends on the type of message which is carried.

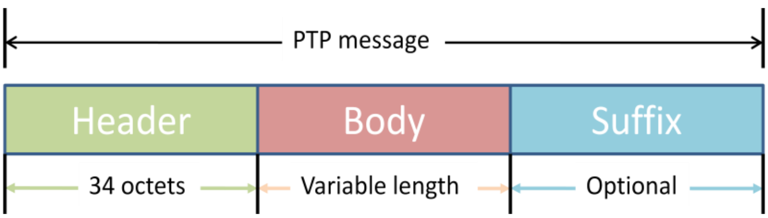


Figure 4-7. Basic PTP message format

The body of a Synch, Follow_Up and Delay_Req messages is long 10 octets (i.e. it consists of the timestamp only), while the Delay_Resp message contains the requesting port besides the timestamp in order to associate it with the related Dely_Req message, for a total length of 20 octets. These four messages thus present the format illustrated in Figure 4-8.

Bits							
1	2	3	4	5	6	7	8
PTP header [34 octet]							
Timestamp (Origin timestamp for Sync,Follow_Up, Delay_Req messages) (Received timestamp for Delay_Resp message) [10 octets]							
RequestingPortIdentity (Delay_Resp only) [10 octets]							

Figure 4-8. General format for the body of a Sync, Follow_Up, Delay_Req or Delay_Resp message

Signaling and Management messages are differently structured, their formats are illustrated in Figure 4-9 and Figure 4-10 respectively.

Bits							
1	2	3	4	5	6	7	8
PTP header [34 octet]							
TargetPortIdentity [10 octets]							
One or more TLV [N octets]							

Figure 4-9. Format of a Signaling message

Bits							
1	2	3	4	5	6	7	8
PTP header [34 octets]							
TargetPortIdentity [10 octets]							
StartingBoundaryHops [1 octet]							
BoundaryHops [1 octet]							
Reserved [4 bits]				ActionField [4 bits]			
Reserved [1 octet]							
One or More ManagementTLV [N octets]							

Figure 4-10. Format of a Management message

Of particular interest is the usage in both signaling and management PTP messages of the TLV fields. In Figure 4-11 the format of a Management TLV is shown in detail; on this field we will come back in a second moment in the next part of this document.

Bits							
1	2	3	4	5	6	7	8

tlvTtype [2 octets]
lengthField [2 octets]
ManagementID [2 octets]
dataField [N octets]

Figure 4-11. Format of a Management TLV field

4.8 Issues affecting timing performance in packet networks

In a protocol such as IEEE 1588 which is based on exchange of packets to establish accurate synchronization between two devices, impairments of the packet network can dramatically affect the performance of the timing algorithm. These impairments include:

- *Packet Loss*. Since the timestamps in the PTP packets refer to a stable time reference and not to a previous packet, in case of a loss the slave can simply wait for the next packet and adjust its own clock according to the information obtained from the latter. The loss of an individual packet or even group of packets causes little effect on the clock recovery performance.
- *Packet Error*. In the standard situation of a PTP packet transmitted as UDP/IP packet, in case of bit errors or corruption those are perceived due to a bad checksum or frame check sequence value and discarded. Discarded packets are treated as lost packets, and hence as said above they cause little effect on the clock recovery performance. Even in the extremely unique possibility that one or more bit errors will pass both the CRC check on the frame check sequence value, and the UDP or IP checksum tests, the error may not be in the timestamp and hence would not affect the clock itself. In the opposite case, the timestamp can still be rejected if does not fall inside an expected range of values (too wide mismatch between two consequent values). Finally, a corrupted timestamp which does fall within the range of expected values, is typically averaged with other timestamps, smoothing down the possible effect on the timing algorithm.
- *Extended Packet Loss*. Due to network outages of period of congestions, extended periods of packet loss might take place; the timing source is lost and slave clock enters into an holdover mode. Thanks to conventional clock mechanisms, the slave should be able to not be affected by any degradation of the clock accuracy while path protection mechanisms restore the network into a working state.
- *Packet Delay*. Through a packet network the delay is usually higher than traditional synchronization network but, as can be seen from the master-slave offset computation, an extended delay has no effect on the accuracy of the clock.
- *Packet Delay Variation*. It is the main cause for degradation of performances; it affects the accuracy and stability of slave clocks when timing protocols based upon packet exchange are utilized, exactly as IEEE 1588. Packet Delay Variation (PDV) is caused both by the network elements themselves (e.g. switches or routers), the physical network layer and even the topology of the network. The principal sources of timing impairments for a typical Ethernet-based clock and switch are illustrated in Figure 4-12; they are mainly not-fixed delays and they tend to be correlated to the network

load., i.e. if the amount of traffic in the network increases, the delay variation is also likely to increase. Especially packet contention, input queuing, output queuing and asymmetry in the path taken by the Sync and Delay_Req packets play in favor of PDV.

How these impairments arise in the end-to-end system components is presented in Figure 4-12. They can be due to the Operating System, the Physical Layer, the Network Cable and eventual switching capabilities present between the two end nodes. In case of a switch, the Physical Layer as to be passed through two additional times per switch.

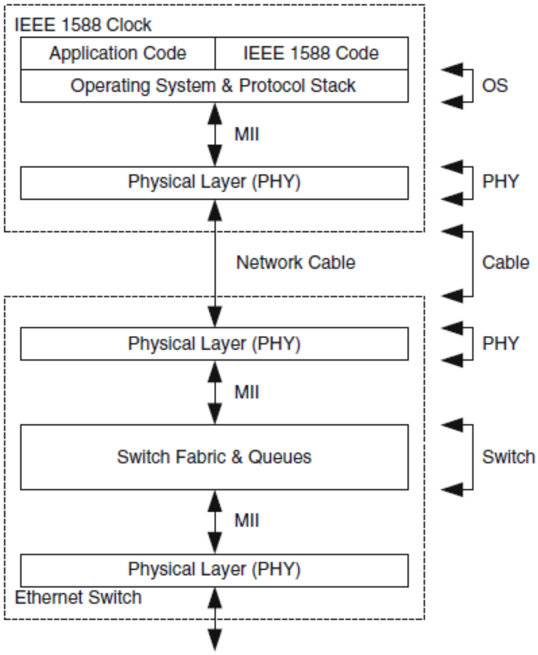


Figure 4-12. Timing impairments introduced by the end-to-end system components

4.9 Delay impairments over an Ethernet-based packet network

If in the previous Section an overview over the issues affecting the performance of the timing algorithm has been provided, in this section we focus our attention on a Ethernet packet network. The values provided in this Section are of valuable interest when the traditional synchronization process will be compared with our proposed scheme and structure (Chapter 9).

In Table 4-4 below approximate values for the time impairments presented in the previous Section and resumed in Figure 4-12 are provided. They report the delay, fluctuations in delay and asymmetry caused by the system components into a end-to-end packet exchange over a Ethernet network. The switch has to be considered as a commercially available Ethernet switch, Operating System (OS) refers to the OS of the node generating the packet. The Physical Layer (PHY) and the Cable rows refer to a 10/100BASE-T Ethernet system. The same values can be transferred to a 1000BASE-T or 10GBASE-T Ethernet systems.

Quantity	Delay	Fluctuations	Asymmetry
OS	0.1 - 3,000 μsec	0.1 – 3,000 μsec	< 3 msec
PHY	< 50 nsec	< 1 nsec	< 100 nsec
Cable	3 nsec/m	<< 1 nsec	< 0.5 nsec/m
Switch	0.4-3,000 μsec	0.1 – 3,000 μsec	< 3 msec

Table 4-4. Timing impairments introduced by the components of an Ethernet-based system.

With the term fluctuations is meant within each range the delay inflicted by the component can vary between two packets passing through. With asymmetry is meant the variation in inflicted delay between the two directions of passing through the component.
In the light of the discussion carried out in the previous Section, it can be seen in Table 4-4 how the switch is the main cause of Packet Delay Variation. In fact the variable delay inflicted by the OS can be avoided by means of an hardware for timestamp generation close to the Physical Layer, as seen in Section 4.4. The value provided here gives an idea to the reader of the contribution of the hardware solution to the final accuracy of the timing algorithm. A fluctuation of up to 3 msec in the sending node and of 3 msec in the receiving node is avoided.
With regard to the asymmetry of the system components, it has also to be pointed out that the possible variations in delay according to which direction the packet is passing through the component are of different nature. Into the PHY, they are primarily due to operation of phase lock loops for recovering of the received clock signal. In cables asymmetry is voluntarily added to reduce cross-talk effect. In the switch and the OS, it is referable to the traffic load or the workload on the processor respectively; which are the same reasons behind the fluctuations in delay.

4.10 Consequence of master-slave asymmetry upon the accuracy of the offset computation

In Section 4.8 we have listed the main issues that show up into a packet network and how they affect the performance of the algorithm for time synchronization. It has been highlighted how Packet Delay Variation is the most critical time impairment. The explanation of this is behind the hypothesis assumed by the IEEE 1588v2 protocol for the offset calculation, that is:

$$sm_{delay} = ms_{delay} = delay_{one-way}$$

Which means that the slave-to-master delay is assumed equal to the master-to-slave delay, and they are both substituted by a one-way delay (Subsection 4.5.3). Among the issues listed above, the PDV is the one which threatens the most the consistency of this assumption. In fact, assumed that the Sync and Delay_Req packets follow the same path in the two directions, it is due to PDV that the two packets might experience two different end-to-end delay. We propose hereafter an example to see which magnitude of inaccuracy is caused by different delays along the two paths. We define the packet delay asymmetry A as:

$$A = ms_{delay} - sm_{delay}$$

where ms_{dealy} is the delay experienced by the Sync message travelling from master to slave and sm_{delay} is the slave-to-master delay of the Delay_Req message.
Let suppose for our example to have a Master and Slave clock suffering of an offset of 1 hour. Let us remember that, according to the formulas presented in Subsection 4.5.3, the offset is calculated as the difference between the time beaten into the Slave Clock and the time beaten into the Master Clock. Therefore an offset of an hour means that the Slave Clock is ahead the Master Clock with a time gap of an hour. Let suppose the Master Clock is set on 1:00 AM, while the Slave Clock on 2:00 AM. We now suppose a packet delay asymmetry of 10 minutes, due to a master-to-slave propagation time of 30 minutes compared to a slave-to-master end-to-end delay of 40 minutes. The

example is resumed in Figure 4-13.

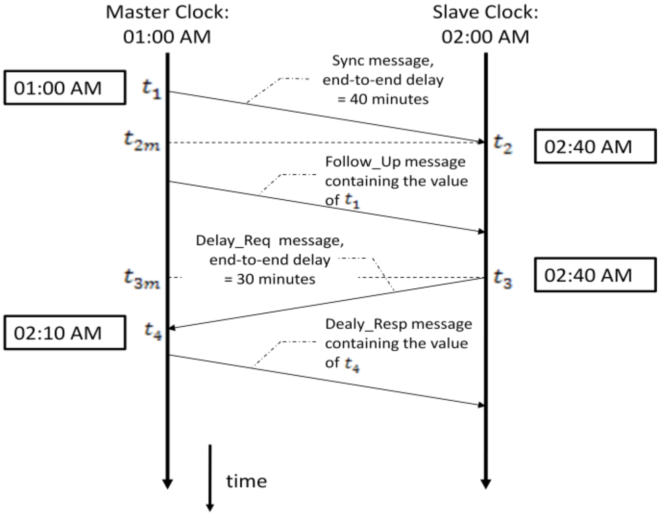


Figure 4-13. An example of a Master Clock and a Slave Clock carrying out a 4-message exchange for the offset calculation. The Slave Clock has an initial offset of 60 minutes (i.e. 1 hour), the Sync Message experience an end-to-end delay of 40 minutes while the Delay_Req message an end-to-end delay of 30 minutes.

In this example the Sync message and the Delay_Req message suffer a packet delay asymmetry A of 10 minutes:

ms_{delay} = 40 min

sm_{delay} = 30 min

A = $ms_{delay} - sm_{delay}$ = 10 min

With the values obtained from the hypothetical 4-message exchange, we can carry out the offset computation as follows:

$$\begin{cases} ms_{difference} = t_2 - t_1 = 02{:}40 - 01{:}00 = 100 \text{ min} \\ sm_{difference} = t_4 - t_3 = 02{:}10 - 02{:}40 = -30 \text{ min} \\ offset = \dfrac{(ms_{difference} - sm_{difference})}{2} = \dfrac{100-(-30)}{2} = 65 \text{ min} \end{cases}$$

The computation gives as result a value that differs from the real offset by 5 minutes, which is half of the packet delay asymmetry A=10 min.

This come straightforward from the formulas and can be generalized to every value of packet delay asymmetry. In fact, before substituting the ms_{delay} and sm_{delay} with a unique $delay_{one-way}$, the offset is calculated as:

$$offset = \frac{(ms_{difference} - sm_{difference}) - (ms_{delay} - sm_{delay})}{2}$$

which after the assumption $sm_{delay} = ms_{delay} = delay_{one-way}$ becomes:

$$offset = \frac{(ms_{difference} - sm_{difference})}{2}$$

If, as we said, the assumption is not valid due to a packet delay asymmetry A = $ms_{delay} - sm_{delay}$, then we have:

$$offset = \frac{(ms_{difference} - sm_{difference}) - A}{2} = \frac{(ms_{difference} - sm_{difference})}{2} - \frac{A}{2}$$

To sum up, the asymmetry between master-to-slave and slave-to-master delay introduces an error θ equal to:

$$\theta = \overline{offset} - offset = \frac{A}{2}$$

where $\overline{offset}$ indicates the calculated value and $offset$ the real value.

In order to avoid that the error θ could prevent the PTP to achieve a high level of synchronization accuracy, the calculated offset undergoes to filtering and averaging algorithms. Their utilization is stated into the body of the protocol (Appendix A, Clause 7.d.2) but they are outside the scope of the standard. The filtering operations are necessary to exclude incoming timestamps that are out of an expected range of values. Averaging algorithms must take engineering trade-offs between the number of samples and the responsiveness to effects other than delay fluctuations, such as oscillator stability.

The filtering and averaging algorithm adopted in commercially available IEEE 1588v2 systems are usually really complex and, as stated above, not standardized. Here in this thesis, in order to quantify the magnitude of the performance degradation as a consequence of asymmetry into the master-slave end-to-end delay, we can suppose to adopt a simple averaging function. We hereafter take into consideration the simple moving average (SMA) and the exponential moving average (EMA) functions. As mentioned, it is not likely that such simple averaging algorithms are used in a real system, but they provide a platform for weighting the effect of time impairments upon the offset calculation.

The SMA provides an unweighted mean of the previous n calculated offsets. If we indicate with A(j) the packet delay asymmetry experienced by the j^{th} 4-message exchange of timestamps:

$$A(j) = ms_{delay}(j) - sm_{delay}(j)\ ,$$

then it will cause an error in the calculated offset equal to:

$$\theta(j) = \frac{A(j)}{2}.$$

The SMA will smooth the calculated offset by averaging it together with the last n sample.

$$SMA(j) = \frac{offset_j + offset_{j-1} + \cdots + offset_{j-n+1}}{n}$$

It means that the newly calculated offset with its error θ(j) will weigh upon the inaccuracy of the timing algorithm with a magnitude equal to $\frac{\theta(j)}{n} = \frac{A(j)}{2n}$. On the other hand it has to be considered that with the addition of the j^{th} 4-message exchange of timestamps, the inaccuracy caused by the $(j-n)^{th}$ exchange is excluded by the computation. Summing up, with the utilization of the SMA function, a packet delay asymmetry equal to A(j) introduces into the offset calculation an inaccuracy of $\frac{\theta(j)}{n} = \frac{A(j)}{2n}$ and a difference in accuracy if compared with the previously calculated offset expressed by the quantity:

$$\theta_{SMA(j)} = \frac{\theta(j)}{n} - \frac{\theta(j-n)}{n} = \frac{A(j) - A(j-n)}{2n}$$

Hence, if we assume a packet delay asymmetry in the order of one millisecond and a number n=10 of samples, it will affect the accuracy by a quantity in the order of the several tens of microseconds (around 50). Increasing the number of samples to n=100, a slower response time to oscillations will be caused but the error is reduced to few microseconds (around 5).

We are going now to explore the case of adopting the EMA function. If we opt for this second option, the newly calculated offset has more weight with respect to the older samples. Into the EMA function, a weighting factor which decrease exponentially is applied to the calculated values. By means of the weighting factor, more importance is given by recent observations (i.e. a better response time to oscillations into the clocks) and at the same time older samples are not discarded entirely. The EMA function is expressed by the following formula:

$$EMA(j) = \alpha \cdot\ offset_j + (1-\alpha) \cdot offset_{j-1}$$

where α is called the smoothing factor. It is a rational number between 0 and 1 and it expresses the degree of weighting decrease. By acting on the smoothing factor, the system can be set up for faster or slower response time. Substituting the calculated offset present into the formula above with the sum of the exact offset and the error θ due to packet delay asymmetry A(j), we obtain:

$$EMA(j) = \alpha \cdot \left[offset_j + \theta(j)\right] + (1-\alpha) \cdot offset_{j-1} \;.$$

It thus it can be seen how the error θ(j) causes and inaccuracy equal to

$$\theta_{EMA(j)} = \alpha \cdot \theta(j) = \; \alpha \cdot \frac{A(j)}{2} \,.$$

This error $\theta_{EMA(j)}$ will be carried along in the EMA calculated subsequently, but the with exponentially reduced importance. In fact we can rewrite the formula defined for the EMA function showing how the calculated average steps forward the latest received sample.

$$EMA(j) = EMA(j-1) + \alpha \cdot (offset_j - EMA(j-1) \;.$$

It can be seen how the newly calculated EMA steps forward from the latest one calculated only by a portion of the difference. If we go further and expand out the formula just provided, we can see how the weighting factor decreases exponentially with the time:

$$EMA(j) = \alpha \cdot \left[offset_j + (1-\alpha) \cdot offset_{j-1} + (1-\alpha)^2 offset_{j-2} + (1-\alpha)^3 offset_{j-3} + \cdots\right]$$

As a straightforward consequence of the last formula, also the effect of the error $\theta(j)$ decreases exponentially. Its effects on the calculated EMA and how they decrease with time are dependent on the smoothing coefficient α. If we assume α=0.1, a packet delay variation in the order of a millisecond will cause an error into the calculated EMA in the order of hundreds of microseconds (around 500). Such a small smoothing coefficient however translates also into more enduring effects on the EMAs calculated afterwards. On the other hand, a higher α implies faster response time to sharp oscillations in the clocks to but less lasting effects.

Chapter 5

A more efficient topology for mobile backhauls: the cluster-type network topology

As mentioned in Chapter 2, the 3G mobile technology is currently the prevalent reality among the wireless service provider but the rollout of the successor 4G base stations has already started: on December 14th, 2009 the Scandinavian telecom operator TeliaSonera claims to be the first operator of the world launching 4G services [26]. The incoming new technology will not only put heavier requirements on the capacity of the links, but also a new structure for the backhaul (or RAN, Radio Access Network, equivalent) is suggested in order to have better load balancing into the RAN. The leading factors for the need of a more performing architecture are a higher number of Base Stations and a higher handover frequency. They are not caused exclusively by the dramatic growth of mobile traffic, but also by the expected lower cell radius of 4G systems. In fact 4G systems will offer a bit rate over the air up to ten-fold higher than the 3G technology [27]: the cell radius covered by a RAN BS generally decreases if, assuming all other conditions are the same, radio signals are transmitted at higher bit rates. This is due to the noise level; the received signal level must be higher than that at a lower transmission bit rate to compensate the increased noise component [28]. A lower radius cell translates into a higher handover frequency, which requires a more frequent BS-to-NC and NC-to-BS communication.

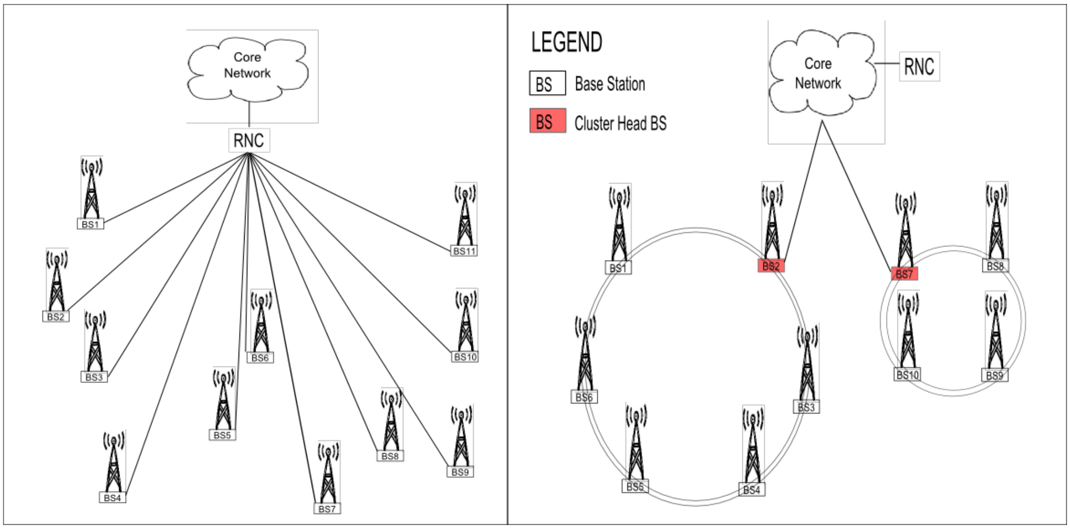
It is thus expected that the 4G RAN will transfer 23-fold more traffic than the current RANs [29]. A topology that meets the requirements of the 4G technology would imply a better load balancing among the links and a saving in costs for operators.

We are going to suggest in this Chapter a network topology that is more efficient with regard to reliability, link cost parameter and handling of diversity handover. In Section 5.1 the factors calling for an improved physical layer configuration are presented. The cluster-type topology is also introduced. In Section 5.2 is presented how, by means of this improved physical layer configuration, diversity handover is better handled. In Section 5.3 the focus shifts upon higher reliability and better link cost parameter provided by the cluster-type topology. Therefore, after having motivated the adoption of this architecture in our model of mobile backhaul, in Section 5.4 the OpMiGua HPTS scheme is introduced. In particular, it is dealt with the number of time-slots necessary in order to reach full mesh connectivity into a ring network, that is the fundamental element of the cluster-type network topology.

After this Chapter, all the necessary information in order to introduce properly our proposal have been given.

5.1 The factors calling for an improved physical layer configuration: the cluster-type topology

If with 3G technology a vertical-tree structure might be sufficient to provide a 2 Mbit/sec bandwidth to every subscriber, when the degree of traffic increase is in the order of 23-fold a new and more efficient physical link configuration might be advisable. In Figure 5-1.a the star-type (or radial) configuration of a 3G RAN is showed, while in Figure 5-1.b a suggested new topology is presented. The latter is presented in [30] and brings some enhancements both in terms of reliability and cost parameter and in terms of handling handover between two RAN BSs of fourth generation (i.e. diversity handover). Hereafter the new network topology is presented; then in the next paragraphs the diversity handover technique is introduced to the reader and how the new model for backhauls structure brings improvements upon its handling. Finally, the last paragraph of this section reports quantitative estimations of reliability and cost parameter of the suggested model in comparison with those of the radial topology.



(a) (b)

Figure 5-1. Two different network topologies: star-type topology (a) and cluster-type topology (b).

The new configuration presented in Figure 5-1.b is called cluster-type [30]: RAN BSs are grouped into clusters and organized into rings, a cluster-head RAN BS is present, which is connected to the RAN NC through the packet-based Core Network (CN). The position of the RAN NC is arbitrary inside the Core Network, but a good network planning is needed in order to not degrade the backhaul performances; communication between the RAN NC and the RAN BSs is needed for handover, control and management issues.

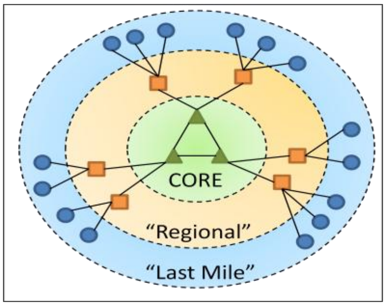
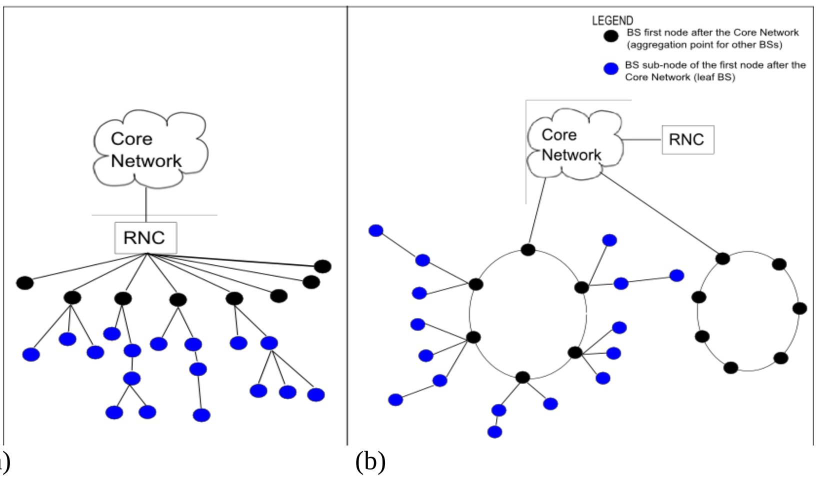


Figure 5-2. Schematic hierarchical architecture for current tree-structure mobile backhauls

It has to be pointed out that the two configurations illustrated in Figure 5-1 are a simplified scheme, the topologies stop at the first level after the Core Network. In a nutshell, from a network hierarchy perspective the mobile network can be broadly divided into three parts, as illustrated in Figure 5-2: last mile, regional network and core network. The backhaul network covers the last mile and regional network; it connects the elements of these two parts with the main elements (i.e. RAN NCs) present into the core network. If we refer to this schematic hierarchical architecture, we would say that in Figure 5-2 only the core network and the regional backhaul is considered. In a mobile backhaul scenario, often the *leaf* BSs (i.e. referring to the tree topology, the BS included into the last mile part) are connected towards the Core Network by point-to-point microwave bridges with a closer BS. This is what is called the microwave backhaul [31]. The factors influencing the presence of point-to-point microwave bridges over the air inside the mobile backhaul are numerous. In a first place, microwave can be deployed rapidly and cost-effectively; excavations are not needed for cable implantations which enables services to be established quickly in a competitive environment. Utilization of microwaves implies important drawbacks as well [24]: reduced capacity during severe weather conditions, higher operational expenditures (OPEX), utilization of a valuable and expensive resource, that is the licensed spectrum. Due to the listed drawbacks, in a scenario like dense urban settings where leased lines have an extended availability and have reached high degree of penetration, they are the preferred solution by mobile operators. Here the pre-existing excavations (e.g. current, gas, aqueduct, …) and the availability of leased lines lower down appreciably the CAPEX (capital expenditure) as well as the time necessary for making the services available.

In considering which topology is better suitable as mobile backhaul, we stop at the regional part of the backhaul; the RAN BSs illustrated into Figure 5-1 have to be actually considered as aggregation points. Below those BSs, a reduced-size radial or tree topology is likely to be present. It is a valuable contribution to propose modifications and improvements upon the RAN physical configuration at the regional level; transmission distances, terrain characteristics and base sites density are the main factors dictating the technology and the topology to be used into the last mile part [32]. The business case is defined by those factors and, even if a different topology might bring enhancements with regards to traffic load and signal processing at the network elements, the costs to adopt a different structure can represent a barrier too high from the operator's point of view. On the other hand the regional part is likely to coincide with urban areas where, as said above, the increasing spread of fiber infrastructures and their degree of penetration makes easier to introduce a new network topology. It is for this set of reasons that we decide to disregard the last mile part of the backhaul and focus on the regional and core sections.

In conclusion, in Figure 5-3 are illustrated the architectures which have to be born in mind, both the already rolled-out radial structure (Figure 5-3.a) and the suggested cluster-type topology (Figure 5-3.b). For the leaf RAN BSs belonging to the last mile, not a unique assumption can be done.



(a) (b)

Figure 5-3. The real-like scenario with the two different network topologies: radial topology (a)and cluster-type topology (b) with nodes as aggregation points, having underneath reduced-size radial or tree topology of 'leaf' BSs

5.2 Cluster-type network topology for efficient diversity handover

As already stated, the proposal of a new structure for the Radio Access Network is called to counter-act the higher number of Base Stations and the more frequent handover from BS to BS. The load of traffic that handover causes over the network is made a bigger burden by the introduction, already with the 3G technology, of the Macro Diversity Handover (MDHO) [33].
Handover, or handoff equivalently, is generally performed when the quality of the link (measured in terms of the power of the received signal) between the RAN BS and the Mobil Terminal (MT) on the move is decreasing and it is possible to hand over the connection to another cell with better radio characteristics. In previous 2G systems like GMS, the hard handover process was applied: when moving from one cell to another, the existing connection was torn down and replaced with a new connection to the new cell. The user was handed over to the so-called “target” cell with a different frequency. Thus, in hard handover technique, the connection was interrupted for a short period of time not noticeable by the end user. As a result, the MT communicates with only one RAN BS in a time.
A new technique known under the name of soft handover was introduced in 3G systems [34]. It is based on the MT and the RAN BS maintaining a “diversity set”, from which the name Macro Diversity Handover. Diversity set is defined for each of the MTs in the network, a MT communicates with all the RAN BSs in its diversity set. Thus the diversity handover method allows a Mobile Terminal (MT) to communicate with plurality of RAN BSs simultaneously in order to enhance radio-signal quality at the cell edges: the quality of the radio signal between MT and RAN BS is established by means of small feedbacks in the MT -> RAN BS direction. When the MT had to be handed off form one cell to another, the new connection it setup before the existing one is torn down. How this new technique affects the traffic load over a network with radial topology or cluster-type topology is shown schematically in Figure 5-4.

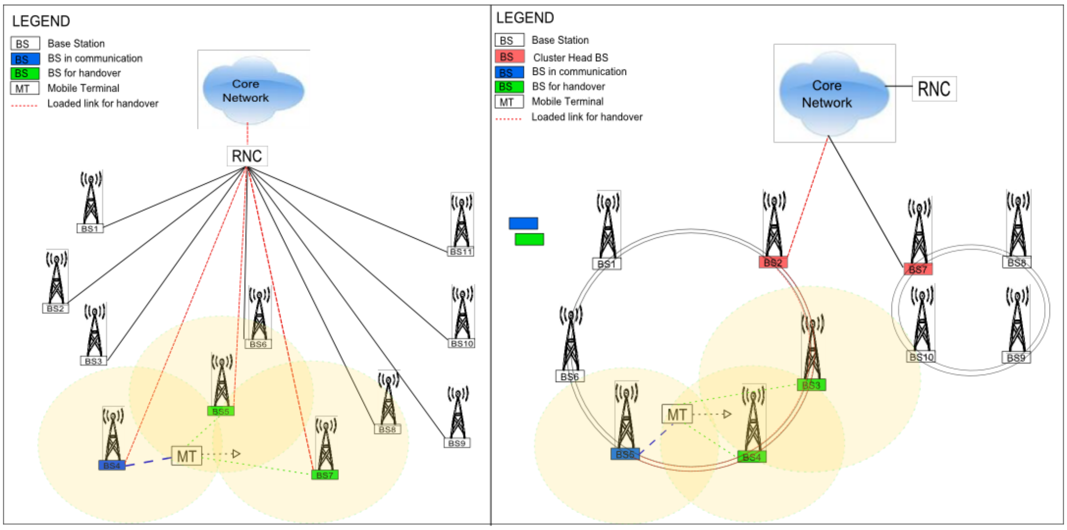


Figure 5-4. Diversity handover processing in a radial network topology (a) and in a cluster-type network topology (b).

In the 3G topology illustrated in Figure 5-4.a, when a MT can communicate with multiple BSs, all signals of the uplink (MT to BSs) received at each BS are transmitted to the RNC which combines them in a user data stream; similarly, signals of the downlink (BSs to MT) are transmitted from the multiple BSs and combined into a user data stream from the MT. As an example, if the MT is in the coverage area of three BSs simultaneously, the bandwidth utilization on the links between the RNC and the BSs are tripled compared to the original user information. In the cluster-type topology illustrated in Figure 5-4.b, diversity handover can be processed in a distributed manner within the cluster (localized handover processing) with considerable sparing of resource utilization on the links; a temporal agent related to the MT (the one with better feedback about radio-signal quality from the MT) can be elected as responsible for combining of uplink and downlink user data stream, saving load on the link between cluster-head BS and RNC.

5.3 Higher reliability and better link cost parameter

If in the previous paragraph it has been showed how diversity handover is better processed with a cluster-type network topology, hereafter the focus shifts to better reliability and cost parameter of the introduced topology in comparison with the radial structure. We will briefly provide some quantitative estimations; a more detailed analysis can be found in [35]. The radial structure and the cluster-type structure are compared leading to the results in Figure 5-5 and Figure 5-6.

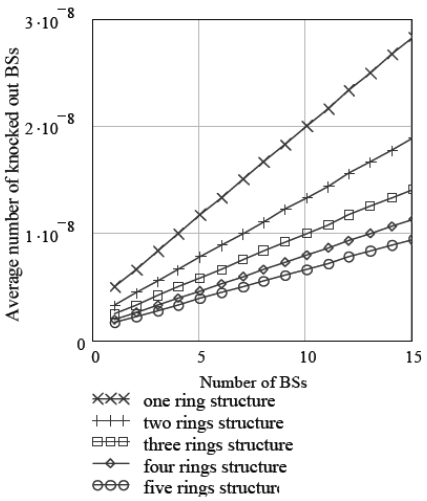


Figure 5-5. Reliability estimation of a cluster-type network. The average number of knocked out Bss (i.e. not able to communicate with the rest of the network) is illustrated depending on the number of rings in the RAN and the number of BSs in the ring. A probability p_0 of zero damage in the link equals to 0.9999 is assumed.

In Figure 5-5 a reliability estimation of the cluster-type structure is presented; it is illustrated the average number of knocked out BSs depending on the number of rings in the RAN and the number of BSs per the ring, where a BS has to be considered knocked out when it cannot serve its mobile terminals because of damages in the RAN physical links (fault on the backhaul). In the figure, a probability p_0 of zero damage on the link (i.e. the availability of the link in term of the reliability theory) equals to 0.9999 is considered. The graph is extrapolated from the following formula, we report it for sake of completion but its derivation can be found in [35]:

$$X(N,M,L) = \frac{(M+2)(-\ln p_0)^2 p_0}{3(N+1)} + \frac{(M+2)(2N+1)(-\ln p_0)^3\, p_0}{6(N+1)(N+2)}$$

where:

- $X(N,M,L) \epsilon \{1,2,\dots,z\}$: average number of knocked out BSs
- z is the number of BSs in the RAN, $z \epsilon \{1,2,\dots\}$
- N is the number of rings in the RAN, $N \epsilon \{1,2,\dots\}$

- L is the number of damages in the RAN, $L\epsilon\{0,1,2,\dots\}$
- M is the number of BSs in a ring, $M\epsilon\{1,2,\dots\}$. The number of BSs in each ring is equal.

The probability of damages appearing in the RAN physical links is assumed as Poisson process

$$p_L(T) = (\lambda T)^L \frac{e^{-\lambda T}}{L!}$$

where $p_L(T)$ is the probability of L damages appearing in the RAN for a given interval time T, λ is the flow parameter of damages.

It is seen in the graph that, given the ring topology as basic constitutive element, increasing the rings number yields to a considerable rise-up in RAN. Thus the average number of knocked out BSs with a link availability p_0 of 99.99%:

- equals $1 * 10^{-8}$ if one ring is present and 4 BSs in the ring
- equals $1 * 10^{-8}$ if two rings are present and 7 BSs per ring
- equals $1 * 10^{-8}$ if three rings are present and 10 BSs per ring
- equals $1 * 10^{-8}$ if four rings are present and 13 BSs per ring
- equals $1 * 10^{-8}$ if five rings are present and 15 BSs per ring

In order to give a basis for comparison, it is valuable to report the formula to calculate the average number of knocked out BSs for the radial structure:

$$X(M,L) = -(\ln p_0)p_0 + (-\ln p_0)^2 p_0 + \frac{(-\ln p_0)^3\, p_0}{2}$$

If we assume a link availability p_0 of 0.9999 as above, then the average number of knocked out BSs for a radial structure equals 10^{-4}, about four orders greater than the cluster-type structure. This value is also independent from the number of BSs present into the RAN; this can be seen in the formula and it is due to the fact that a single failure on the link between BS and RNC causes the BS to be knocked out, while in the cluster-type topology a BS has to be considered knocked out if failures occur on both links at the two sides of the BS.

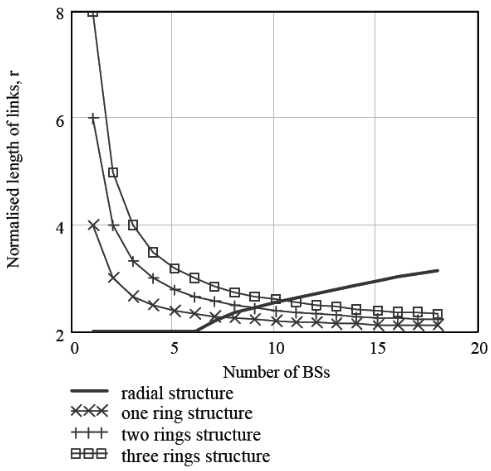


Figure 5-6. Comparison of cost estimation for a cluster-type and a radial network. With regard to the cluster-type topology, normalized length of links against the rings number and the number of BSs per ring is illustrated.

Concerning the cost estimation, Figure 5-6 shows the normalized length of links against the ring number and the number of BSs per ring in the backhaul. It is seen that if the number of BSs in the RAN is from 1 to 7 the cost parameter of the radial structure is more beneficial than the ring structure ones; if the number of BSs is 7 or more, then the ring structures are preferable to radial from the viewpoint of costs.

From the quantitative results reported above, together with the considerations upon handling of diversity handover from the two network architectures, it looks clear how the cluster-type topology seems to be more convenient for mobile operators. It provides a higher reliability together with a better link cost parameter. Link cost parameter becomes an important factor since the regional part of the backhaul, the one that the cluster-type topology is proposed for, is likely to fall into urban areas, as stated above. Here leased lines are preferable to microwave backhauls and hence the length of the links impacts on the economy of the operators. This, together with a better load balancing of traffic upon the links for handling of diversity handover, can make the cluster-type topology more interesting at the eyes of the mobile operators. We thus find convenient to propose exploitation of the OpMiGua concept (and in particular its HPTS scheme) over this new proposal of RAN physical layer configuration: OpMiGua installation for a mobile operator must be a long-term solution, thus being suitable for future scenarios has to be considered as an important added value.

5.4 On the number of time-slots for full mesh connectivity in a ring network topology exploiting OpMiGua HPTS scheme

On the light of the considerations and quantitative results shown in the previous section, the hypothesis to insert the OpMiGua HPTS scheme into a mobile backhaul presenting a cluster-type topology has been adopted. Since the ring topology with employment of time-slots is going to be the basic constitutive element, hereafter we are going to analyze the number of time-slots needed in order to reach full connectivity within the ring.

In order to derivate the solution of the problem, its analogy with a well-studied case can be exploited: the number of wavelengths needed in order to reach full WRON mesh connectivity in a unprotected bidirectional ring network (Figure 5-7).

As already reported in Section 3.1, a WRON (Wavelength Routed Optical Network) is an optical network where the traffic is transported transparently from end to end forwarding packets according to the wavelength which is carrying them. The Color Clash constraint for optical signals belonging to different channels (addressed to different end nodes) has to be observed in a WRON. It states that optical signals simultaneously sharing a single fiber are required to have different wavelengths [38]. The channel uniqueness stated by the Color Clash constraint is equivalent to the following valid for OpMiGua HPTS scheme: it is not allowed to set up more than a single GST Label Switched Path within each time-slot in the OpMiGua HPTS scheme [10].

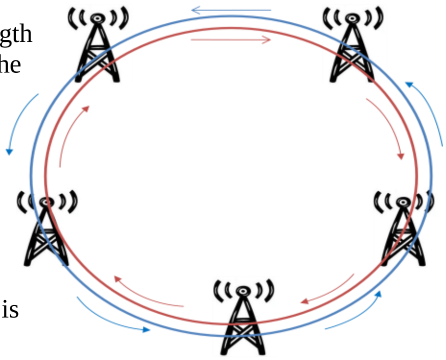


Figure 5-7. Unprotected bidirectional ring network, one link per transmission direction

In conclusion, deriving the minimum number of wavelengths needed in order to assure that the WRON ring is fully connected has a problem formulation and problem constraints that are equivalent to those of our problem; hence the two problems can be considered equivalent.

The analogy between the two cases can be observed in Figure 5-8.

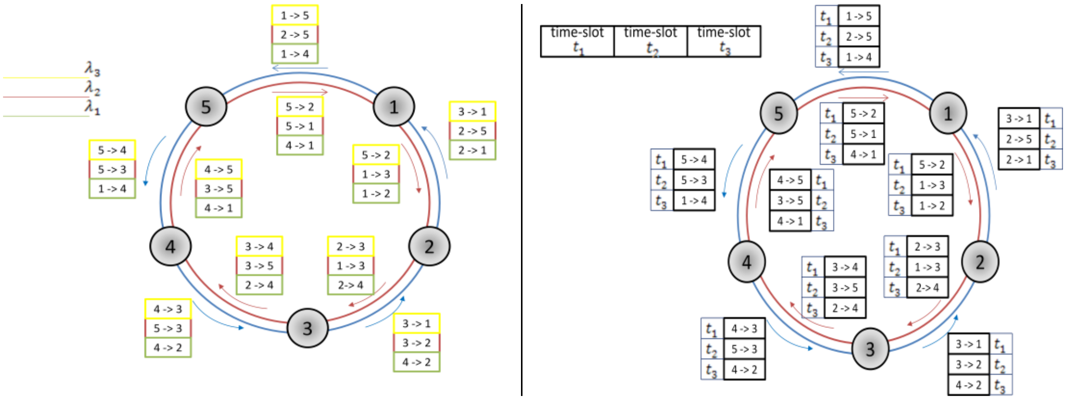


Figure 5-8. Analogy between (a) number of wavelengths needed for full mesh connectivity into a WRON optical ring and (b) number of time-slots needed for full mesh connectivity into a OpMiGua ring with HPTS scheme exploitation.

Since the issue regarding the number of wavelengths necessary in a WRON ring to reach full connectivity has been already studied, the same problem solving process can be utilized for our equivalent problem. The minimum number of wavelengths can be translated into the number of time-slots needed in the ring OpMiGua network with exploitation of HPTS transmission scheme.
As derived in [37], for achieving full WRON mesh connectivity in an unprotected bidirectional ring with x nodes approximately $W \approx x^2/8$ wavelengths are required. Nominally:

- For an odd number of nodes on the ring

$$W_{odd} = \frac{x^2 - 1}{8}$$

- For an even number of nodes on the ring
  - in case the number of nodes N is divisible by 4:

$$W_{even,e} = \frac{x^2}{8}$$

  - in case the number of nodes N is divisible by 2 but not by 4:

$$W_{even,o} = \frac{x^2 + 4}{8}$$

In case all the nodes belonging to the network are capable of wavelength conversion, this is referred to as full wavelength conversion. Into a ring network with full wavelength conversion, the values reported above do not represent only the minimum number of wavelengths needed but actually the real value of wavelengths required [37]. If the nodes are not wavelength conversion capable, the number of necessary wavelengths is not depending uniquely on the number of nodes but also on the efficiency of the algorithm for their assignment [38].
A solution that places itself in between is the sparse wavelength conversion. A network is defined capable of sparse wavelength conversion when only part of the network nodes have the capability of wavelength conversion, while others have no conversion capability [39]. In this case, a further factor impacts on the number of wavelengths needed to reach full WRON connectivity: the wavelength converters placement. Some studies have been carried out also upon this particular issue; a number of heuristic algorithms have been proposed both for some realistic network topologies [40] [41] [42] and for the special case of the ring topology [43].
As argued at the beginning of this section, the equivalence between the number of wavelengths into a WRON network and the number of time-slots into a OpMiGua HPTS network allows us to translates the values obtained for the former into the equivalent searched values for the latter. It comes out as a result that the minimum number TS of time-slots needed in order to reach full mesh connectivity in an unprotected ring with N nodes when a single channel (i.e. wavelength) is available is as follows:

- For an odd number of nodes on the ring

$$TS_{odd} = \frac{N^2 - 1}{8}$$

- For an even number of nodes on the ring
    - in case the number of nodes N is divisible by 4:

$$TS_{even,e} = \frac{N^2}{8}$$

    - in case the number of nodes N is divisible by 2 but not by 4:

$$TS_{even,o} = \frac{N^2 + 4}{8}$$

Above has been said, upon the number of wavelengths necessary for full mesh connectivity into a ring WRON, that the reported values indicate the minimum number of wavelength; the actual number of them that needed is influenced by the wavelength assignment algorithm, whether the nodes are wavelength conversion capable or not and the placement of those that are capable. Equivalently, the values TS_{odd}, $TS_{even,e}$ and $TS_{even,o}$ here reported represent the minimum number of time-slots needed. The capability of the OpMiGua nodes to convert the time-slot of a GST packet passing through impacts on the number of time-slots necessary. Analogously to the multi-wavelength problem, the OpMiGua network exploiting the HTPS scheme can be referred to as full time-slot conversion, in case all the nodes are time-slot conversion capable, or as parse time-slot conversion, in case part of the nodes are time-slot conversion capable and part are not. The capability of time-slot conversion is easily implementable into an electrical domain, for instance widely used in SDH/SONET nodes. Into the optical domain it does not come straightforward since a variable optical delay is implied, although the delay can be expressed in steps of fixed amounts (step increments).

In conclusion, the case of full time-slot conversion implies that the values reported above do not represent only the minimum number of time-slots needed but actually the real value of time-slots required. If we are in a scenario of parse time-slot conversion instead, number and placement of the time-slot conversion capable nodes are further factors. Finally, in a OpMiGua HPTS network where none of the nodes is time-slot conversion capable, the real number of time-slot requested is dependent on the time-slot assignment algorithm.

Chapter 6

Synchronization within OpMiGua-based mobile backhaul

The path followed so far as started in Chapter 2 with an overview of the mobile backhauls and how they are required to switch from a Time Division Multiplexing base to a all-packet base. In Chapter 3 the OpMiGua HPTS scheme is introduced, with its valuable hybrid feature. In Chapter 4 the Precision Time Protocol is explained, a protocol able to provide synchronization among the nodes without the TDM signal but rather thanks to message exchange. In Chapter 5 the cluster-type network topology is suggested as improved and more suitable physical layer configuration for NGNM backhauls. In this Chapter we are going to put together all the pieces of the puzzle into our proposal: a synchronization technique over a mobile backhaul in which nodes exploit the OpMiGua HPTS scheme and are organized into a cluster-type physical layer configuration.
The use case scenario which reflects the considerations carried out so far is presented in Section 6.1. In the rest of the Chapter general issues and guidelines are present. A detailed explanation is provided in the next chapter. Section 6.2 consists of general directions upon the communication between Grandmaster Clocks and the cluster-head RAN BS. It is followed, in Section 6.3, by the issue of the cluster-node hosting more than one clock information. The traditional architecture of a node hosting IEEE 1588 functions is reported; then the architecture of a node hosting IEEE 1588 functions but belonging to a plurality of subdomains is suggested. The focus shifts in Section 6.4 upon the ring, fundamental element of the cluster-type topology. By means of this specific structure and the synchronization technique we are going to propose, a more uniform and simply manageable Master-Slave hierarchy is achievable. The Chapter is concluded by Section 6.5 which provides the general guidelines for our proposal of time synchronization technique. It is called the IEEE 1588v2 HPTS scheme. A structure for the OpMiGua node is presented and a new terminology introduced by the new technique is provided. We will return to these guidelines and new terms and we will elaborate on them into the next chapters.

6.1 Use case scenario

In Chapter 2 Carrier Ethernet is presented as most promising technology for mobile service-providers. Thanks to the enhancements brought by the Metro Ethernet Forum working group, Carrier Ethernet is able to satisfy the restraining requirements of a Next Generation Mobile Network. This makes it the most cost-effective solution for the NGMN backhauls. In Chapter 3 the features of the OpMiGua HPTS scheme are shown: its capability to provide finer granularity and higher number of circuits over the same GST path, in comparison with the original OpMiGua concept, is highlighted. They are both characteristic that makes the OpMiGua HTPS scheme really attractive for service providers in a scenario such mobile backhauls where a few or only a single channel are available. In Chapter 4 a peculiar aspect of mobile backhauls is taken into consideration: synchronization among RAN BSs. When evolving from TDM-based to packet-based backhaul, the possibility to trace back a time reference from the wireline signal (i.e. physical layer) is taken away. Hence it comes to the aid of clock synchronization a packet-layer based protocol that experienced a certain degree of popularity among service providers: IEEE 1588, aka PTP, in particular its second version released in the year 2008. Finally, in Chapter 5, a more efficient topology for mobile backhauls is presented, that is the cluster-type topology. It is underlined how this physical layer configuration provides for a better handling of diversity handover, a technique already introduced with the 3G technology. Besides this important feature, the cluster-type topology also allows a higher reliability and a better link cost parameter for the network architecture. Finally, in the third part,

In the light of the study carried on so far it is, therefore, in the following use case scenario that we are going to introduce our proposal for a more efficient and less bandwidth-consumptive timing distribution technique: a cluster-type structure with nodes exploiting the OpMiGua HPTS scheme for circuit establishing and Carrier Ethernet for packet delivering. IEEE 1588v2 will be the platform for clock synchronization among the RAN BSs; our proposal relies on and is integrated with the PTP scheme.

The cluster-head RAN BS is connected to the Core Network through a unique bidirectional link (or equivalently a pair of unidirectional links); the RAN NC is situated arbitrarily inside the CN. The Core Network is packet-based, likely based upon Carrier Ethernet just as the backhaul is expected to be. The links (one per transmission direction) connecting the nodes into a ring topology are wired; it can be both a copper link (electrical domain) or an optical fiber (optical domain). Both cases will be explored, with the optical domain going to be considered a special case of the electrical domain: in fact, in optical domain State of Polarization can be used as label for GST and SM packets distinction as stated into the OpMiGua project, while in case of copper links an electrical (bits) label has to be used. Beneath the nodes belonging to the ring, a reduced-size radial or tree topology is present. As argued in Section 5.1, this structure is not convenient to be modified due to high costs of changing preexisting links into a new structure for hardly reachable nodes. These links can be of any type: optical fiber, copper wire, microwave bridge or optical wireless.

With OpMiGua HPTS scheme exploitation for circuit establishing, the Guaranteed Service Traffic (GST) circuit is divided into time-slots in order to allow higher granularity and higher number of circuits. A more proper term would be “hybrid circuit establishing” since the circuit setup does not prevent packet-switched Statistically Multiplexed (SM) traffic to exploit under-utilized resources within the circuits. Its utilization in the rings, which represent the basic constitutive element of cluster-type topology, means two trains of time-slots travelling through the nodes of each ring; one train is carried clockwise and the other counter-clockwise. The time-slots are grouped into time-frames, where each time-frame contains the number of time-slots necessary to reach full mesh connectivity within the ring. Hence the number of time-slots into each frame and the length of each time-slot will depend on the number of nodes in the ring and the vendor configuration respectively, but the general situation can be seen as illustrated in Figure 6-1.

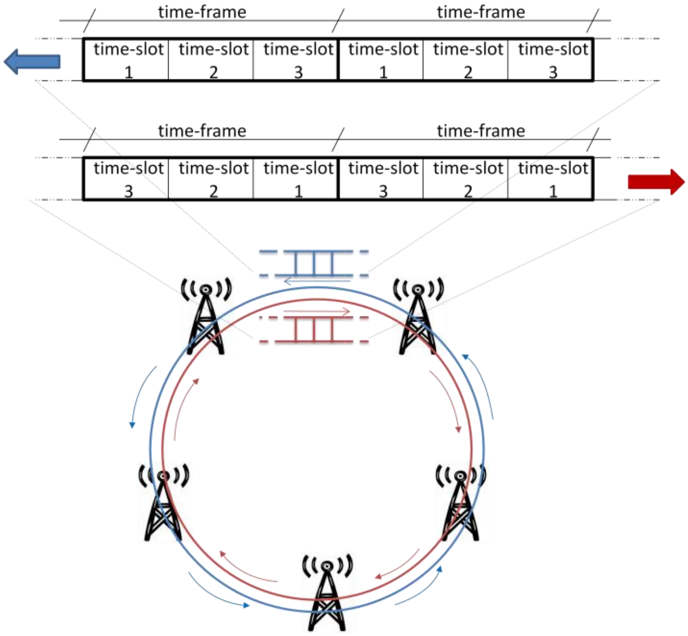


Figure 6-1. General situation of a ring structure adopting the OpMiGua HPTS scheme. Two train of time-slots circulate through the ring; one clockwise and one counter-clockwise. The time-slots are organized in time-frames.

About the network dimensioning, the following values are reasonable for a realistic scenario:

- 10 to 15 nodes (i.e. Base Stations) per ring
- 5 to 10 leaf nodes (i.e. Base Stations) per node belonging to the ring

These values will be analyzed deeper when end-to-end delay constrictions of the provided mobile services and available bandwidth per RAN BS will be taken into consideration. Especially the delay aspect impedes the aggregation of a wide number of RAN BSs into a unique ring: as stated in Section 5.4, if one only channel (i.e. wavelength in case of optical fiber or copper cable into the electrical domain)is available then the minimum number of time-slots TS increases approximately as $N^2/8$ where N is the number of nodes belonging to the ring. This means that, given a fixed duration for the time-slots, having an high number of nodes increases the time interval between two subsequent time-slots for the same destination approximately as $N^2/8$ as well. Adding too much delay at the sending node might be critical in terms of total end-to-end delay, even if circuit-switched quality is applied for delivering the packets.
The multi-operator situation is considered: different providers are located within the same ring of the cluster, every provider will refer to a different Grandmaster clock as time-base and need to communicate with a different RAN NC for OA&M (Operation, Administration and Management). The communication between the cluster-head RAN BS and the RAN NC is beyond the scope of this document, while some guidelines on the packet exchange between the Grandmaster Clock and the cluster-head RAN BS is provided in the next paragraph. Both the packet exchanges take place through the Core Network, which differently from the mobile backhaul hosts every kind of application and cannot be designed to be mobile-operators oriented; nevertheless, it is worth to give a hint on how PTP packets are carried from end to end throughout a traditional packet-based network. This can be used as base of comparison with our proposal for clock synchronization within the ring topology when OpMiGua HPTS scheme is utilized; furthermore, where the ring topology is not advisable (i.e. in the radial or

tree topology beneath the nodes belonging to the ring), traditional PTP packets will be used. A particular attention to the multivendor feature will be given; a realistic scenario is to have up to four different operators within the same ring, more likely from two to three.

6.2 Communication between Grandmaster Clocks and the cluster-head RAN BSs

PTP packets between the Grandmaster Clock and the cluster-head BS are delivered by a common packet-based network, designed with a set of options for addressing quality of service: priority management, bandwidth reservation, load balancing, traffic policing or shaping, etc. These mechanism, necessary for a well functioning of a network hosting every kind of application, on the other hand impede the router or switch from transferring the packet from one port to another at the raw maximum speed and with fixed delay. Better performances would be achievable if the various QoS features are turned off, of course not possible into a Core Network, but at least the following general guidelines should be respected:

- In case bandwidth reservation is used, it has to be ensured that sufficient bandwidth is allocated to the timing traffic
- Traffic shaping has to be avoided for PTP packets, it is rather advisable to discard the packet (i.e. traffic policing) than to add arbitrary delay; as said above, the effects of a wide Packet Delay Variation vanish the utility of a packet-based synchronization.
- In case output queuing management is used, strict mechanisms of priority have to be applied. Applying scheduling algorithms as Fair Queuing and Round Robin causes addition of arbitrary delay and, as a consequence, important contribution to the PDV parameter.

Bearing in mind whether these features are applicable or not into the network components, distribution of Grandmaster Clocks within the core network has to be designed. In dealing with distributed Grandmaster, a rule of thumb that might be convenient to apply states that is better to place a boundary clock after 5 hops than to attempt to span the entire 10 hops network; translated into our use case scenario, it means that between the Grandmaster Clock and the cluster-head RAN BS an intermediary boundary clock might be present. This does not affect the general purpose of our use case scenario: what is relevant to us is the cluster-head receiving clock information from the time bases present into the CN. A lack of accuracy into the synchronization of the cluster-head RAN BS with the Grandmaster Clock impacts on the accuracy of synchronization for the rest of the cluster-type topology.

6.3 Issue of a node hosting more than one clock information

To be the study case as close as possible to real-life scenarios, the assumption of multivendor backhaul has been adopted; this might create a conflict issue at the cluster-head RAN BS for clock synchronization. In fact, in our proposal of a new model for Next Generation Mobile Networks, a timing mechanism more suitable to the specific case of cluster-type topology with exploitation of OpMiGua HPTS scheme for traffic delivering is going to be proposed. This mechanism foresees the cluster-head RAN BS to be Master Clock of all the nodes into the ring, independently from the Grandmaster clock they are taking as a reference. Except for the case of particular agreements among the vendors, thus a particular situation that cannot be taken as granted, it is likely that the RAN BSs belonging to different vendors refer to different Grandmaster Clocks.

Each system which has a different Grandmaster Clock as time base, as a result, maintains its time scale independently form the others; these independent synchronization systems are called "**subdomains**". Subdomains are implemented by defining a namespace, so that each subdomain is distinguished by a subdomain name. All PTP messages contain the name of the applicable subdomain. All interactions, communications, and other features of IEEE 1588v2 occur within a single subdomain, and are logically independent of similar operations in other subdomains. When a clock belongs to a single domain, the architecture of the node hosting the clock is as illustrated in Figure 6-2 in case it is in a master state or as illustrated in Figure 6-3 in case it is in a slave state; these two figure refers to the transmission and reception of a Sync message (and related Follow_Up message), a similar sequence of events is involved when the Delay_Req and Delay_Resp messages occur between the slave and the master. Both figures are specific for an Ethernet-based node; similar techniques can be used for other network technologies, although some details has to refer to that specific technology (e.g. the MII interface is Ethernet-specific).

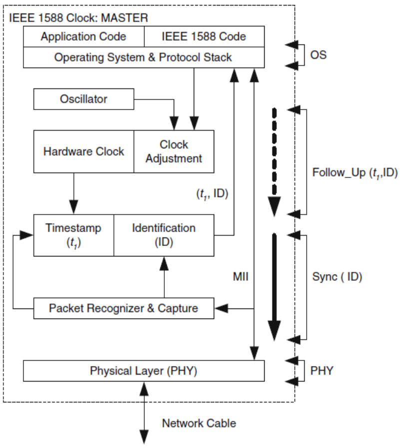


Figure 6-2. Architecture of a node hosting IEEE 1588 functions with the clock in a Master position. The transmission of a Sync and the related Follow_Up messages is shown, the reception of a Delay_Req and the related Delay_Resp transmission follow the same mechanism.

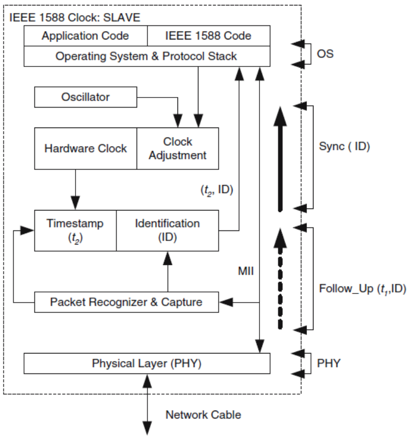


Figure 6-3. Architecture of a node hosting IEEE 1588 functions with the clock in a Slave position. The reception of a Sync and the related Follow_Up message is shown, the transmission of a Delay_Req and the related Delay_Resp reception follow the same mechanism.

The key element is the Packet Recognizer & Capture block (PRC block); it passively observes both transmitted and received packets at a point as close to the network connection as possible. In an Ethernet environment, the media-independent interface (MII) is the easiest point of access. The PRC block selectively detects all Sync and Delay_Req packets, captures a snapshot of the local hardware clock to generate the appropriate timestamp and captures the identification field from the PTP packets in order to associate the timestamp with the correct Sync, Delay_Req or Delay_Resp message.

When the node hosts a unique clock belonging to a single subdomain, the synchronization process can be carried out through the architecture illustrated above. A different and equally efficient architecture has to be utilized instead when the node belongs to different subdomains; it is the very case of the cluster-head node in the cluster-type topology. It has to receive different clock information from different Grandmaster Clock (or intermediary clocks equivalent to the Grandmaster Clocks) and to distribute this set of info into the ring independently from each others.

An architecture suitable to be exploited for managing the sharing of a single node by a multitude of subdomains is as illustrated in Figure 6-4. The clock is connected to the network through several ports instead of one, a pair of ports per subdomain is needed: one to listen to the PTP packets coming from the Grandmaster Clock through the Core Network, one to send IEEE-1588-related packets through the ring. Each port is effectively serviced by a multiplexer that separates PTP traffic from all other network traffic. Non-PTP traffic is directed to the usual switch or router functionality, while PTP traffic is directed to the block labeled "IEEE 1588 functions" that is connected to the clock and implements the PTP functions; those functions are equivalent to those illustrated in Figure 6-2 and Figure 6-3. The "IEEE 1588 functions" block maintains a separate section for each port, to allow each port to be in a different state. Generally a single clock is present into the PTP block serving all the ports, but it might be considered reasonable also the utilization of a specific clock for each subdomain. Considering the position of the cluster-head node into the network architecture, the cost of more than a single clock would be shared by a high number of RAN BSs (and hence a high number of customers). It would also decrease the computation footprint and complexity needed for timing operations, with better performances with regard of delay parameters of PTP packets. Moreover, different subdomains might also adopt different epoch and rate, as they determined by the Grandmaster clock; translating the same clock signal into different time scales (e.g. UTC, GPS, Loran-C, TAI, …) can cause inaccuracy by the block that the translation is committed to.

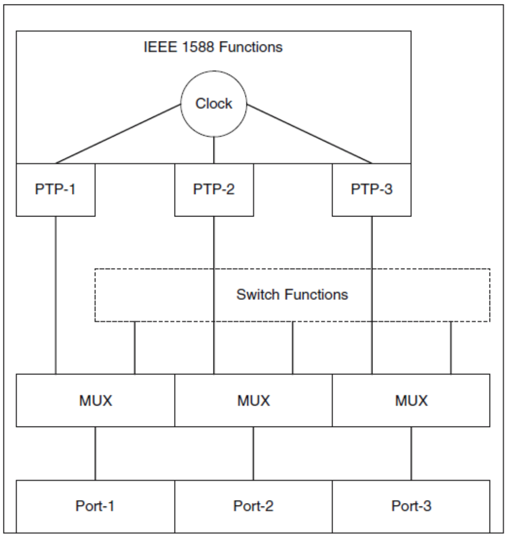


Figure 6-4. Architecture of a node hosting IEEE 1588 functions and belonging to a plurality of a subdomains. It is the very case of the cluster-head node that is Master Clock for all the nodes of the ring structure.

6.4 An improved Master-Slave hierarchy

After having analyzed the issues concerning PTP packets from Grandmaster Clocks to and into the cluster-head RAN BS, we are going now to illustrate how the cluster-type topology brings valuable advantages into the distributed timing.

With the utilization of this topology, a simple and efficient master-slave hierarchy among the clocks of the system can be established. In terms of timing distribution, a simple hierarchical structure with few levels of parent-child relationship limits considerably the cascade effect for

error propagation. When the structure presents different levels of master-to-slave hierarchy, the quantization error and the servo error (i.e. the "offset" computed from the exchanged timestamps) propagate from master to slave summing up at the every step of the cascade. Few levels of hierarchy translates into a homogeneous time base for the whole system. The different master-slave structure of the proposed cluster-type topology compared with the radial topology in use for 3G backhauls is showed in Figure 6-5.
The cluster-head node acts as first slave of the Grandmaster Clocks and, in the same time, Master Clock of all the nodes belonging to the ring. Those nodes connected in the ring topology are, in turn, masters of the reduced-size tree or radial topology underneath. With regard to the topology beneath the nodes into the ring, a radial structure is more convenient than a tree structure since the former avoids to create more sublevels in the master-slave hierarchy. If a tree structure is present, from the point of view of synchronization is better to provide a mechanism which allows the timestamps UDP/IP messages to bypass the intermediate clock than to create another master-slave between the two nodes.

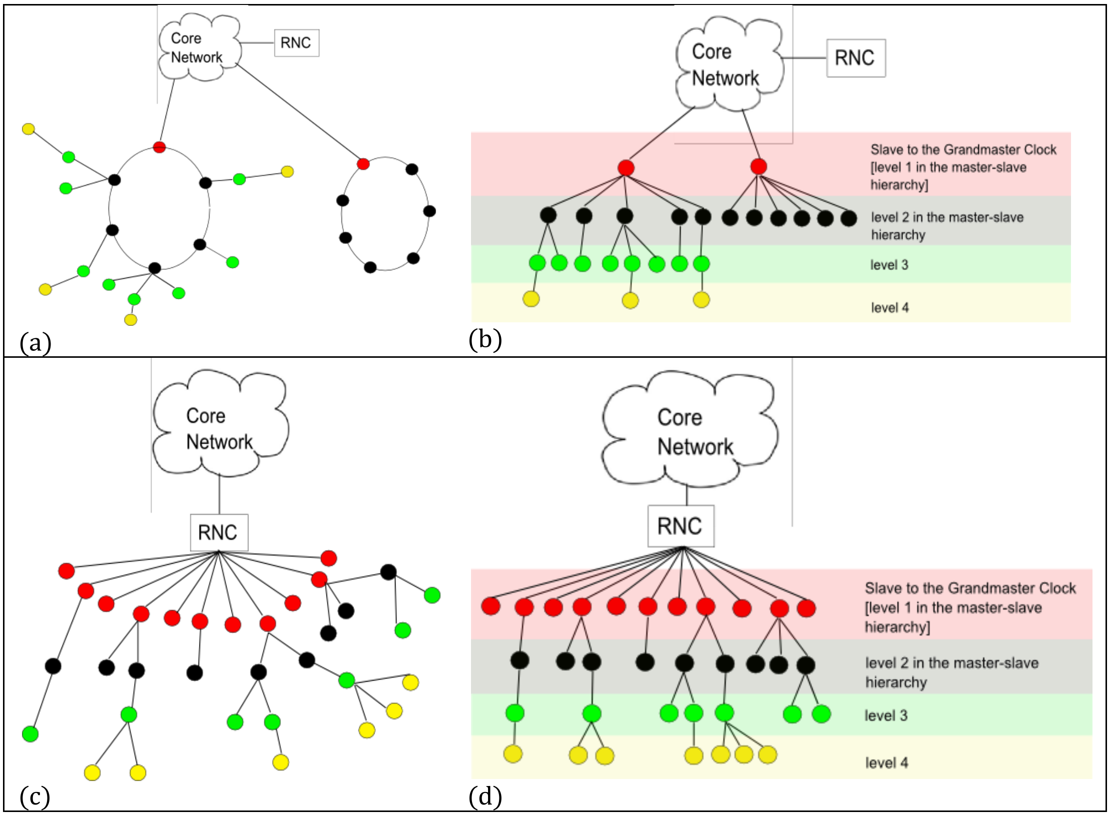


Figure 6-5. Comparison between the Master-Slave hierarchy involved by adoption of a cluster-type topology (a), (b), and a tree architecture (c),(d). If the RAN physical configuration is a cluster-type topology as illustrated in (a), hence the master-slave hierarchy is as illustrated in (b). If the mobile backhaul presents the tree architecture of (c), thus the master slave hierarchy is as shown in (d).

From the comparison of the two figures, it looks clear how the cluster-type topology has a simpler master-slave structure, thus easier to manage and configure. What is more relevant, in the cluster-type topology of Figure 6-5.a a unique link is connecting the cluster-head node to the Grandmaster Clock and the remaining five nodes are connected (synchronized) to the former through the ring topology. On the contrary, in the radial topology of Figure 6-5.c six separate links would be needed.

As a consequence, in the cluster-type topology the timing among the six RAN BSs belonging to the ring is more homogeneous than into the radial topology. As it will be seen in the next chapter, our proposal consists of a timing distribution mechanism within the ring which is independent from the rest of the traffic flow. This entails that the five RAN BSs besides the cluster-head RAN BS of Figure 6-5.a can be considered with good approximation as exact replicas of the latter. Which means that through the ring the time distribution within the different subdomains is homogeneous and the level 1 and level 2 of the master-slave hierarchy of Figure 6-5.b can be fused together into a unique level. With the adoption of the radial topology, each of the six nodes has to communicate with the Grandmaster Clock through standard PTP packets carried throughout the core network by the UDP/IP implementation; the communication takes place through six different paths, the nodes do not have the cluster-head RAN BS as common denominator. As a result, the issues explained in Section 4.8 will affect the timing performance in an independent manner from each other, causing dishomogeneity into the clock synchronization of the nodes.

6.5 General guidelines for the new time synchronization technique within the ring

The mechanism we propose is based on delivering the timestamps necessary for the synchronization algorithm independently from the traffic load. This aim is reached by prepending the interested messages to the time-frame. Messages containing timestamps are Sync, Follow_Up, Delay_Req and Delay_Resp. Management and signaling messages are exchanged as normal packets according to the OpMiGua HPTS scheme, i.e. inserting them into the assigned time-slot for circuit-switched traffic exchange between the two end nodes or making usage of underutilized resources for packet-switched point-to-point traffic.
Announce messages are not needed anymore. As mentioned in Section 4.3, in the IEEE 1588v2 protocol announce messages are used to carry the information of the capabilities of a clock to other clocks; the received information are then exploited for the computations foreseen by the BMC algorithm. The adoption of the cluster-type topology implies a well defined hierarchy within each fundamental ring of this physical configuration: the cluster head is Master Clock for all the nodes belonging to the ring, each answering to its own subdomain. The nodes composing the ring are in turn Master Clocks of the beneath reduced-size tree or radial architecture. Therefore, within the ring announce messages are not any longer transmitted.
Making the packets containing timestamps independent from the traffic load, together with the circuit-switching capability of the OpMiGua nodes, translates into the absence of Packet Delay Variation. Considering also the added value of a simple logical architecture, the right platform for an efficient and homogeneous timing distribution within the backhaul is provided.

6.5.1 A suggested structure for prepending synchronization packets into the OpMiGua HPTS scheme

As said in Section 6.1 and illustrated in Figure 6-1, the adoption of OpMiGua HPTS scheme into the ring topology translates into a train of time-slots travelling throughout the two links of the link, one clockwise and the other counter-clockwise. The train is formed by time-frames of fixed length within the ring, where the length is determined by the number of nodes (thus the minimum number of time-slots necessary is determined as derived in Section 5.4) and by the length of each time-slot. After the HPTS scheme is setup, every node is able to determine the beginning of each time-frame by simple bit counting, since being the length of those given.

We hereafter propose to take advantage of this organized structure prepending a time-frame header that contains the timing information, nominally the timestamps. The header is prepended by the cluster-head node. The latter is elected as Master Clock for all the nodes within the ring. In our proposal, the rest of the nodes (i.e. all the nodes belonging to the ring besides the cluster-head node) are allowed uniquely to read the prepended header while it is passing through without modifying it. It makes exception only a unique field, which can be inserted by one of the nodes only once per time-frame throughout the whole ring. In order to read the header without stopping or delaying the train of time-slots the structure proposed in Figure 6-6 can be exploited.

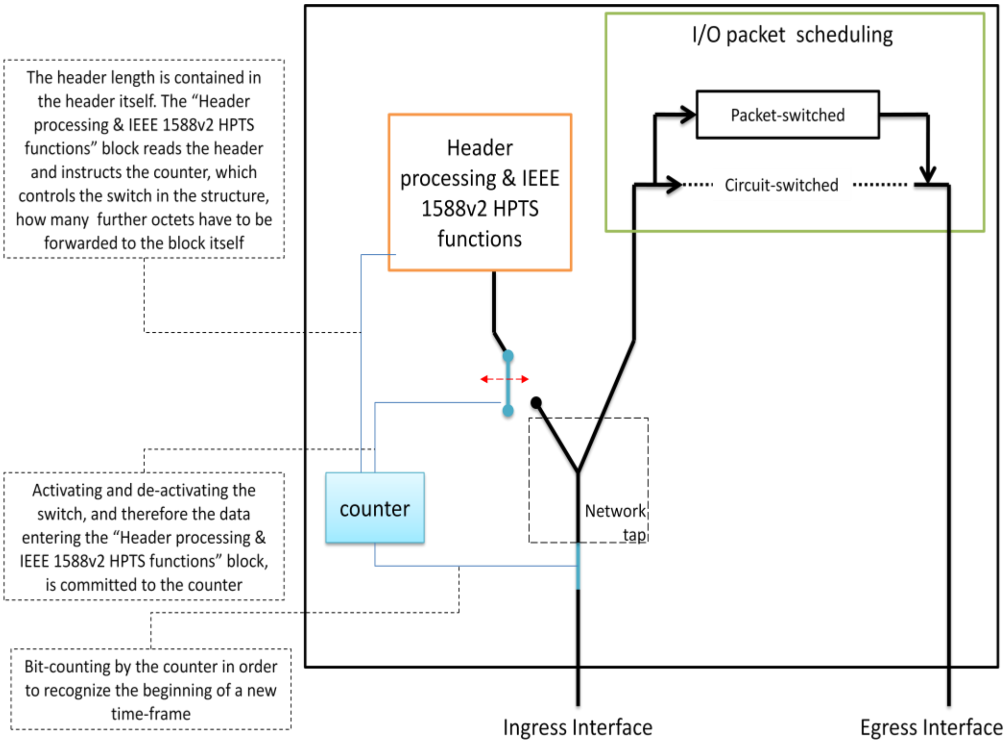


Figure 6-6. Suggested structure for header extraction without stoppage or delay infliction to the train of time-slots. The network tap refers to an electrical domain; in case of processing of an optical signal, it has to be substituted by a 1:2 optical beam splitter.

At the ingress interface a passive network tap is applied; a switch connected to a counter is used to activate and deactivate the data duplication by the tap. As said above, every time-frame has a fixed length; just by counting the bits (function of the counter), it is possible to activate the network tap when the new time-frame enters the node and to deactivate it when the header reception is completed. The header length is contained into the header itself. In case of an optical link, the network tap is substituted by an optical splitter 1:2 with an optical switch on the exit branch that forwards the received data to the "Header processing & 1588v2 HPTS functions" block. The same principle with the utilization of a counter for switching on and off the duplication of the received signal is applicable.

In Figure 6-6 only the structure, that is the fundamental blocks, is shown. How they interact between each other will be presented in Chapter 7 and Chapter 8. By means of this structure, the packets necessary for the completion of the timing algorithm are carried by node to node without stopping them. When they enter the node, they are duplicated and sent to the "Header processing & IEEE 1588v2 HPTS functions". On the other branch of the network tap, they are forwarded by the "I/O packet scheduling" block to the subsequent node of the ring. The forwarding of packets for synchronization is allowed by activating the switch; this is a duty of the counter. As a result, every node reads all the synchronization packets that are exchanged within the ring. Only the packets that are addressed to the node will be processed, the other discarded. How the node recognizes whether it is the addressee of the packet or not is going to be presented in Section 8.4.

6.5.2 An extended terminology for the IEEE 1588v2 HPTS scheme

As already mentioned while introducing the use case scenario (Section 6.1), the IEEE 1588v2 protocol is here adopted as platform for clock synchronization among the RAN BSs. The PTP is designed to be suitable to every packet-based network; hereafter the specific case of a network where OpMiGua nodes are organized according to the cluster-type topology and make usage of the OpMiGua HPTS scheme is taken into consideration. Thus some new terms have to be introduced in order to address our specific case.
Here are two concepts already utilized along the thesis but it is better to return on them, so that any source of confusion is avoided: the time-slots and the time-frames. A single **time-slot** (or a concatenation thereof) represent a circuit or connection in the network. A **time-frame** is here a series of time-slots where the latter are in a sufficient number to reach full mesh connectivity within the ring. Every time-frame includes a fixed number of time-slots. Every time-slot includes a fixed number of time-slots.
Our proposed technique consists of prepending a header and eventual messages at the beginning of the time frame. The header will be called with the name of "**IEEE 1588v2 HPTS header**"; it is always prepended to the time frame and its format and functions are described in Chapter 7. To the messages for synchronization purposes preprended to the time-frame we will refer as "**IEEE 1588v2 HPTS message**". We define it as a reduced PTP message which can be used for timestamp exchange as part of a synchronization algorithm based and integrated with the principles of IEEE 1588v2 Precision Time Protocol for specific utilization into a ring topology with OpMiGua HPTS scheme exploitation. The format and the functions of a IEEE 1588v2 HPTS message will be described in Chapter 8. Within the rings we will make usage of some particular addresses with local validity; they are meaningless out of a ring with nodes answering to the OpMiGua HPTS scheme. We will refer to this specific address as **IEEE 1588v2 HPTS address**. It is assigned through IEEE1588v2 management messages and included in all the management and signaling messages sent to the specific node. Its functions and format are pointed out in Chapter 7 and Chapter 8.

Chapter 7

The IEEE 1588v2 HPTS Header

In this Chapter the header which is prepended to each time-frame is presented. In Section 7.1 its format is described in all its fields. The function of each field is also illustrated. In Section 7.2 the focus falls upon a particular field of the header which allows the Slave clocks to communicate with the Master Clock. An example is presented in Section 7.3 in order to ease the reader's understanding.

7.1 Format and functions of the IEEE 1588v2 HPTS header

With the specific structure of a IEEE 1588v2 HPTS message it will be dealt at a second moment in time (Chapter 8) along the thesis; as first priority we want to make clear the role of the IEEE 1588v2 HPTS header into our exchange of the timestamps necessary for the timing algorithm. For the moment it is enough to born in mind that IEEE 1588v2 HPTS messages have a fixed length of 25 octets and they substitute some of the standard PTP messages within the ring topology.

The time-frame header, which has been said above we will refer to as IEEE 1588v2 HPTS header, consists of only two octets; its structure can be viewed in Figure 7-1. The first octet informs the nodes belonging to the ring if IEEE 1588v2 HPTS messages have been prepended to the train and, if it is so, how many they are. The second octet of the IEEE 1588v2 HPTS header allows the Slave Clocks, i.e. the nodes belonging to the ring, to send IEEE 1588v2 HPTS messages to the Master Clock, i.e. the cluster-head node, according to the directions given by the Master Clock itself. After these two octets informing the nodes how the prepended header is formed, eventual IEEE 1588v2 HPTS messages can be added (attached to the header but still prepended to the time-frame). A IEEE 1588v2 HPTS message for Slave-to-Master communication will always present. In case the designated Slave Clock does not have IEEE 1588v2 HPTS messages to send to the Master Clock, these 25 octets does not contain any valuable information and will be padded with zeros.

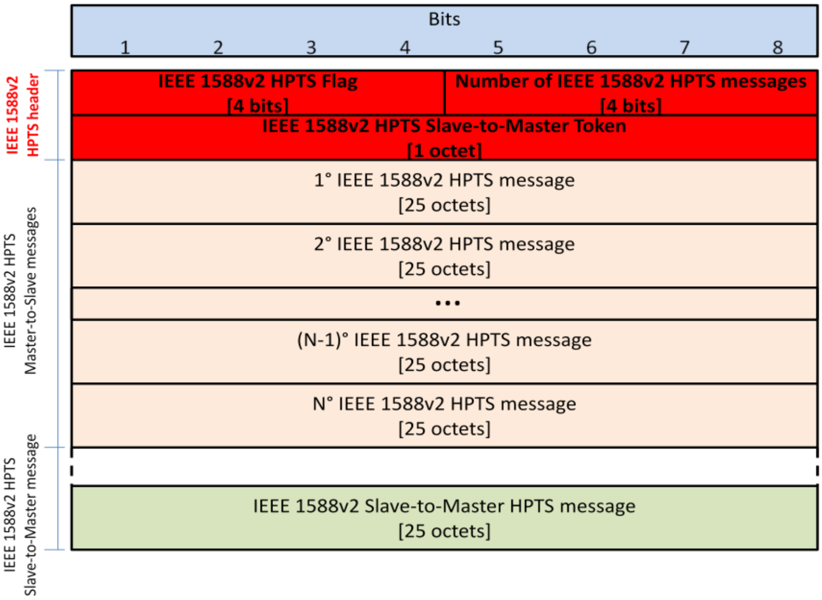


Figure 7-1. Format of the IEEE 1588v2 HPTS header, followed by eventual IEEE 1588v2 HPTS messages and the always-present IEEE 1588v2 HPTS message for Slave-to-Master communication.

The IEEE 1588v2 HPTS header is made up by the following fields:

- *4 bits: IEEE 1588v2 HPTS flag.* It advices of the presence of one or more IEEE 1588v2 HPTS messages from the Master Clock in the cluster-head node to one or more Slave Clocks within the ring. In case one of more IEEE 1588v2 HPTS messages from the Master Clock (i.e. besides the IEEE 1588v2 HPTS message for Slave-to-Master communication) are present, this field is set to 1111. In the opposite case, that is no IEEE 1588v2 HPTS messages present, the field is set to 1111 and the data duplication is limited to the two octets of the IEEE 1588v2 HPTS header.
- *4 bits: Number of Synchronization Messages*. This value indicates how many IEEE 1588v2 HPTS messages are going to be prepended to the train of time-slots; in case no synchronization messages are present, this value is set to 0000. As stated into the description of the use case scenario (Section 6.1), each ring is expected to contain from 10 to 15 nodes; with 4 bits the worst case is servable, i.e. the 15 nodes belonging to 15 different subdomains. Even if this is an option that has to be taken into consideration, it is really unlike to take place. Still as stated into the use case scenario, a realistic case might be up to 4 mobile providers (likely from 2 to 3) operating within the same ring. When the IEEE 1588v2 HPTS header is passing through the node, if the IEEE 1588v2 HPTS flag is activated then this field is read and the number of bits that have to be duplicate from the network tap (or optical splitter) structure is increased by this value times 25 octects, i.e. the fixed length of an IEEE 1588v2 HTPS message.
- *8 bits: IEEE 1588v2 HPTS Slave-to-Master Token*. This octet is used from the cluster-head node to instruct the nodes belonging to the ring which one is allowed to send a IEEE 1588v2 HTPS message to the Master Clock (hence to the cluster-head node itself) into the next time-frame. In a nutshell, it can be seen as a token distributed from the cluster-head node to the nodes, one token to a unique node per time-frame. This token consists of a IEEE 1588v2 HPTS address. How this specific field is used into the Slave-to-Master communication is explained into the next Section.

7.2 An organized scheme for Slave-to-Master communication

It is said above that, through the insertion of a IEEE 1588v2 HPTS address into the last field of the IEEE 1588v2 HPTS header, a node belonging to the ring (i.e. a Slave Cock) is allowed to send a IEEE 1588v2 HPTS message to the Master Clock. In this Section the organized scheme for the regulation of the Slave-to-Master message flow is presented. The presentation is followed by an example, in order to bring up a real case and describe step by step the scheme. In the example it is dealt only with the mechanism for managing the insertion of IEEE 1588v2 HPTS messages by nodes hosting a clock in the slave position.
Before all this, it is worth to dwell upon the IEEE 1588v2 HPTS address, at the base of our scheme, and its assignment. We propose to assign the IEEE 1588v2 HPTS address by means of a management message. As already mentioned in the general guidelines of our technique (Section 6.5), synchronization messages that do not contain a timestamp are exchanged as normal packets. Which means that a management message is either inserted into the time-slot assigned to the destination node (circuit-switched path) or fit into underutilized resources (packet-switched path). They will answer to the standardized format of PTP messages, thus as explained in Section 4.7. The following managementTLV field is proposed for IEEE 1588v2 HPTS address assignment:

- tlvType: 0x0001 (MANAGEMENT)
- lengthField: 0x0001(value field consists exclusively of one octet)
- ManagementID: 0x0001 (a random value is used here, to be chosen out of the full list of Management IDs contained into the IEEE 1588v2 standard)
- dataField: the IEEE 1588v2 HPTS address assigned to the node

Into all the signaling and management messages directed to the nodes, the IEEE 1588v2 HPTS address proper of the target node has to be included as first TLV or managementTLV field respectively. This condition allows to have a fresh check upon the IEEE 1588v2 HPTS addresses within the ring, after they have been assigned to the nodes. The same ManagementID value used for the assignment of the IEEE 1588v2 HPTS address can be used also for its "*check*" carried out as just explained.
By means of the IEEE 1588v2 HPTS address and the IEEE 1588v2 HPTS StM (Slave-to-Master) Token, the transport of IEEE 1588v2 HPTS messages from the Slave Clocks (i.e. nodes belonging to the ring) to the Master Clock (i.e. the cluster-head node) is made possible. The scheme is carried out as follows:

- The cluster-head RAN BS starts the transmission of a time-frame: the IEEE 1588v2 HPTS header is prepended. In the header, the IEEE 1588v2 HPTS address of the node that will be allowed to send a IEEE 1588v2 HPTS message to the Master Clock into the next time-frame is inserted into the IEEE 1588v2 HPTS StM Token field. Eventual IEEE 1588v2 HPTS messages are thus prepended to the time-frame, after the IEEE 1588v2 HPTS header. When all the synchronization messages have been clocked out, but still before the set of time-slots begins, an amount of bits set to 0 equals to 25 Bytes (i.e. the size of a IEEE 1588v2 HPTS message) is inserted. Thus the first time-slot of the time-frame begins.
- At every node, the counter for activate and deactivate the switch present in Figure 6-6. (i.e. the data duplication and consequent analysis from the *Header processing and IEEE 1588 HPTS functions* block) is set to the minimum value of two octets. This makes every node able to read the IEEE 1588v2 HPTS header, also in the case that no IEEE1588v2 HPTS messages from the Master Clock are present. To be precise, the reading of the IEEE 1588v2 HPTS header as well as of the eventual IEEE 1588v2 HPTS messages is committed to the *Header processing and IEEE 1588 HPTS functions* block. Those, in addition to the 25 "padding octets" (set to all 0s by the cluster-head node), on the other branch of the network tap are just let pass through the *I/O packet*

scheduling block thanks to the circuit switching capabilities of the OpMiGua node. What is important here is to point out that, as a result, the IEEE 1588v2 HPTS StM Token is read by every node.

- The value inserted by the cluster-head node into the IEEE 1588v2 HPTS StM Token field is hence read by the *Header processing & IEEE 1588 HPTS functions* block. In case it matches with the node's own IEEE 1588v2 HPTS address, we can define the node as "*holding the token*" at the next time-frame passing through it.
- When a new time-frame enters into the node, in the case of a node holding the token, the 25 "padding octets" (set to all 0s) are going to be ignored and a Delay_Req in the format of IEEE 1588v2 HPTS message for the node's own Master Clock (i.e. the cluster-head node) is going to be inserted instead. In case the node is holding the token but does not need to send a Delay_Req message to its Master, then the padding octets are substituted with other 25 octets padded with 0s. The substitution of the 25 octets with the Delay_Req or eventually with other 25 octets of 0s interests only the *I/O packet scheduling* block.
- The time-frame with eventually the Delay_Req message substituting the 25 padding octets travels through node to node into the ring topology. All the other nodes did not receive the IEEE 1588v2 HPTS StM Token in the previous time-frame, i.e. they are not holding the token, and hence their *I/O packet scheduling* blocks just let the 25 octets pass through unmodified.
- Handed over from node to node with the IEEE 1588v2 HPTS header, the eventual IEEE 1588v2 HPTS messages and the 25 octets for Slave-to-Master communication prepended, the time-frame finally reaches the cluster-head node. Into the cluster-head , uniquely the last 25 prepended octets are taken into consideration. In case they are not set to all 0s, then they make up a Delay_Req message in the format of a IEEE 1588v2 HPTS message and processed according to the PTP protocol.

7.3 Example of Use Case on Slave-to-Master communication

In order to make clear the scheme above described for the Slave-to-Master communication within a ring using the OpMiGua HPTS scheme for hybrid circuit establishing, a use case is hereafter presented. We here disregard how the IEEE 1588v2 HPTS messages are structured; a detailed analysis is going to be provided in the next chapters. We just have to bear in mind that they have a fixed length of 25 octets and they contain the timestamps necessary for the timing algorithm (i.e. t_1, t_2, $t_3$and t_4 of Figure 4-1).
The scenario of Figure 7-2 is used: five nodes connected into a unprotected bidirectional ring topology, the node 1 is hosting the cluster-head RAN BS. Into the table next to the ring topology, it is summarized the status of the IEEE 1588v2 HPTS Slave-to-Master token; when a node is holding the token, an **x** will be present (clockwise or counter-clockwise) on its line.

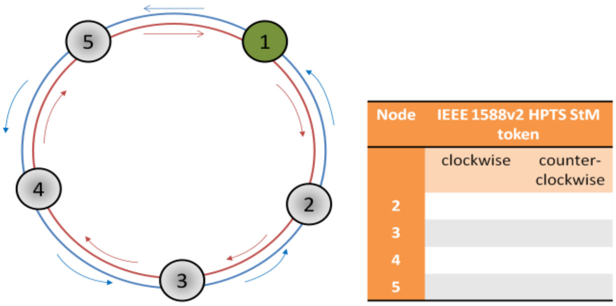


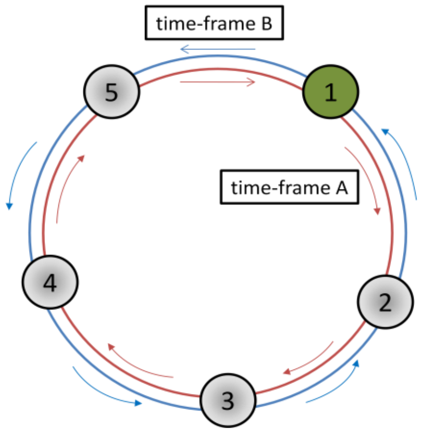
Figure 7-2. The scenario for the use case presented. Bearing in mind that the ring physical layer configuration represents the fundamental topology for the cluster-type topology, a ring network with 4 nodes beside the cluster-head node is adopted for our use case. In green is the cluster-head node. Throughout the blue link, the counter-clockwise communication takes place; throughout the red link, packets are sent clockwise.

For sake of simplicity, we assume that only one time-frame per link will be present; this is an unrealistic assumption but allows to make the use case of easier and of more linear understanding. For the use case, we also assume a fixed length of 1000 Bytes per time-slot. According to Section 5.4 and as illustrated in Figure 5-8as well, three time-slots are necessary to assure a full-mesh connectivity among the nodes of a ring composed by 5 elements. We also suppose the nodes to have the IEEE 1588v2 HPTS addresses been assigned as illustrated in Table 7-1.

Node	IEEE 1588v2 HTPS address
2	0x01
3	0x02
4	0x03
5	0x04

Table 7-1. IEEE 1588v2 HPTS addresses assigned to the nodes of the use case scenario

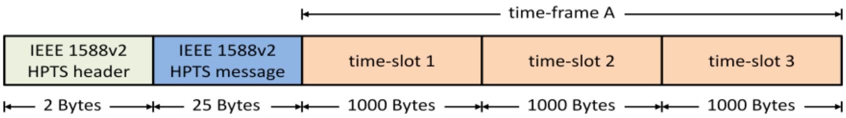
After the assumptions have been stated, we can analyze step by step our proposal of the internal scheme which allows the Slave Clocks organized into a ring structure to communicate with the Master Clock. The node 1 (i.e. the cluster-head node) starts transmitting the first two time-frames, one on the clockwise link (time-frame A) and one on the counter-clockwise link (time-frame B). Let suppose that no IEEE 1588v2 HPTS messages to any Slave Clock were to be transmitted. Therefore on the network the time-frames and the IEEE 1588v2 HPTS StM token are distributed as resumed in Figure 7-3.



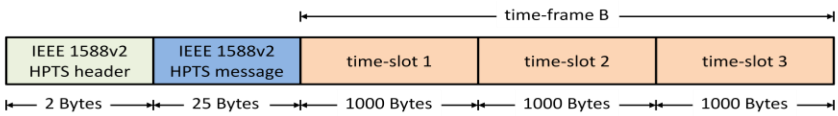
Node	IEEE 1588v2 HPTS StM token	
	clockwise	counter-clockwise
2		
3		
4		
5		

Figure 7-3. Distribution of time-frames and IEEE 1588v2 HPTS StM tokens over the network after the first transmission of time-frames

Since at the moment of transmission of time-frame A and time-frame B no IEEE 1588v2 HPTS messages were to be transmitted, the two time-frames are structured as illustrated in Figure 7-4.



Field	Value	Binary value
IEEE 1588v2 HPTS flag	0xA	1010
Number of IEEE 1588v2 HPTS messages	0x0	0000
IEEE 1588v2 StM token	0x01	00000001

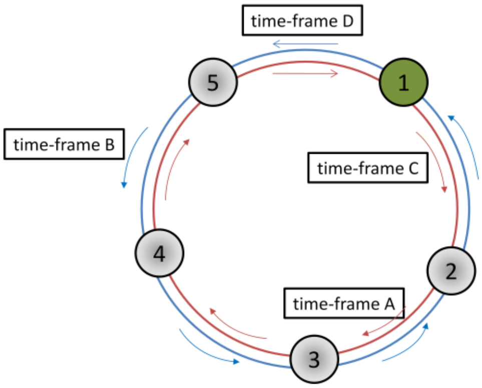


Field	Value	Binary value
IEEE 1588v2 HPTS flag	0xA	1010
Number of IEEE 1588v2 HPTS messages	0x0	0000
IEEE 1588v2 StM token	0x04	00000100

Figure 7-4. Structure of time-frames A and B of the use case. In detail the format of their IEEE 1588v2 HPTS headers.

In both the time-frames, after the IEEE 1588v2 HPTS header, 25 octets of padding are added. What indicated as the blue IEEE 1588v2 HPTS message is actually filled with 25 octets of zeros, which are supposed to be modified a real message by the nodes belonging to the ring. The token for Slave-to-Master communication is assigned to node 2 from time-frame A and to node 5 from time-frame B. When the time-frames reach node 2 and node 5, the *Header processing & IEEE 1588v2 HPTS functions* block reads the “IEEE1588v2 HPTS StM token” field and compare it with its personal IEEE 1588v2 HPTS address. Since they match, they instruct their *I/O packet scheduling* blocks informing them that at the next time-frame passing through they will be “*holding the token*”. IT means that the 25 octets of padding are to be excluded and a Delay_Req in the format of a IEEE 1588v2 HPTS message can be inserted instead. Or, in case

the node does not need to send a Delay_Req message, other 25 octets of padding have to be inserted.
The transmission of two new time-frames is carried out from node 1: time-frame C clockwise and time-frame D counter-clockwise. In the meantime time-frame A has been clocked out by node 2 and is located on the link between node 2 and node 3. The same with time-frame B, now located on the link between node 5 and 4. Node 2 has been instructed from time-frame A to insert an eventual IEEE 1588v2 HPTS message, if present, addressed to the Master Clock into the next frame passing through clockwise. Time-frame B had handed in the virtual token for the Slave-to-Master communication in the counter-clockwise direction to Node 5. The two tokens (clockwise and counter-clockwise) are independent from each other, as they are concerned with different interfaces. Thus the complete situation is as summarized in Figure 7-5.



Node	IEEE 1588v2 HPTS StM token	
	clockwise	counter-clockwise
2	x	
3		
4		
5		x

Figure 7-5. Distribution of time-frames and IEEE 1588v2 HPTS StM tokens over the network after the second transmission of time-frames

We suppose that at the moment of this second transmission a HPTS 1588v2 HPTS message is to be transmitted to the Slave Clocks of a particular subdomain; this message is sent in both directions in order to have the shortest path to all the nodes of the ring, whether they are positioned closer to the node 1 following a clockwise or a counter-clockwise path. Thus the four time frames present into the ring are structured as illustrated in Figure 7-6.

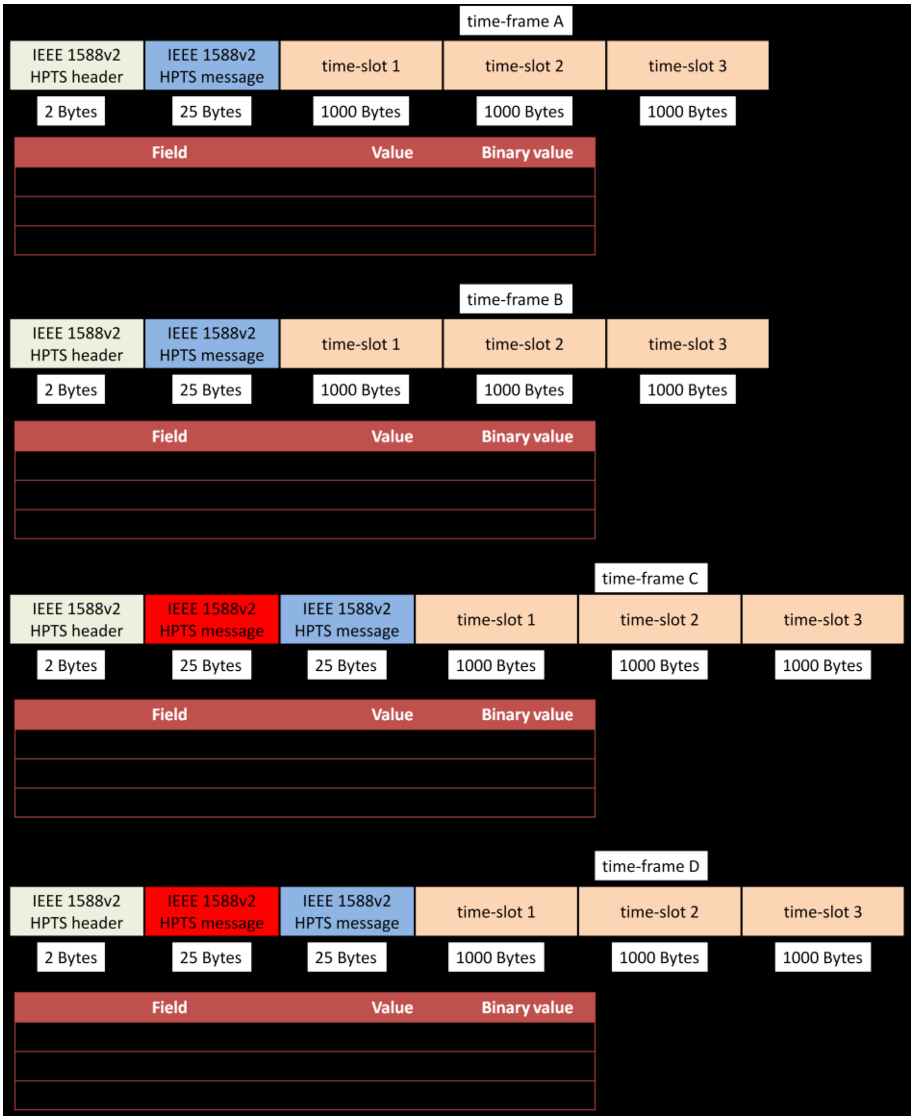
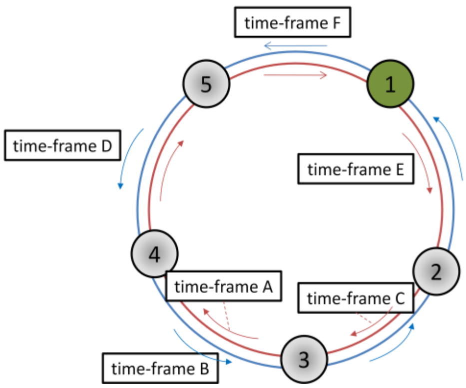


Figure 7-6. Structure of time-frames A, B, C and D of the use case. In detail the format of their IEEE 1588v2 HPTS headers

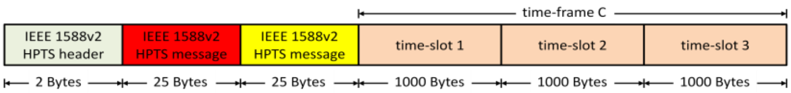
In time-frame C and time-frame D, the IEEE 1588v2 HPTS flag is activated and the field which indicates the number of IEEE 1588v2 HPTS messages carried is set to 1. With regard to the IEEE 1588v2 HPTS StM token, time-frame C carries the token for Node 3 while time-frame D assigns the right for Slave-to-Master transmission to Node 4. When time-frames C and D reach node 2 and 5 respectively, the "IEEE1588v2 HPTS StM token" is read and, since in both cases there is no match between the contained value and the own IEEE 1588v2 HPTS address, the communication with the *I/O packet scheduling* block is not performed. The usual 25 octets of padding are transmitted by the node 1 before the first bit of the first time-slot is clocked out.
The transmission of two new time-frames is carried out from node 1: time-frame E clockwise and time-frame F counter-clockwise. The first contains the IEEE 1588v2 StM token for Node 4, the second for Node 3. The location of all the six time-slots and which nodes are holding the two tokens is as resumed in Figure 7-7.



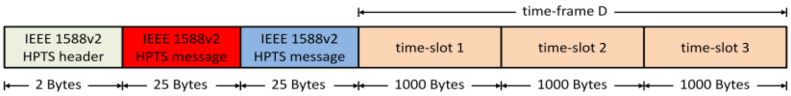
Node	IEEE 1588v2 HPTS StM token	
	clockwise	counter-clockwise
2		
3	x	
4		x
5		

Figure 7-7. Distribution of time-frames and IEEE 1588v2 HPTS StM tokens over the network after the third transmission of time-frames

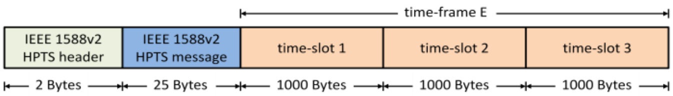
We assume that at the moment of transmission the Master Clock (i.e. the cluster-head node) has no IEEE 1588v2 HPTS messages for the Slave Clocks to be transmitted. The structures of time-frame A and B are not modified, while rest of the time-frames have the formats shown in Figure 7-8.



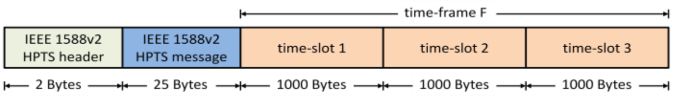
Field	Value	Binary value
IEEE 1588v2 HPTS flag	0xF	1111
Number of IEEE 1588v2 HPTS messages	0x1	0001
IEEE 1588v2 StM token	0x02	00000010



Field	Value	Binary value
IEEE 1588v2 HPTS flag	0xF	1111
Number of IEEE 1588v2 HPTS messages	0x1	0001
IEEE 1588v2 StM token	0x03	00000011



Field	Value	Binary value
IEEE 1588v2 HPTS flag	0xA	1010
Number of IEEE 1588v2 HPTS messages	0x0	0000
IEEE 1588v2 StM token	0x03	00000011



Field	Value	Binary value
IEEE 1588v2 HPTS flag	0xA	1010
Number of IEEE 1588v2 HPTS messages	0x0	0000
IEEE 1588v2 StM token	0x02	00000010

Figure 7-8. Structure of time-frames C, D, E and F of the use case. In detail the format of their IEEE 1588v2 HPTS headers

The relevant change here in Figure 7-8 from Figure 7-6 is the IEEE 1588v2 HPTS packet marked as yellow in time-frame C. Let suppose that, at the moment in which time-frame C entered the Node 2, a Delay_Req message was to be sent to the cluster-head node. Since time-frame A gave to Node 2 the permission (i.e. the virtual token) to send a IEEE 1588v2 HPTS packet to the Master Clock (i.e. a Delay_Req packet) into the next time-frame passing through the node clockwise, Node 2 substituted the 25 octets of padding prepended to time-frame C with its own IEEE 1588v2 HPTS packet. This is done according to the following mechanism:

- Node 2, as illustrated in Figure 7-7, is holding the token for slave-to-master communication given by time-frame A. The *I/O packet scheduling* block has been instructed and ready to perform the packet insertion into the next time-frame passing through (clockwise in this case)
- Time-frame C enters node 2; the two octets of the IEEE 1588v2 HPTS header are duplicate by the structure of Figure 6-6 and processed. The IEEE 1588v2 HPTS flag is activated, thus the field *Number of IEEE 1588v2 HPTS messages* is read: one IEEE 1588v2 HPTS message for the Slave Clocks is present. The *I/O packet scheduling* block is informed of the presence of this message.
- The time-frame C enters the *I/O packet scheduling.* The first two octets are let pass through as well as the IEEE 1588v2 HPTS message marked as red in Figure 7-8. When the last bit of the latter has been clocked out, the 25 padding octets that are coming after are ignored and the 25 octets of the IEEE 1588v2 HPTS message marked as yellow in Figure 7-8 are inserted.
- The first bit of time-slot 1 can be inserted, hence the whole time-frame. The packets carried into the time frame are processed and forwarded exclusively by the *I/O packet scheduling*.
- When time-frame C is passing through the other nodes until it reaches the cluster-head node again, the IEEE 1588v2 HPTS message inserted is considered in the same way as the 25 padding octets. In fact, none of the other nodes is holding the IEEE 1588v2 HPTS StM token for this specific time-frame. As a result they just let pass through the total 52 Bytes (2+25+25) prepended to the three time-slots of time-frame C.

The process is illustrated in Figure 7-9 and Figure 7-10. It refers to the specific case of the time-frame C of our use case, but a variable number of IEEE 1588v2 HPTS messages can be attached before the 25 octets of padding.

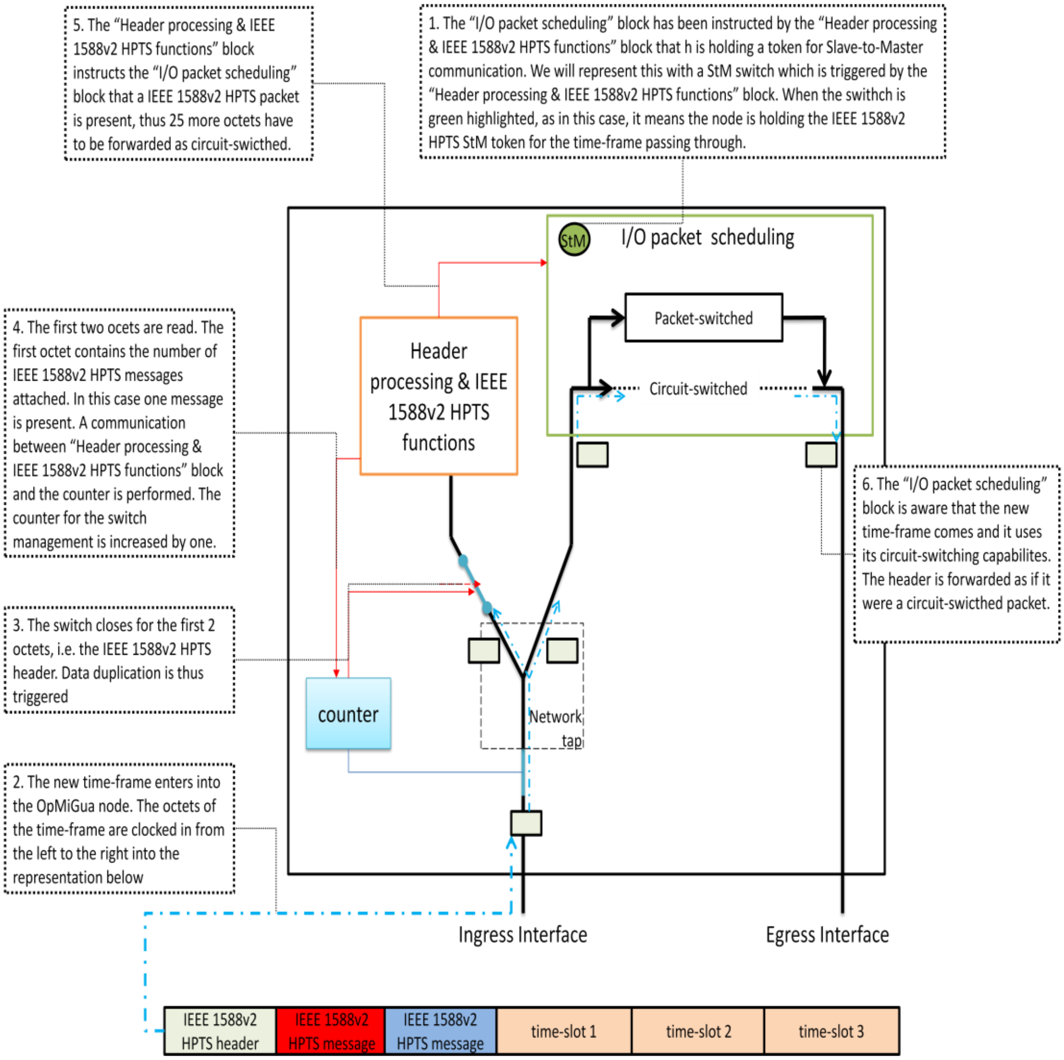


Figure 7-9. Processing of the IEEE 1588v2 HPTS header prepended to the time-frame (1 of 2)

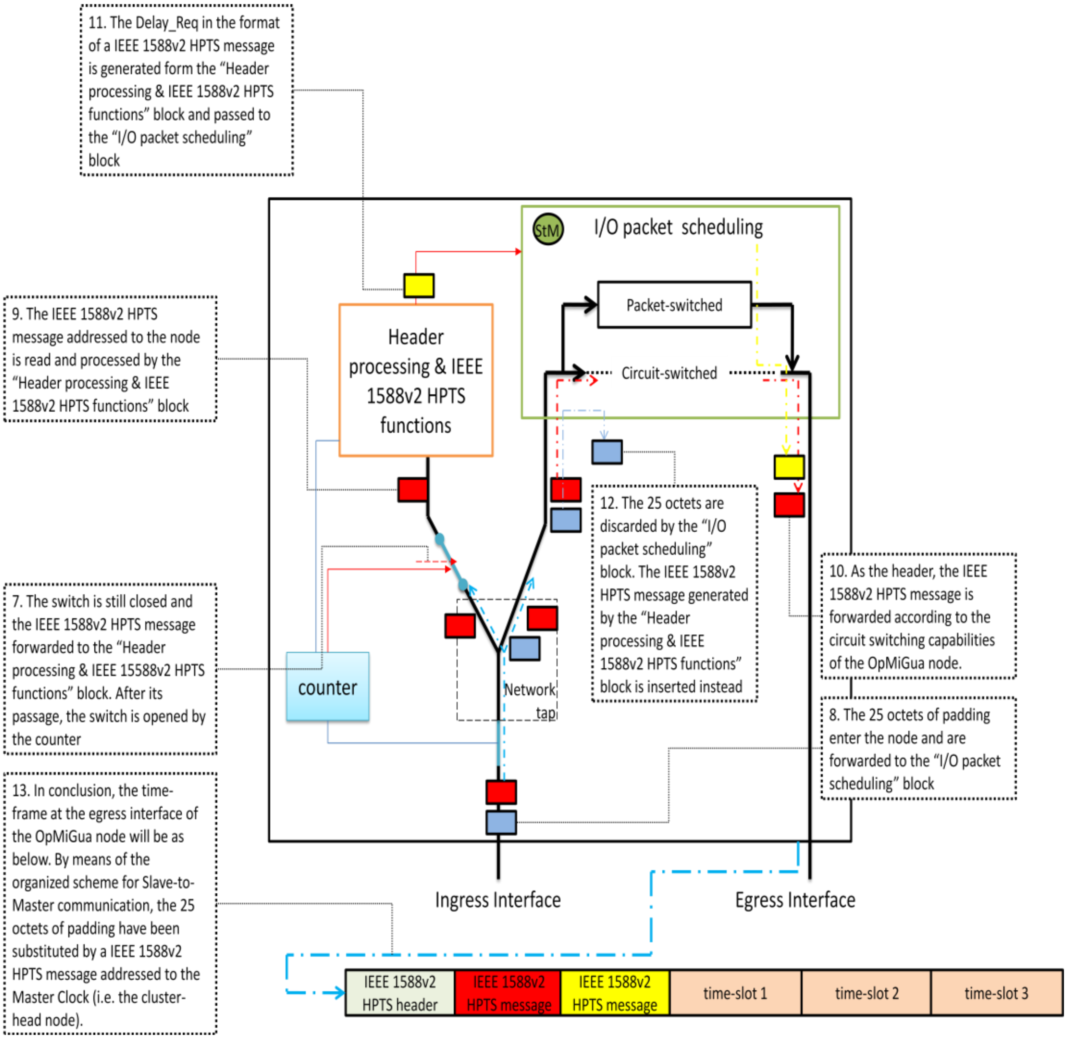


Figure 7-10. Processing of the IEEE 1588v2 HPTS header prepended to the time-frame (2 of 2)

When the time-frame C has completed the passage through all the nodes of the ring, it reaches the cluster-head node again. The latter will be aware of which node was allowed to send its own IEEE 1588v2 HPTS message to the Master Clock into that specific time-frame, having been Node 1 itself that distributed the IEEE 1588v2 HPTS StM tokens. Thus the packet marked as yellow in Figure 7-10 will be sent to the correct port of Figure 6-4 according to the subdomain the sending node belongs to.

Chapter 8

The IEEE 1588v2 HPTS Message

In this Chapter we bring our proposal for the format of a message bearing a timestamp. In Section 8.1 the format and the functions of each field are illustrated. In Section 8.2, since the messages are duplicated and read by each node, is presented an organized scheme to distinguish each chain of 4-message exchange between two nodes within the ring. As in the previous Chapter, the scheme is subsidized by an example to facilitate its comprehension. The example is contained in Section 8.3. In conclusion, Section 8.4 it is proposed how a IEEE 1588v2 HPTS message is processed in each of the nodes hosting a Slave Clock.

8.1 Format and functions of a IEEE 1588v2 HPTS message

Throughout the two previous sections, the frame format of the IEEE 1588v2 HPTS messages has been disregarded. All it has been said, is to bear in mind that a IEEE 1588v2 HPTS message has a fixed length of 25 octets. In this section we are going to propose its frame format. The various fields contained into the 25 octets and their functions are hereafter presented.
The structure can be viewed in Figure 8-1 beneath.

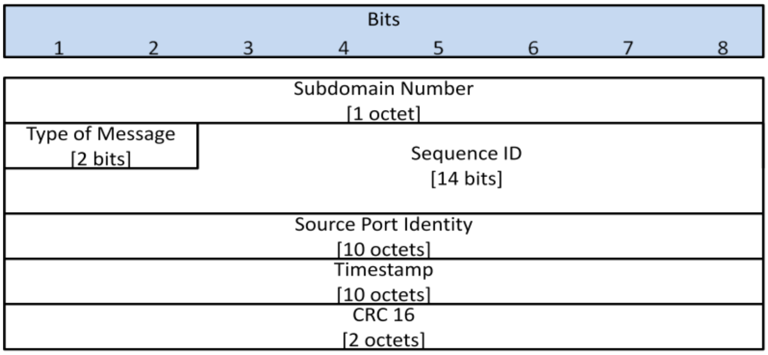


Figure 8-1. Format of a IEEE 1588v2 HPTS message

The IEEE 1588v2 HPTS message is made up by the following fields:

- 8 bits: Subdomain number. This field identifies the domain the IEEE 1588v2 HPTS message belongs to. As stated in Section 6.3, subdomains are logical synchronization systems using different time-bases (i.e. different reference Grandmaster Clocks) and independently from each others. The length of 8 bits is due to compatibility with IEEE 1588v2 protocol, where the same length is defined for subdomain identification.
- 2 bits: Type of Message (ToM). This field identifies which type of message is contained into the IEEE 1588v2 HPTS message. As stated in Section 6.5, only messages

containing timestamps are exchanged by preprending them to the time-frame; this makes them independent from traffic load and reduces dramatically the Packet Delay Variation of these specific packets. Hence, the only packets that are exchanged as IEEE 1588v2 HPTS messages are Synch, Follow_Up, Delay_Req and Delay_Resp messages. The other messages (i.e. signaling or management) are sent as standard PTP packets, that is by inserting them into the appropriate time-slot if circuit-switched quality wants to be provided or inserting them in under-utilized resources as SM packets. As a result, with 2 bits I can define all the necessary messages. The values of Table 8-1 are proposed.

Bit values		Type of Message
1	2	
0	0	Delay_Req
0	1	Sync
1	0	Follow_Up
1	1	Delay_Resp

Table 8-1. Proposed values for the field Type of Message

- 14 bits: Sequence ID. With this field, the message exchange between the Master Clock and a specific Slave Clock is carried out. The timing algorithm necessitates of four timestamps that are exchanged through a chain of four messages: in succession, Synch, Follow_Up, Delay_Req and Delay_Resp. Every chain has to be univocally identified. Every timestamp carried or triggered by a IEEE 1588v2 HPTS message has to be inserted without possibility of mistake into the correct chain of messages. This is committed to the Sequence ID field. Its detailed functioning will be explained in the next Section 8.2.
- 80 bits (10 octets): Source Port Identity. This field identifies the originating port for the message. As stated in Section 6.3, the cluster-head node receives different clock information from different Grandmaster Clocks and distributes this set of information into the ring. The subdomains present into the ring must be able to communicate with the cluster-head without interfering with each others. For this reason the multi-port structure of Figure 6-4 has been proposed. The address of the port originating the message is inserted in this field. The length of 10 octets is due to compatibility with IEEE 1588v2 protocol, where the same length is defined for the port address.
- 80 bits (10 octets): Timestamp. In this field is inserted the timestamp carried by the IEEE 1588v2 message. The length of the timestamp respects the length defined in IEEE 1588v2 protocol: 48 bits of seconds and 32 bits of nanoseconds. If a different time-scale from UTC is used, the length must stay fixed to 80 bits.
- 16 bits (2 octets): CRC-16. In this field a checksum over the IEEE 1588v2 HPTS packet is inserted, to assure integrity of data. This prevents bit errors from affecting the success of the timing algorithm. In case of error detection, the IEEE 1588v2 HPTS packet is discarded. As stated in Section 4.8, whilst a lost packet causes little or no effect on the clock recovery performance, a corrupted timestamp might move the offset computation from the expected value. The Cyclic Redundancy Code (CRC) 16 is used; it provides an output of 16 bits which is inserted into this field.

By means of the structure previously proposed (Figure 6-6) the IEEE 1588v2 HPTS messages in the Master-to-Slave direction (i.e. packets originated at the cluster-head node and addressed to one or several nodes belonging to the ring) are forwarded from node to node. As already mentioned in the general guidelines of our proposal (Section 6.5), the result of this scheme is that every node reads all the synchronization packets that are exchanged within the ring. Only the packets that are addressed to the node will be processed, the other discarded. The scheme for unambiguous identification is explained in the next Section.

8.2 An organized scheme for unambiguous identification of the master-slave chain of messages

In this section we will illustrate how each node recognizes whether it is the addressee of a message or not. In particular how the 4-message exchange is performed between the Master and the Slave Clocks, and how this exchange is univocally identified within the ring.
The Sync message initializes the timing algorithm; a fundamental requirement for the success of the offset calculation is a clear correlation of the Sync message with the other messages containing and triggering the remaining timestamps necessary for the final computation. This correlation is committed to the *Sequence ID* field. When the cluster-head node sends a Sync message to all the Slave Clocks of a particular subdomain, a Sequence ID value is established and incorporated into the IEEE 1588v2 HPTS message. The subsequent Follow_Up message carrying the precise value of t_1 will be stamped with the same Sequence ID of the Sync message it refers to. In this way the receiving node is able to insert the received value of t_1 into the timing algorithm initialized by the related Sync message a kiss without possibility of mistake. What has to be avoided is possible conflicts among Sequence IDs inside a specific subdomain.
In fact, as we will see in Section 8.4, the nodes besides the Master Clock decide whether to process or discard a packet on the basis of the Subdomain Number and the Sequence ID fields. For the Sync and Follow_Up message, the only Sudomain Number field is enough. A Sync message, and the correlated Follow_Up message, is processed by all the nodes belonging to the same subdomain. Which means the transmission of a Sync message from the Master Clock triggers the timing algorithm in all the Slave Clocks belonging to the subdomain that the Sync message is addressed to. Hence it happens that all the nodes, in turn when they are holding the StM token as explained in Chapter 4 and according to the four-message exchange, reply sending a Delay_Req message. Here an organized scheme has to be introduced in order to avoid the conflicts among messages addressed to Slave Clocks belonging to the same subdomain mentioned above. Without an organized scheme, all the reply Delay_Resp messages would be stamped with the same *Sequence ID* value: the identification of the addressee of the message within the subdomain would not be possible anymore. Unambiguous distinction of the chains of messages between subdomains and within the same subdomain is a fundamental requirement for the success of the timing algorithm.
The organized scheme we are going now to explain is based on the utilization of IEEE 1588v2 HPTS addresses. They have been defined in Subsection 6.5.2 as addresses with local validity, whilst meaningless out of the ring made up by nodes answering to the OpMiGua HPTS scheme. Let us remind to the reader that a IEEE 1588v2 HPTS address has a length equal to 8 bits. The platform of our suggested scheme resides in the Master Clock, i.e. the cluster-head node. In fact the origination of a new master-slave chain of messages is committed to the cluster-head node which marks a fresh Sync message with a new *Sequence ID* value. When a new Sync message is originated, the value to be inserted into the Sequence ID value is increased by 256 from the precedent value. In this way, the least 8 important bits of the Sequence ID field are not modified from one Sync message to the sequent; we can assume here to set the 8 least important bits to all zeros.
As an example, we can assume a Sync message is sent with the Sequence ID value equals to 0x3700, as illustrated into Figure 8-2.

Bits													
13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	1	0	1	1	1	0	0	0	0	0	0	0	0

Figure 8-2. Example of the Sequence ID value of an hypothetical Sync message in the format of a IEEE 1588v2 HPTS message.

The related Follow_Up message will carry the same Sequence ID value. The next Sync message sent by the Grandmaster Clock, to the same or a different subdomain, will have the Sequence ID obtained by the one of the previous Sync message increased by 256. The 8 least important bits are hence not modified. As a result, the next Sync message will be characterized by a Sequence ID value of 0x3800, as illustrated into Figure 8-3.

Bits													
13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	1	1	0	0	0	0	0	0	0	0	0	0	0

Figure 8-3. Sequence ID value of the hypothetical Sync message in the format of a IEEE 1588v2 HPTS message which follows the Sync message of the example illustrated in the previous figure.

The expedient of to not modify the 8 least important bits allows the Slave Clocks to establish different "message exchange flows" (i.e. different 4-message chains) with the Master Clock without interfere among each others. This is done by marking the Delay_Req for the Master Clock with the following value:

$$(\text{Sequence ID}) \oplus (\text{IEEE 1588v2 HTPS address})$$

where the symbol $\oplus$ represents the XOR operator and at its right is the own address of the sending node. Being the IEEE 1588v2 HPTS address unique for each node within the ring and 8-bit long, this operation modifies only the 8 least important bits of the Sequence ID value and avoids any ambiguity within the ring. Every Delay_Resp message in the format of a IEEE 1588v2 HPTS message will be stamped with a different *Sequence ID* value, which combined with the *Subdomain Number* allows the Slave Clock to recognize its own messages and to exclude messages addressed to other nodes.

8.3 Example of Use Case on unambiguous identification of the master-slave chain of messages for timing algorithm completion

In order to make clear the scheme above presented to establish different "message exchange flows" within the same subdomain and between different subdomains, a use case is hereafter presented.
For our use case, let suppose to be in a ring with ten nodes besides the cluster-head node and four of them belonging to the same subdomain. Let suppose the subdomain being identified by the value 0x65 (01100101_2); this *Subdomain Number* will identify our subdomain of interest. The Use case scenario is thus as illustrated in Figure 8-4. In green is highlighted the cluster-head node, i.e. the Master Clock, while in red the nodes belonging to the specified subdomain. The grey nodes are the rest of the nodes.

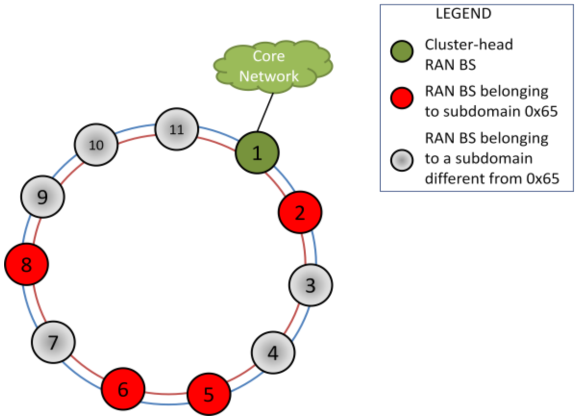


Figure 8-4. Scenario for our use case. Ten nodes belonging to the ring besides the cluster-head node. Four nodes belong to the subdomain identified by the Subdomain Number 0x65. In green is the cluster-head node. In red are the nodes belonging to the subdomain of interest. In grey are the other nodes.

We also suppose that the IEEE 1588v2 HPTS addresses assigned to the 4 nodes belonging to the subdomain 0x65 are as illustrated in Table 8-2.

Node	IEEE 1588v2 HTPS address
2	0x01 (00000001_2)
5	0x04 (00000100_2)
6	0x05 (00000101_2)
8	0x07 (00000111_2)

Table 8-2. IEEE 1588v2 HPTS addresses assigned to the nodes of the use case scenario belonging to our subdomain of interest

We now suppose that a Sync message is sent to the subdomain of interest, that is the one identified by the Subdomain Number 0x65. Let also suppose that a Sequence ID value equals to 0x3700 is here generated from the cluster-head node and inserted into the IEEE 1588v2 HPTS message. Therefore the Sync in the format of a IEEE 1588v2 HPTS message will be as illustrated in Figure 8-5. Neglecting the values of the Source Port Identity, the Timestamp and the CRC 16 fields, only the first three octets of the message are reported. The Subdomain Number, Type of Message and Sequence ID fields are those important for our unambiguous identification of the master-slave message exchange.

Bits							
1	2	3	4	5	6	7	8
0	1	1	0	0	1	0	1
0	1	1	1	0	1	1	1
0	0	0	0	0	0	0	0

Figure 8-5. First three octets of an hypothetical Sync message in the format of a IEEE 1588v2 HPTS message from the Master Clock to the Slave Clocks of our subdomain of interest.

The IEEE 1588v2 HPTS message respects the format presented in Section 8.1. The Subdomain Number field identifies to which subdomain the message is addressed to; only the nodes belonging to that specific subdomain will make usage of this message. The two bits of the Type of Message field indicates this is a Sync Message.
The Sync message will be followed by a Follow_Up message containing the precise sending time of the previous one. The first three octets of the related Follow_Up message are as illustrated in Figure 8-6.

Bits 1	2	3	4	5	6	7	8
0	1	1	0	0	1	0	1
1	0	1	1	0	1	1	1
0	0	0	0	0	0	0	0

Figure 8-6. First three octets of the Follow_Up message which is related to the Sync message illustrated in the previous figure.

Compared with the related message of Figure 8-5, the Subdomain Number is the same, since the two messages are inside the same subdomain. The Type of Message is changed from 01_2 to 10_2, in order to indicate the different nature of the two messages. Finally, the Sequence ID value is exactly the same; this allows the receiving Slave Clocks to correlate this message with the previously received Sync message. Hence the timestamp carried by this Follow_Up message can be inserted together with the timestamp triggered y the Sync message in view if the final offset computation.
When the four nodes belonging to the subdomain 0x65 require a Delay_Resp message from the Master Clock (i.e. the cluster-head node) through a Delay_Req message, the four different Delay_Req messages have to be diversified from each other. Thus each message will combine the Sequence ID received from the Master Clock with its own IEEE 1588v2 HPTS address according to the operation defined above, that is:

(Sequence ID) $\oplus$ (IEEE 1588v2 HTPS address)

According to the scheme for slave-to-master communication illustrated in Section 7.2, a node is allowed to send a message to the Master Clock by preprending it to the time-frame passing through when it is holding the IEEE 1588v2 HPTS Slave-to-Master Token. In turn, the four nodes belonging to subdomain 0x65 will receive the token and, one at a time, send their Delay_Req messages to the Master Clock. The four messages will be as illustrated in Figure 8-7 below.

NODE 2 (IEEE 1588v2 HPTS address: 0x01)

Bits 1	2	3	4	5	6	7	8
0	1	1	0	0	1	0	1
0	0	1	1	0	1	1	1
0	0	0	0	0	0	0	1

NODE 6 (IEEE 1588v2 HPTS address: 0x05)

Bits 1	2	3	4	5	6	7	8
0	1	1	0	0	1	0	1
0	0	1	1	0	1	1	1
0	0	0	0	0	1	0	1

NODE 5 (IEEE 1588v2 HPTS address: 0x04)

Bits 1	2	3	4	5	6	7	8
0	1	1	0	0	1	0	1
0	0	1	1	0	1	1	1
0	0	0	0	0	1	0	0

NODE 8 (IEEE 1588v2 HPTS address: 0x07)

Bits 1	2	3	4	5	6	7	8
0	1	1	0	0	1	0	1
0	0	1	1	0	1	1	1
0	0	0	0	0	1	1	1

Figure 8-7. Four different Delay_Req messages stamped with four different Sequence ID values for unambiguous identification of master-slave chains of messages

As it can be seen, the four Sequence ID fields are different from each others; they are the same Sequence IDs that will be inserted in the four related Delay_Resp messages.
As a result, the Sync message initializes the offset computation for all the Slave Clocks belonging to a particular subdomain, but every Slave Clock brings to completion the timing algorithm through a separate chain of message exchange.

8.4 Processing of the IEEE 1588v2 HPTS message in a node hosting a Slave Clock

After we have seen the format of the IEEE 1588v2 HPTS header in Section 7.1, how Slave-to-Master communication is made possible within the ring in Section 7.2, the format of the IEEE 1588v2 HPTS message in Section 8.1 and how the exchange of messages between the Master Clock and the Slave Clock is distinguished by the others in a unambiguous manner in Section 8.2, we are now finally going to present how a IEEE 1588v2 HPTS message is processed into a node hosting a Slave Clock. That is every node belonging to the ring structure besides the cluster-head node.
Thanks to our proposal of structure for the OpMiGua node, the passing IEEE 1588v3 HPTS messages are duplicated and forwarded in the same time. The duplicate is processed by the "Header processing & IEEE 1588v2 HPTS functions" block. The drawback of such a structure is that all the messages are duplicated and processed, also those addressed to a node belonging to a different subdomain or to a different master-slave chain. Nevertheless, the computation footprint required for the packet analysis is constrained to at most three octets. As foreseen for the format of the IEEE 1588v2 HPTS message (Section 8.1), into the very first three octets the Subdomain Number and Sequance ID value are contained. Only the first or both the fields are enough for a node in order to recognize himself as addressee of the message. In case the two fields do not indicate the processing node as addressee of the IEEE 1588v2 HPTS message, this is discarded and ignored.
Our suggested structure in detail for the "Header processing & IEEE 1588v2 HPTS functions" block, part of the structure presented in Figure 6-6 and recalled along the thesis, is as illustrated in Figure 8-8 below. It recalls the architecture of a common node hosting IEEE 1588 functions (Figure 6-2 and Figure 6-3), with three main differences:

- In our suggested structure the Application Code and the Operating System & Protocol Stack are missing. This is due to the fact that in the "Header processing & IEEE 1588v2 HPTS functions" block only the IEEE 1588v2 HPTS header and the IEEE 1588v2 HPTS messages are forwarded. The IEEE 1588v2 HPTS Code is all is needed for this part of the node.
- The IEEE 1588v2 HPTS Code communicates with the outside, namely with the "I/O packet scheduling" block. In fact, if we look again at the definition of a IEEE 1588v2 HPTS message provided in Subsection 6.5.2, it says: "we define it as a reduced PTP message which can be used for timestamp exchange as part of a synchronization algorithm based and integrated with the principles of IEEE 1588v2 Precision Time Protocol…". It is well specified which they are integrated with the Precision Time Protocol. In fact PTP messages which do not carry or trigger timestamps are exchanged as normal packets according to the OpMiGua HPTS scheme. These packets are not forwarded to the "Header processing and IEEE 1588v2 HPTS functions" block, but exclusively to the "I/O packet scheduling" block. They thus have to be passed from the latter to the former and here processed according to the PTP standardized procedures. This is the reason why we have to different blocks: the "IEEE 1588v2 Code" block and the "IEEE 1588v2 HPTS Code" block. The first answers to PTP standard and deals with Announce (if present), Management and Signaling messages forwarded to it from the "I/O packet scheduling block". The second processes the IEEE 1588v2 HPTS header and the eventual IEEE 1588v2 HPTS messages preprended to the time-frame. Thus we can say that on one hand

the IEEE 1588v2 Code represents the Precision Time Protocol, which are proposal is integrated to; on the other hand we have the IEEE 1588v2 HPTS Code, which is where are proposal resides.

- Once a IEEE 1588v2 HPTS message enters the IEEE 1588v2 HPTS Code block, it can be discarded and ignored based on the directions given by the Identification (ID) block.

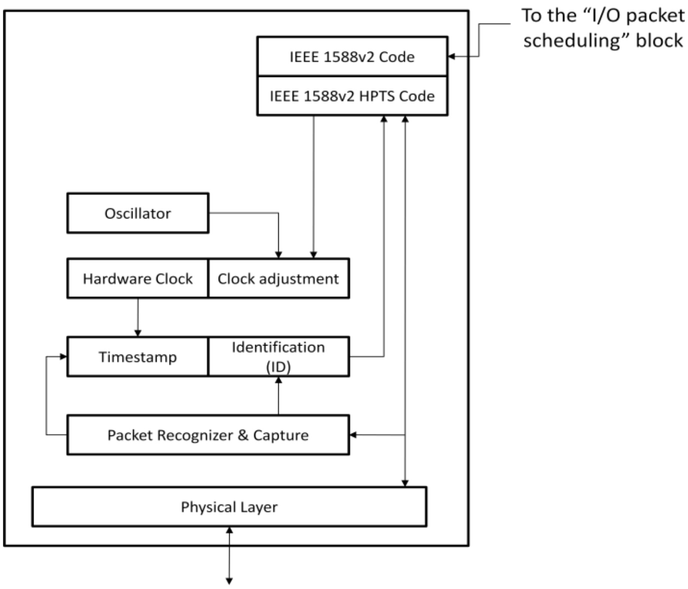


Figure 8-8. Suggested structure for the "Header processing & IEEE 1588v2 HPTS functions" block

The IEEE 1588v2 HPTS header is passed to the IEEE 1588v2 HPTS Code block without being captured by the Packet Recognize & Capture (PRC) block. Eventual IEEE 1588v2 HPTS messages involves also the PRC block and the Identification (ID) block. The complete process is carried out as described beneath:

1. The duplicate of the IEEE 1588v2 HPTS message enters the "Header processing & IEEE 1588v2 HPTS functions" block.
2. Its first three octets are recognized and captured by the PRC block. They contain the Subdomain Number, the Type of Message and the Sequence ID values. This action triggers the generation of a timestamp. Based on the three octets, the instruction to whether process the IEEE 1588v2 HPTS message to the IEEE 1588v2 HPTS Code block or discard it is taken. If the message is to be processed, the three octets are forwarded to the IEEE 1588v2 HPTS Code Block. The decision is taken according to the simple flowchart of Figure 8-9. In case the IEEE 1588v2 HPTS message is processed, together with the octets also the timestamp which has been generated is forwarded. Otherwise it is just ignored.
3. In case the IEEE 1588v2 HPTS message is a Sync message (ToM = 01), the triggered timestamp initializes a new chain of master-slave exchange of messages. In case it has been received either a Follow_Up (ToM = 10) or a Delay_Resp (ToM = 11) message, the carried timestamp is inserted in a master-slave chain already initialized. The timestamps, associated with their own Sequence IDs, are stored into the IEEE 1588v2 HPTS Code.
4. The Source Port Identity is only used as a check by the Master Clock at the moment of the reception of a Delay_Req from one of its Slave Clocks. Into the node hosting the slave Clock, when a Sync message is received, the Source Port Identity value is stored associated with the timestamp and the Sequence ID. After the reception of the correlated Follow_Up message (which will carry the same Source Port Identity), a Delay_Req will be sent to the

Master Clock. This message will contain the same Source Port Identity of the Sync message initializing the master-slave chain. As explained in Section 7.2, it is the same Master Clock that distributes the token for Slave-to-Master communication. It is thus aware of which node has sent the received IEEE 1588v2 HPTS message, nonetheless the Source Port Identity value is used as a further check by the Master Clock. Apart from this, since our proposal is integrated with the Precision Time Protocol, the Source Port Identity value can also be used to relate a received IEEE 1588v2 HPTS message with a already received stander PTP packet. In fact the same field can be found in both the frame formats, giving a platform for interoperability.

5. The timestamp is read and inserted into the timing algorithm.
6. The integrity of the IEEE 1588v2 HPTS message is checked with the CRC 16 field. It runs a checksum over the rest of the IEEE 1588v2 HPTS message, that is the 23 octets previous to the CRC 16 field. The integrity check has to be considered as failed when the checksum run by the receiving node does not match with the value carried into the CRC 16 field. If so, the message is discarded and all the operations implied by the carried or triggered timestamp are voided. In the opposite case of positive check, the operations are carried out by the IEEE 1588v2 HPTS Code.

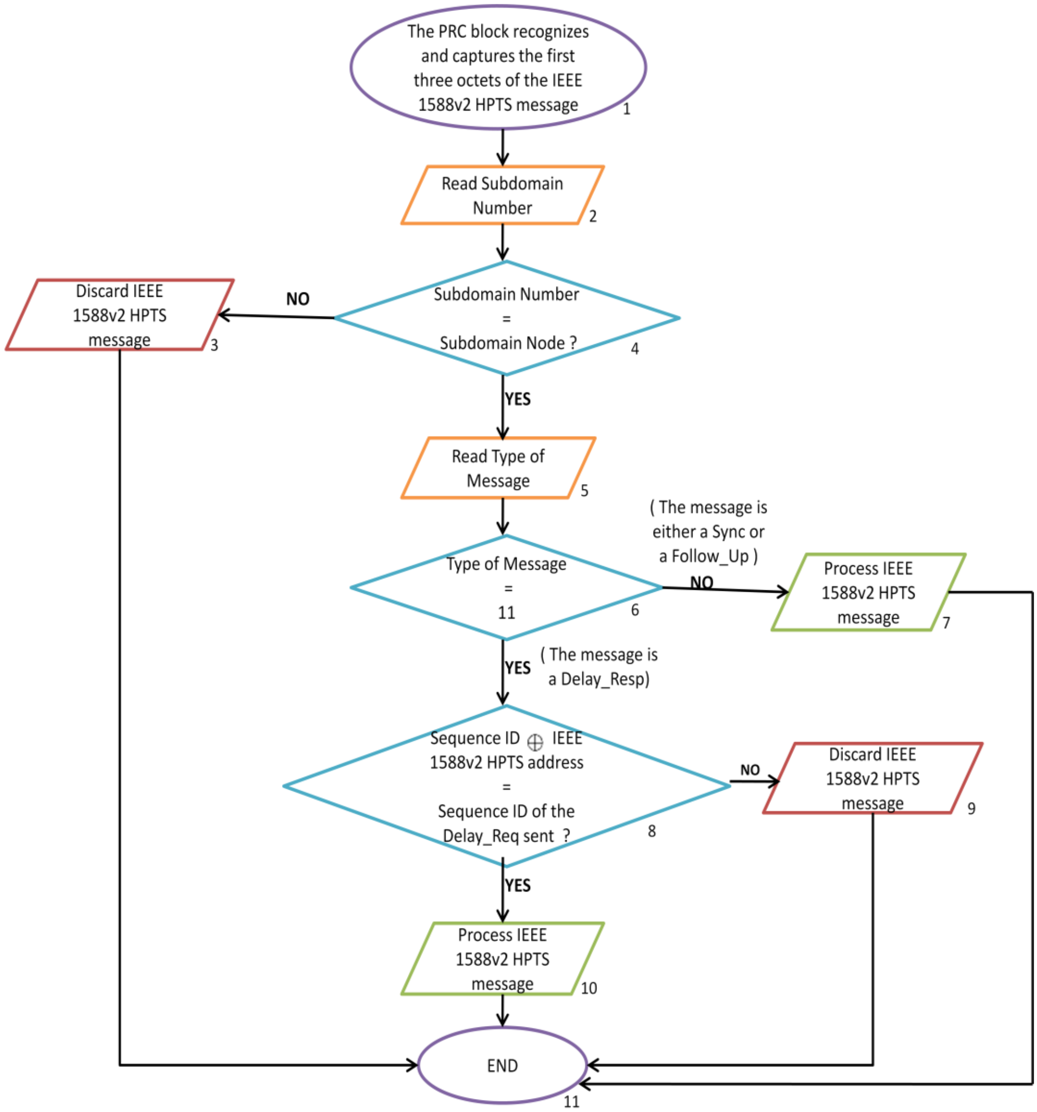


Figure 8-9. Decision algorithm to establish whether the node is the addressee of the IEEE 1588v2 HPTS message or the latter has to be discarded

Chapter 9

Analysis of the proposed IEEE 1588v2 HPTS scheme

9.1 Utilization of resources

In Chapter 4 the description of the format of a PTP message has been provided. In Chapter 7 and Chapter 8 the format of the IEEE 1588v2 HPTS header and of the IEEE 1588v2 HPTS message respectively are presented. We hereafter compare the resource and bandwidth consumptions implied by the two formats.
A PTP message has a header of fixed length equal to 34 octets, while Sync, Follow_Up, Delay_Req and Delay_Resp messages present a body of 20 octets. Hence the header and the body together make up a PTP packet which is 54-octet long for each of the four listed messages. Moreover we assumed as most advantageous solution the utilization of Carrier Ethernet for bearing the packets throughout the mobile backhaul, as well as the Core Network. It implies that each PTP message is mapped over the IEEE 802.3 format; as illustrated in Figure 4-6, a Ethernet header and an Ethernet Frame Check Sequence (FCS) are added before and after the PTP message respectively. The former has a length of 14 octets, the latter of 4 octets, for a total number of additional octets per packet equal to 18. Therefore the final computation says that every PTP message carrying or triggering a timestamp involves a total resource utilization of 72 octets.
In our proposed technique for synchronization among OpMiGua nodes organized into a ring structure, the IEEE 1588v2 HPTS message format is introduced. This format is used to replace the four PTP messages listed above, that are those bearing or triggering the creation of a timestamp. The proposed format has a fixed length of 25 octets and, since the messages are just prepended to the train of time-slots, the encapsulation over a Link Layer protocol is not needed. Only the general IEEE 1588v2 HPTS header is present. Its presence and its length (i.e. 2 octets) are independent from the number of IEEE 1588v2 HPTS messages preprended to the train of time-slots. Moreover, they have to be considered the 25 octets of padding added by the cluster-head node in order to allow Slave clocks to send IEEE 1588v2 HPTS messages to the Master Clock. This scheme has been presented in Section 7.2. The 25 octets can either carry a IEEE 1588v2 HPTS message whose addressee is the cluster-head node or do not bear any meaningful information. The latter case is when the node holding the Stave-to-Master token does not need to send any message to the Master Clock. In both cases, a fixed amount of 25 octets is utilized.
With regard to the other messages foreseen by the IEEE 1588v2 protocol (i.e. Management, Signaling and eventual Announce messages), as stated in Section 6.5 they are exchanged in the standard PTP format. Thus they do not imply changes in the utilization of the available resources.
In the light of the considerations above, putting into comparison the two formats imply an

important difference: with IEEE 1588v2 HPTS messages, the resource utilization is not linearly related to the number of messages transmitted. In fact with the format foreseen by the Precision Time Protocol every message involve a resource utilization of 72 octets. On the other hand, with our proposed format there are some octets that are allocated independently from the presence of messages. Namely, the IEEE 1588v2 HPTS header and the 25 octets of padding. Thus the resource utilization for our proposed synchronization scheme can be summarized by the following formula:

$$N_{octets} = 25 \cdot (n+1) + 2$$

The 2 which is added at the end is due to the IEEE 1588v2 HPTS header. n is the number of IEEE 1588v2 HPTS messages which are sent to the Slave Clocks at the same time (i.e. preprended to the same train of time-slots). A unity is added to n due to the 25 octets of padding. These 25 octets can be considered as efficient payload, in case they carry a Slave-to-Master message, or just as padding octets.
For standard PTP messages the relation between number of messages and resource utilization is easily represented by the following formula:

$$N_{octets} = 72 \cdot n$$

The resource utilization implied by the two formats are compared in Table 9-1. It has to be kept in mind that the proposed IEEE 1588v2 HPTS format is to substitute only the four PTP messages which carry or trigger the creation of a timestamp.

Number of messages (in Master-to-Slave direction)	Resource utilization	
	PTP standard format [in octets]	IEEE 1588v2 HPTS format [in octets]
0	0	27
0+1 in Slave-to-Master direction	72	27
1	72	52
1 +1 in StM direction	144	52
2	144	77
2+1 in StM direction	216	77
3	216	102
3+1 in StM direction	288	102
4	288	127
4+1 in StM direction	360	127

Table 9-1. Comparison between the resource utilization implied by the PTP and the IEEE 1588v2 HPTS scheme for the exchange of timestamps.

In Table 9-1 it can be seen how the presence of a IEEE 1588v2 HPTS message from a Slave Clock to the Master Clock (i.e. the cluster-head node) does not imply a higher resource utilization. On the other hand, even if there are not IEEE 1588v2 HPTS messages to be transmitted,
The key for its interpretation is that, as reminded above, the 25 octets of padding are present independently from the need of the Slave Clock holding the StM token to send a message to the cluster-head node. If a message has to be sent, those octets will bring a meaningful information. But in both cases the 25 octets are allocated.
Only up to four messages (with in addition a possible message in StM direction) have been considered according to what already stated in Section 7.1, that is: it is a realistic assumption to have up to 4 mobile providers (likely from 2 to 3) operating within the same ring. For higher numbers of messages, the two formulas provided above still give the comparison of the two formats.

For the sake of completion we now consider the complete exchange of the four messages necessary for carrying out the synchronization algorithm. If the standard PTP is used, the calculation easily provides a total exchange of 288 octets (i.e. the 72 octets per message mentioned above). In case the IEEE 1588v2 HPTS format is used, the final result is not so straightforward. We take into consideration the worst case: only the Master Clock and a specific node into the ring are exchanging synchronization packets into the format of IEEE 1588v2 HPTS messages. We refer to this as the worst case because the header and the padding octets (i.e. the octets that are allocated independently from the presence of messages) are present exclusively for the exchange of messages of our interest. According to our proposed structure and scheme, several IEEE 1588v2 HPTS messages can be prepended to the train of time-slots taking advantage of the same IEEE 1588v2 HPTS header. In the same time, the 25 octets that are considered of padding in the worst case example are likely to be filled with a IEEE 1588v2 HPTS message whose addressee is the Master Clock. Clarified this, if the Master Clock and one of its Slave Clocks are the only two nodes exchanging timestamps in the format of IEEE 1588v2 HPTS messages, the required number of octets is as follows:

- 52 octets for the Sync message from Master to Slave (2 octets of IEEE 1588v2 HPTS header + 25 octets of the Sync IEEE 1588v2 HPTS message + 25 octets of padding)
- 52 octets for the related Follow_Up message from Master to Slave (made up as the previous)
- 27 octets for the Delay_Req message from Slave to Master (2 octets of IEEE 1588v2 HPTS header + 25 octets of the Delay_Req IEEE 1588v2 HPTS message substituting the 25 octets of padding)
- 52 octets for the Delay_Resp message from Master to Slave (made up as the Sync message)

All together, it gives a result of **183 allocated octets**. Compared with the **288 octets** required by the standard four PTP packets over Ethernet, a saving in resources of 103 octets is achieved. It is equivalent to a saving of the **35.76%** of the standard resource consumption. The average amount of spared resources in terms of octets over a realistic scenario has to be considered higher since a plurality of nodes will be exchanging timestamps with the Master Clock they have in common (i.e. the cluster-head node) at the same time. Thus more timestamps exchanged in the format of IEEE 1588v2 HPTS messages will take advantage of the fixed allocated octets, that are the header and the octets of padding. Thus a global computation over all the nodes will return a lower number of octets per node that have been allocated in order to bring the four-message exchange to completion.

9.2 Accuracy of the synchronization

After an analysis of the saving in resources that our proposed technique implies, we hereafter provide some theoretical numbers on the achievable enhancements upon the accuracy of the synchronization.

The added value of our proposal is that the timestamps travel throughout the ring topology independently from the traffic load. In Section 4.8 the issues affecting the performance of the timing algorithm over a packet network have been presented. It has been highlighted how the Packet Delay Variation (PDV) is the main cause of degradation of performances. In fact this time impairment over the network makes inconsistent the fundamental assumption of the final Master-Slave offset computation. As already mentioned, the assumption states that the delay from slave to master equals the delay from master to slave. The slave-to-master and the master-to-slave delays are both substituted by a common one-way delay. For simplicity of reading, we report hereafter in Table 9-2 the values upon timing impairments (i.e. delay, fluctuations, asymmetry) due to the system components in the end-to-end packet exchange over a Ethernet network.

Quantity	Delay	Fluctuations	Asymmetry
OS	0.1 - 3,000 μsec	0.1 – 3,000 μsec	< 3 msec
PHY	< 50 nsec	< 1 nsec	< 100 nsec
Cable	3 nsec/m	<< 1 nsec	< 0.5 nsec/m
Switch	0.4-3,000 μsec	0.1 – 3,000 μsec	< 3 msec

Table 9-2. Timing impairments introduced by the components of an Ethernet-based system

The fluctuations and asymmetries due to the Operating System (OS) will not be considered, since we discussed in Section 4.4 how the utilization of hardware for timestamp creation avoids its degrading effects.

In Section 4.10 we also analyzed how the master-slave packet delay asymmetry due to time impairments affects the accuracy of the offset computation. Again for simplicity of reading, we report hereafter the obtained results:

- error in the offset calculation after the j^{th} 4-message exchange of timestamps: $\theta(j) = \frac{A(j)}{2}$.
- error in the offset calculation when the Simple Moving Average (SMA) function is assumed to be applied over n calculated offsets : $\theta_{SMA(j)} = \frac{\theta(j)}{n} = \frac{A(j)}{2n}$.
- error in the offset calculation when the Exponential Moving Average (EMA) function is assumed to be applied: $\theta_{EMA(j)} = \alpha \cdot \theta(j) = \alpha \cdot \frac{A(j)}{2}$.

Throughout this analysis section, we will adopt a number of samples n for the SMA equals to $\boldsymbol{n}$**=100** and a smoothing factor α equals to $\boldsymbol{\alpha}$**=0.1** .

After having refreshed the aspect of the timing impairments that has been dealt with along the thesis and in the light of our proposed technique, we can now compare the two solutions.

In line with the considerations presented in Chapter 5, we focus our attention on the cluster-type topology. We can suppose a cluster-type structure consistent with the network dimensioning mentioned in Section 6.1, that is:

- 10 to 15 nodes (i.e. RAN BSs) per ring
- 5 to 10 leaf nodes per node belonging to the ring

Hence we can suppose to use for our analysis and comparison the structure presented in Figure 9-1 below.

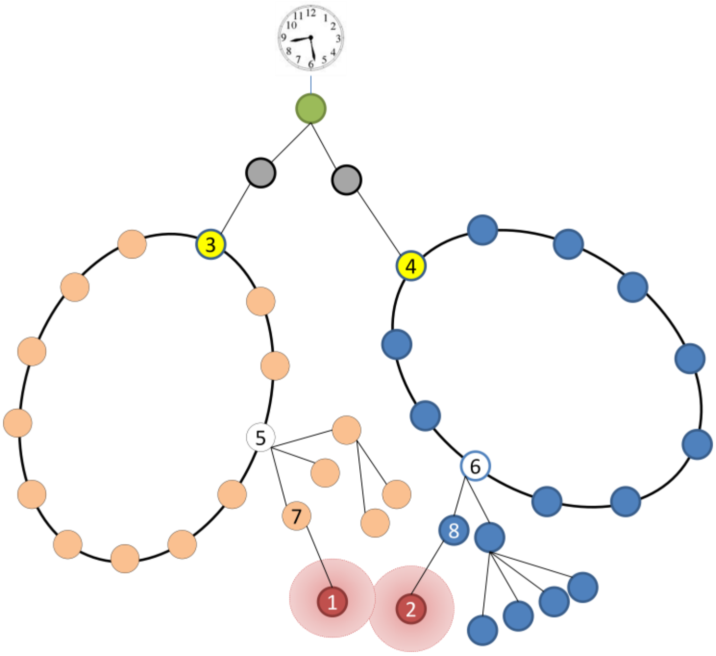


Figure 9-1. Use case scenario for the analysis on the accuracy improvements achievable by means of the IEEE 1588v2 HPTS technique

A cluster-type architecture made of two rings is taken into consideration. The first ring is made up of 12 nodes besides the cluster-head node; on the right, 10 nodes in addition to the cluster-head node are linked together into the ring structure. In each ring, the focus is upon a reduced-size tree structure. Namely, in the left ring the structure beneath the node identified by the number 5 while in the right ring the tree architecture beneath the node number 6. The node identified by the number 1 belongs to the structure beneath node 5, on the other cluster the node 2 belongs to the reduced-size tree topology which originates from node 6. Both nodes refer to a Grandmaster Clock that is hosted into the green node on top. The Grandmaster Clock is located into the Core Network. In order to respect the verisimilitude of our example, an intermediary node between the Grandmaster and the cluster-head node is hypothesized for both rings. Grandmaster Clocks are distributed inside the CN, it is likely that PTP packets have to follow different paths to reach different cluster-head nodes. We have been mentioned in Section 6.2 that intermediary clocks can be present in order to improve the accuracy of synchronization between Grandmaster Clock and cluster-head nodes. We do not take this hypothesis into consideration for our example. With regards to the length of the links, a typical LTE call size is 1 to 5 km; therefore we can assume the links having a length of 10 kms (i.e. two times the cell coverage). This assumptions does not affect the results we will obtain since does not imply significant changes between the traditional PTP scheme and our proposed IEEE 1588v2 HPTS technique.
According to the considerations presented in Section 6.4, the master-slave hierarchy of the considered network will be structured as in Figure 9-2.

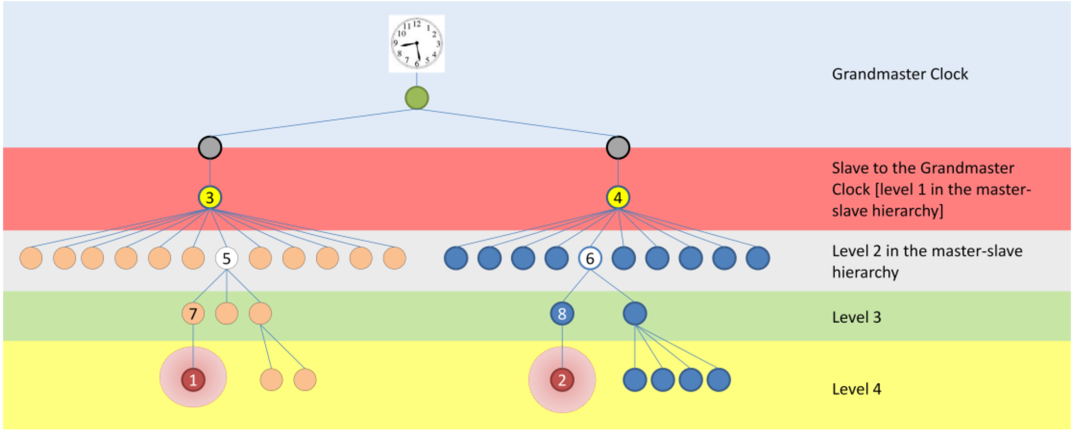


Figure 9-2. Master-Slave hierarchy of the structure presented for the analysis on the accuracy improvements achievable by means of the IEEE 1588v2 HPTS technique

The accuracy of the synchronization between node 1 and node 2 is hereafter analyzed. The two nodes have to be synchronized in case a Mobile Terminal is handed over from one to the other. In the first place the achievable accuracy of synchronization by means of standard PTP technique is taken into consideration. Hence the obtained results are compared with those theoretically achievable with our proposed technique.
In order to have an estimate of the degree of accuracy achievable with standard PTP messages, we have to insert the asymmetries and delay fluctuations reported in Table 9-2 into the network structure presented for our analysis. It has to be reminded that the error is propagated according to a cascade effect for each parent-child level of relationship. As a consequence, in order to calculate a possible offset error between node 1 and the Grandmaster Clock in Figure 9-1, first we have to consider the possible error of cluster-head node 3 compared to the Grandmaster. Hence the error between node 5 and 3, being node 5 the Slave of the Slave of the Grandmaster Clock (Figure 9-2). After this, the error to reach node 7 has to be added. The last step is adding the possible error in offset calculation from node 7 to node 1. The possible inaccuracy between node 1 compared to the Grandmaster Clock will result from the sum of these four errors.
Analogously, the possible error for offset calculation in node 2 compared to the time beaten by the Grandmaster Clock will result from the sum of four different components. That are the error of node 4 compared with the Grandmaster Clock, the error between node 4 and node 6, thus between node 6 and node 8, and finally from node 8 to node 2.
We start on focusing on node 1. The error introduced in synchronization of node 3 with the Grandmaster Clock is caused by possible packet delay asymmetry due to Physical Layer (crossed four different times), Cables (two) or the switching capabilities of the intermediary node. The time impairments are resumed in Figure 9-3.

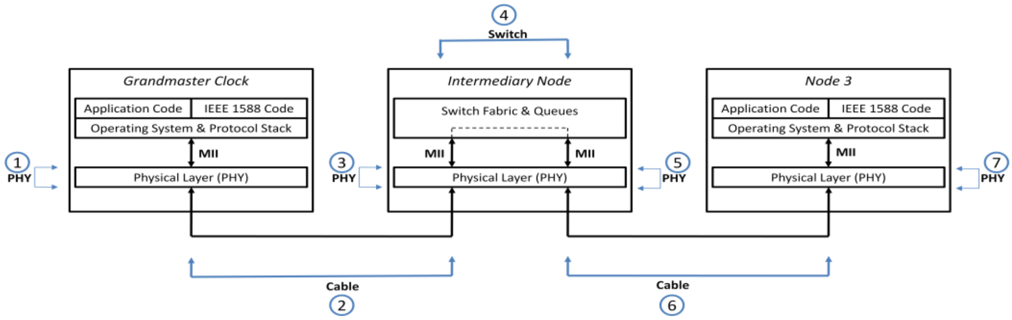


Figure 9-3. Graphical representation of the time impairments introduced by the system components present between the Grandmaster Clock and Node 3.

According to the values reported in Table 9-2, the PTP packets between the Grandmaster Clock and the Node 3 are likely to suffer of the delays, fluctuations and asymmetries reported in Table 9-3 below.

Quantity		Delay	Fluctuations	Asymmetry
1	PHY	< 50 nsec	< 1 nsec	< 100 nsec
2	Cable	30 msec	<< 1 nsec	< 5 msec
3	PHY	< 50 nsec	< 1 nsec	< 100 nsec
4	Switch	0.4-3,000 μsec	0.1 – 3,000 μsec	< 3 msec
5	PHY	< 50 nsec	< 1 nsec	< 100 nsec
6	Cable	30 msec	<< 1 nsec	< 5 msec
7	PHY	< 50 nsec	< 1 nsec	< 100 nsec

Table 9-3. Values of the time impairments introduced by the system components present between the Grandmaster Clock and Node 3

The only considerable fluctuations are due to the presence of the switch. The other fluctuations can be neglected, all together they create a possible delay variation in the order of one nanosecond. The variable delay by the switching capability (i.e. its fluctuations) is the reason behind the asymmetry this introduces into the packet delay between Master and Clock. This asymmetry can be quantified with **up to 3 milliseconds.** Hence focusing on the packet delay asymmetry, besides the presence of the switch we have another important contribution: the asymmetry due to the cables. Nevertheless, it has been mentioned how this asymmetry is added voluntarily in order to reduce the cross-talk effects. It means that this contribution to asymmetry can be easily reduced by means of a simple calibration process. This is valid for the case of Ethernet cables in an electrical domain which we are taking into consideration in this example (10/100/1000/10GBASE-T Ethernet); if we took into consideration the optical domain, asymmetry into the cables would arise for chromatic dispersion. Either ways, calibration process and network engineering are enough to reduce noticeably the effects. Going back to the study case of our example, a contribution to asymmetry in the order of **up to 1 millisecond** can be ascribed to the two cables together. Moreover we add **up to 100 nanoseconds** due to the Physical Layers all together.
Summing up, a packed delay asymmetry **up to 4.1 milliseconds** can affect the synchronization process of Node 3 with the Grandmaster Clock. We are going to indicate the packet delay asymmetry in this line of the network as $\boldsymbol{A_{GM-3}}$. According to the results of Section 4.10 refreshed at the beginning of this current Section, A_{GM-3} will cause an error in the offset

calculation equal to $\theta_{GM-3} = \frac{A_{GM-3}}{2}$. This error:

- if inserted into a SMA function, it affects the accuracy of the offset calculation by the quantity $\boldsymbol{\theta_{SMA,GM-3}} = \frac{\theta_{GM-3}}{n} = \frac{A_{GM-3}}{2n} = \frac{4.1\ msec}{2\cdot 100} \approx \mathbf{20\ \mu sec}$
- if inserted into a EMA function, it affects the accuracy of the offset calculation by the quantity $\boldsymbol{\theta_{EMA,GM-3}} = \alpha \cdot \theta_{GM-3} = \alpha \cdot \frac{A_{GM-3}}{2} = 0.1 \cdot \frac{4.1\ msec}{2} \approx \mathbf{200\ \mu sec}$.

So far the calculations have been carried out considering as valid the worst values of asymmetry. We can repeat the calculations with better values of asymmetries. With a good calibration process, the asymmetry due to the cables can be further reduced: we can assume a contribution of **1 microsecond** by both the cables. With regards to the switching component, its fluctuations in delay are due to the traffic load. If we reasonably assume that the traffic load does not change sensibly between the moment in which the master-to-slave and the slave-to-master packets pass through the switch, it is hypothesizable that an asymmetry of about **100 microseconds** is caused by the switch. Taking these values as good and adding the **100 nanoseconds** by the PHYs all together, we have a total packet delay asymmetry of $\overline{\boldsymbol{A_{GM-3}}}$ =**100.2 microseconds.** Such an asymmetry into the master-slave path will cause an error in the offset calculation equal to $\overline{\theta_{GM-3}} = \frac{\overline{A_{GM-3}}}{2}$ which:

- if inserted into a SMA function, it affects the accuracy of the offset calculation by the quantity $\overline{\boldsymbol{\theta_{SMA,GM-3}}} = \frac{\overline{\theta_{GM-3}}}{n} = \frac{\overline{A_{GM-3}}}{2n} = \frac{101.1\ \mu sec}{2\cdot 100} \approx \mathbf{0.5\ \mu sec}$
- if inserted into a EMA function, it affects the accuracy of the offset calculation by the quantity $\overline{\boldsymbol{\theta_{EMA,GM-3}}} = \alpha \cdot \overline{\theta_{GM-3}} = \alpha \cdot \frac{\overline{A_{GM-3}}}{2} = 0.1 \cdot \frac{101.1\ \mu sec}{2} \approx \mathbf{5\ \mu sec}$.

Following the same criteria on the other lines of the network, we can derive the theoretical values for the node 5 synchronizing with the node 3, according to Figure 9-4.

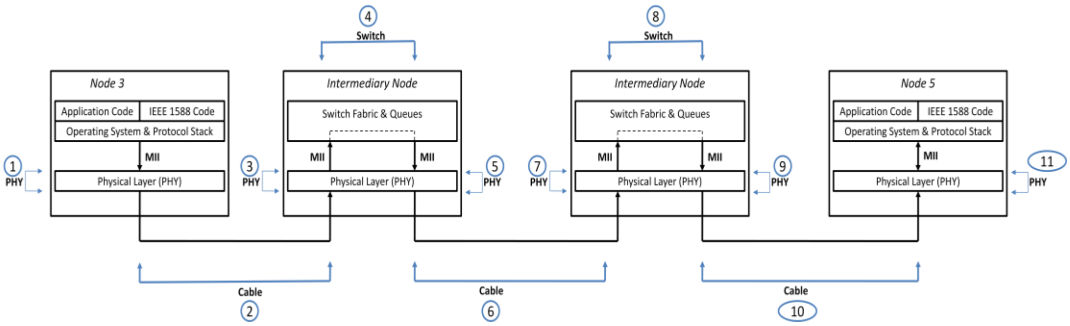


Figure 9-4. Graphical representation of the time impairments introduced by the system components present between the Node 3 and Node 5

The time impairments of Table 9-4 are thus obtained.

Quantity		Delay	Fluctuations	Asymmetry
1	PHY	< 50 nsec	< 1 nsec	< 100 nsec
2	Cable	30 msec	<< 1 nsec	< 5 msec
3	PHY	< 50 nsec	< 1 nsec	< 100 nsec
4	Switch	0.4-3,000 µsec	0.1 – 3,000 µsec	< 3 msec
5	PHY	< 50 nsec	< 1 nsec	< 100 nsec
6	Cable	30 msec	<< 1 nsec	< 5 msec
7	PHY	< 50 nsec	< 1 nsec	< 100 nsec
8	Switch	0.4-3,000 µsec	0.1 – 3,000 µsec	< 3 msec
9	PHY	< 50 nsec	< 1 nsec	< 100 nsec
10	Cable	30 msec	<< 1 nsec	< 5 msec
11	PHY	< 50 nsec	< 1 nsec	< 100 nsec

Table 9-4. Values of the time impairments introduced by the system components present between the Grandmaster Node 3 and Node 5

Proceeding as above, we can consider a worst case (the maximum asymmetries in delays) and hypothesize a situation where better values in asymmetries are experienced. For the worst case we have a contribution of **3 milliseconds** by each switch, **1.5 milliseconds** by the cables and about **150 nanosecond** by the PHYs. They can be reasonably downsized to a contribution of **150 microseconds** by each switch, **1 microsecond** by the cables and **150 nanoseconds** by the PHYs. Compared with the values hypothesized above, the contribution by each switch is supposed higher since two intermediary nodes have to be crossed and thus the traffic load on each switch can experience wider variation, since it takes place over a wider gap of time.
Said so, the following values for the synchronization of Node 5 to Node 3 are obtained:

- $\boldsymbol{\theta_{SMA,3-5}} = \frac{\theta_{3-5}}{n} = \frac{A_{3-3}}{2n} = \frac{7.65\ msec}{2 \cdot 100} \approx \mathbf{38\ \mu sec}$
- $\boldsymbol{\theta_{EMA,3-5}} = \alpha \cdot \theta_{3-5} = \alpha \cdot \frac{A_{3-5}}{2} = 0.1 \cdot \frac{7.65\ msec}{2} \approx \mathbf{380\ \mu sec}$
- $\overline{\boldsymbol{\theta_{SMA,3-5}}} = \frac{\overline{\theta_{3-5}}}{n} = \frac{\overline{A_{3-5}}}{2n} = \frac{301.15\ \mu sec}{2 \cdot 100} \approx \mathbf{1.5\ \mu sec}$
- $\overline{\boldsymbol{\theta_{EMA,3-5}}} = \alpha \cdot \overline{\theta_{3-5}} = \alpha \cdot \frac{\overline{A_{3-5}}}{2} = 0.1 \cdot \frac{301.15\ \mu sec}{2} \approx \mathbf{15\ \mu sec}$

Finally, in both the lines from Node 5 to Node 7 and, in turn, from Node 7 to Node 1 no intermediary nodes are present. Thus the time impairments will result as illustrated in Figure 9-5 for both cases.

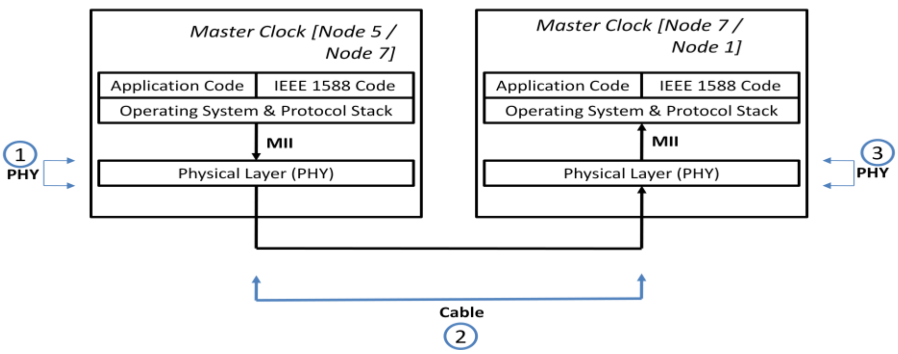


Figure 9-5. Graphical representation of the time impairments introduced by the system components present between Node 5 and Node 7 and between Node 7 and Node 1.

Therefore the time impairments of Table 9-5 have to be considered in these two cases.

Quantity		Delay	Fluctuations	Asymmetry
1	PHY	< 50 nsec	< 1 nsec	< 100 nsec
2	Cable	30 msec	<< 1 nsec	< 5 msec
3	PHY	< 50 nsec	< 1 nsec	< 100 nsec

Table 9-5. Values of the time impairments introduced by the system components present between Node 5 and Node 7 and between Node 7 and Node 1.

As a result, the following values can be obtained:

- $\boldsymbol{\theta_{SMA,5-7}} = \boldsymbol{\theta_{SMA,7-1}} = \frac{\theta_{7-1}}{n} = \frac{A_{7-1}}{2n} = \frac{1\ msec}{2\cdot 100} \approx \mathbf{5}\ \boldsymbol{\mu sec}$
- $\boldsymbol{\theta_{EMA,5-7}} = \boldsymbol{\theta_{EMA,7-1}} = \alpha \cdot \theta_{7-1} = \alpha \cdot \frac{A_{7-1}}{2} = 0.1 \cdot \frac{1\ msec}{2} \approx \mathbf{50}\ \boldsymbol{\mu sec}$
- $\overline{\boldsymbol{\theta_{SMA,5-7}}} = \overline{\boldsymbol{\theta_{SMA,7-1}}} = \frac{\overline{\theta_{7-1}}}{n} = \frac{\overline{A_{7-1}}}{2n} = \frac{1.1\ \mu sec}{2\cdot 100} \approx \mathbf{5}\ \boldsymbol{nsec}$
- $\overline{\boldsymbol{\theta_{EMA,5-7}}} = \overline{\boldsymbol{\theta_{EMA,7-1}}} = \alpha \cdot \overline{\theta_{7-1}} = \alpha \cdot \frac{\overline{A_{7-1}}}{2} = 0.1 \cdot \frac{1.1\ \mu sec}{2} \approx \mathbf{50}\ \boldsymbol{nsec}$

Due to the mentioned cascade effect for each parent-child relationship, in order to obtain a theoretical value for achievable accuracy of synchronization of Node 1 to the Grandmaster Clock we have to sum together the possible errors for each analyzed portion of the network. As final result we obtain the values listed below:

- $\boldsymbol{\theta_{SMA,GM-1}} = \boldsymbol{\theta_{SMA,GM-3}} + \boldsymbol{\theta_{SMA,3-5}} + \boldsymbol{\theta_{SMA,5-7}} + \boldsymbol{\theta_{SMA,7-1}} = (\mathbf{20} + \mathbf{38} + \mathbf{5} +$ ***5 msec=68 msec***
- $\boldsymbol{\theta_{EMA,GM-1}} = \boldsymbol{\theta_{EMA,GM-3}} + \boldsymbol{\theta_{EMA,3-5}} + \boldsymbol{\theta_{EMA,5-7}} + \boldsymbol{\theta_{EMA,7-1}} = (\mathbf{200} + \mathbf{380} + \mathbf{50} +$ ***50 msec=680 msec***
- $\overline{\boldsymbol{\theta_{SMA,GM-1}}} = \overline{\boldsymbol{\theta_{SMA,GM-3}}} + \overline{\boldsymbol{\theta_{SMA,3-5}}} + \overline{\boldsymbol{\theta_{SMA,5-7}}} + \overline{\boldsymbol{\theta_{SMA,7-1}}} = \mathbf{500}\ \boldsymbol{nsec} + \mathbf{1.5}\ \boldsymbol{\mu sec} + \mathbf{5}\ \boldsymbol{nsec} + \mathbf{5}\ \boldsymbol{nsec} = \mathbf{2.01}\ \boldsymbol{\mu sec}$
- $\overline{\boldsymbol{\theta_{EMA,GM-1}}} = \overline{\boldsymbol{\theta_{EMA,GM-3}}} + \overline{\boldsymbol{\theta_{EMA,3-5}}} + \overline{\boldsymbol{\theta_{EMA,5-7}}} + \overline{\boldsymbol{\theta_{EMA,7-1}}} = \mathbf{5}\ \boldsymbol{\mu sec} + \mathbf{15}\ \boldsymbol{\mu sec} + \mathbf{50}\ \boldsymbol{nsec} + \mathbf{50}\ \boldsymbol{nsec} = \mathbf{20.1}\ \boldsymbol{\mu sec}$

Here above we have the derived theoretical values of accuracy achievable by means of the standard Precision Time Protocol. With $\boldsymbol{\theta_{SMA,GM-1}}$ is indicated the value calculated with the widest asymmetries into the four master-slave portions of the network and the employment of the Simple Moving Average function. In case the Exponential Moving Average function is supposed to be used as averaging algorithm instead of the SMA, the value named as $\boldsymbol{\theta_{SMA,GM-1}}$ is obtained. The values under the names of $\overline{\boldsymbol{\theta_{SMA,GM-1}}}$ and $\overline{\boldsymbol{\theta_{EMA,GM-1}}}$ are those obtained in case of reasonably better values for master-slave asymmetry in case the SMA or the EMA functions, respectively, are used as averaging algorithm.

Using these theoretically obtained values as a platform, we can provide an estimation of the enhancements brought by our proposal. As already mentioned, the fundamental added value of our proposed IEEE 1588v2 HPTS technique is that by prepending the timestamps to the time-frames and passing them transparently from node to node into the ring, they are made independent from the traffic load. Which means that the contribution to packed delay asymmetry by the switching components can be theoretically considered negligible in its entirely. Some small fluctuations in delays might be caused by the internal structure of the OpMiGua node, but at the current state-of-the-art of the OpMiGua concept those fluctuations are not quantifiable. We can consider some possible small variations keeping for example the asymmetry caused by the MII. This interface will not be traversed into the OpMiGua node since the timestamps are not carried as Ethernet frames. Maintaining this cause of packet delay asymmetry we can hypothesize some fluctuations due to the internal structure of the OpMiGua node.

Therefore the accuracy obtained thanks to the adoption of our proposed technique can be derived by repeating the calculations carried out above but this time neglecting the contribution in packet delay asymmetry due to the presence of switching capabilities into the ring. Which means the in the portion of network from Node 3 to Node 5 the situation is as illustrated in Figure 9-6.

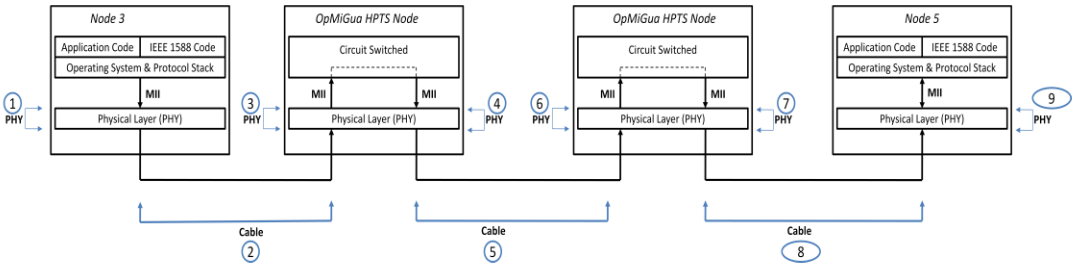


Figure 9-6. Graphical representation of the time impairments introduced by the system components present between the Node 3 and Node 5 when the circuit-switch capable OpMiGua HPTS nodes are exploited into the ring structure

As a consequence, the following possible errors in the offset calculation are derived:

- In case the maximum asymmetries in delays are considered (i.e. the worst case):
  - $\boldsymbol{\theta^{hpts}_{SMA,3-5}} = \frac{\theta^{hpts}_{3-5}}{n} = \frac{A^{hpts}_{3-5}}{2n} = \frac{1.5\ msec}{2 \cdot 100} \approx \boldsymbol{7.5\ \mu sec}$
  - $\boldsymbol{\theta^{hpts}_{EMA,3-5}} = \alpha \cdot \theta^{hpts}_{3-5} = \alpha \cdot \frac{A^{hpts}_{3-5}}{2} = 0.1 \cdot \frac{1.5\ msec}{2} \approx \boldsymbol{75\ \mu sec}$

  With the superscript $*^{hpts}$ we diversify the error in offset calculation when the proposed IEEE 1588v2 HPTS technique is employed.
- In case the contribution of the system components to master-slave packet delay asymmetry are downsized (i.e. better calibration to counteract the asymmetries in cables):
  - $\overline{\boldsymbol{\theta^{hpts}_{SMA,3-5}}} = \frac{\overline{\theta^{hpts}_{3-5}}}{n} = \frac{\overline{A^{hpts}_{3-5}}}{2n} = \frac{151.15\ \mu sec}{2 \cdot 100} \approx \boldsymbol{755\ nsec}$
  - $\overline{\boldsymbol{\theta^{hpts}_{EMA,3-5}}} = \alpha \cdot \overline{\theta^{hpts}_{3-5}} = \alpha \cdot \frac{\overline{A^{hpts}_{3-5}}}{2} = 0.1 \cdot \frac{151.15\ \mu sec}{2} \approx \boldsymbol{7.55\ \mu sec}$

We can finally insert these values into the final computation in order to the theoretically achievable accuracy between Node 1 and the Grandmaster Clock:

- $\boldsymbol{\theta^{hpts}_{SMA,GM-1} = \theta_{SMA,GM-3} + \theta^{hpts}_{SMA,3-5} + \theta_{SMA,5-7} + \theta_{SMA,7-1} = 20\ msec + 7.5\ \mu sec + 5\ msec + 5\ msec = 30.0075\ msec}$
- $\boldsymbol{\theta^{hpts}_{EMA,GM-1} = \theta_{EMA,GM-3} + \theta^{hpts}_{EMA,3-5} + \theta_{EMA,5-7} + \theta_{EMA,7-1} = 200\ msec + 75\ \mu sec + 50\ msec + 50\ msec = 300.075\ msec}$
- $\boldsymbol{\overline{\theta^{hpts}_{SMA,GM-1}} = \overline{\theta_{SMA,GM-3}} + \overline{\theta^{hpts}_{SMA,3-5}} + \overline{\theta_{SMA,5-7}} + \overline{\theta_{SMA,7-1}} = 500\ nsec + 755\ nsec + 5\ nsec + 5\ nsec = 1.265\ \mu sec}$
- $\boldsymbol{\overline{\theta^{hpts}_{EMA,GM-1}} = \overline{\theta_{EMA,GM-3}} + \overline{\theta^{hpts}_{EMA,3-5}} + \overline{\theta_{EMA,5-7}} + \overline{\theta_{EMA,7-1}} = 5\ \mu sec + 7.55\ \mu sec + 50\ nsec + 50\ nsec = 12.65\ \mu sec}$

These values have to be compared with those obtained without exploitation of OpMiGua HPTS nodes into the ring structures of the clusters. The two achieved theoretical results are compared in Table 9-6.

Achievable accuracy between Grandmaster Clock and Node 1		
	PTP	IEEE 1588v2 HPTS
SMA algorithm	$\overline{\boldsymbol{\theta}_{SMA,GM-1}} = \mathbf{2.01}\ \boldsymbol{\mu sec}$	$\overline{\boldsymbol{\theta}^{hpts}_{SMA,GM-1}} = \mathbf{1.265}\ \boldsymbol{\mu sec}$
EMA algorithm	$\overline{\boldsymbol{\theta}_{EMA,GM-1}} = \mathbf{20.1}\ \boldsymbol{\mu sec}$	$\overline{\boldsymbol{\theta}^{hpts}_{EMA,GM-1}} = \mathbf{12.65}\ \boldsymbol{\mu sec}$

Table 9-6. Comparison between the theoretical achievable results by adopting the proposed IEEE 1588v2 HPTS scheme instead of the standard PTP.

Chapter 10

Conclusions and further works

10.1 Conclusions

Mobile backhauls are undergoing a process of major changes in order to be able to serve the dramatic increase of traffic over the air. With the adoption of packet-based backhauls instead of TDM-based, an important aspect such as synchronization among RAN BSs becomes challenging.
Along this thesis a new **synchronization** technique has been presented for mobile backhauls with exploitation of the OpMiGua Hybrid Packet/Time Slotted Circuit Switched Scheme (HPTS). The cluster-type network topology has been chosen as specific physical configuration since more suitable for the Next Generation Mobile Network (NGMN), i.e. the LTE mobile technology.
The proposed technique has been given the name of **IEEE 1588v2 HPTS scheme**. The name reflects the nature of the proposal: a technique based on and integrated with the successful protocol IEEE 1588v2 (aka PTP, Precision Time Protocol) which takes advantage of the hybrid nature of the OpMiGua HPTS concept. Two important enhancements are brought: the Master-Slave hierarchy is made simpler and the exchange of timestamps is made independent from the traffic load.
The Master-Slave hierarchy foresees the cluster-head node to be elected as Master Clock for all the nodes belonging to the ring beneath, independently from their subdomain. By means of this hierarchical structure, together with the newly proposed manner of distributing and carrying timestamps within the ring, all the nodes linked together into the ring architecture reach a more uniform and easily manageable synchronization.
With regard to the exchange of timestamps through the network, they are prepended by the cluster-head node to the time-frames, whose utilization is implied by the OpMiGua HPTS scheme. A possible structure for the OpMiGua node is suggested. By data duplication and taking advantage of the hybrid circuit-switching capabilities of the OpMiGua node, the timestamps preprended to the time frame are passed transparently from node to node and read in the same time. This yields to a fixed end-to-end delay for the timestamps. A new format is defined for messages carrying timestamps necessary for the synchronization and the name of “**IEEE 1588v2 HPTS message**” assigned to it. It has a fixed length which is sensibly shorter than the one of standard PTP messages, since the mapping over a Link Layer protocol (i.e. Ethernet) is not necessary anymore. Also a header to the whole time-frame is defined and given the name of “**IEEE 1588v2 HPTS header**”. It contains the number of IEEE 1588v2 HPTS messages prepended to the time-frame and distributes a virtual token (named “**IEEE 1588v2 HPTS StM token**”) which allows the Slave-to-Master sending of timestamps. The transmission and transport of timestamps to the Master Clock (i.e. the cluster-head node) is again carried out transparently and free of Packet Delay Variation.
Among the nodes belonging to the backhaul but not to a ring structure, the timestamps are exchanged according to the standard IEEE 1588v2 protocol. Within the same ring topology, fundamental element of the considered cluster-type physical configuration, PTP messages not containing timestamps (i.e. management and signaling) are exchanged as normal PTP packets over Ethernet framing.
Some theoretical values upon the accuracy achievable by means of our proposed technique, compared with the standard PTP scheme, are provided into the enclosing analysis part. It is

highlighted how the lack of timing impairments due to switching components throughout the packet network implies a considerably improved accuracy. Making the timestamps independent form the traffic load allows to guarantee a fixed end-to-end delay to the timestamps between the Master and the Slave Clocks. It translates into a better accuracy, the ultimate goal of a synchronization technique. This is especially valuable in the scenario of mobile backhauls, where the requirements upon synchronization are very strict.

10.2 Further Works

The technique proposed can be used for further studies in several aspects. A simulation of the proposed backhaul would be the natural introducing step to an experimental verification. The latter would be a prototype of a OpMiGua HPTS node which exploits the proposed and analyzed scheme. A theoretical approach has been used along the thesis, fused together with the numerous studies already carried out upon the standard PTP scheme.
A simulation would also lead to a comparison between the proposed technique and the already existing alternatives to packet-layer based approach. At the current state-of-art, other technologies are the GPS and the Carrier Ethernet. Since the former is well define and extremely accurate, the comparison should be focused on the trade-off between costs saving and achievable accuracies. Many aspects of the latter cab be investigated and compared.
Finally, the proposed technique of prepending a header to the time-frame into the specific physical configuration of the ring structure can be exploited for other aims besides the only synchronization. More features can be offered by proposing a more complete header.

Bibliography

1. F. Ivanek, "Mobile Backhaul, from the Guest Editor's Desk", IEEE Microwave Magazine vol. August 2009, pp. 10-20
2. M. Ritter, "Mobile Backhaul Evolution, White Paper", ADVA Optical Networking, October 2009
3. S. Chia, M. Gasparroni, P. Brick, "The Next Challenge for Cellular Networks: Backhaul", IEEE Microwave Magazine vol. August 2009, pp. 54-66
4. "Cisco Visual Networking Index: Global Mobile Data Forecast Update", Cisco Public Information, January 29th, 2009
5. NEC Corporation, "Mobile Backhaul Evolution, for deploying Mobile Next Generation Networks, White Paper", February 2007
6. NGNM Alliance, "Next Generation Mobile Networks Optimised Backhaul Requirements", August 2008
7. Metro Ethernet Forum, "MEF 22. Mobile Backhaul Implementation Agreement, Phase 1", January 2009
8. S. Bjørnstad, D. R. Hjelme and oth. "The OpMiGua project – Final report", Telenor R&I Research Report, January 2007
9. S. Bjørnstad, D. R. Hjelme, N. Stol, T. Olsen "Optical Packet Switched Migration Capable network with Guaranteed Services (OpMiGua) – a concept description", Project Internal Report Telenor R&I, June 2004
10. S. Bjørnstad, A. Kimsas, "Hybrid Packet/Time Slotted circuit switched scheme (HPTS)", 10th Anniversary International Conference on Transparent Optical Networks (ICTON), 2008
11. Symmetricon, White Paper, "Deployment of Precision Time Protocol for Synchronization of GSM and UMTS Basestations" (01/2008), retrieved at http://www.chronos.co.uk/pdfs/tel/symmetricom/Deployment_of_Precision_Time_Protocol_GSM_UMTS.jun08.pdf
12. ITU-T Recommendations G.8261/Y.1361 (04/2008), "Timing and synchronization aspects in packet network"
13. Institute of Electrical and Electronics Engineer, IEEE Std 1588-2008 "Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems", July 2008
14. www.metroethernetforum.org
15. V. L. Tuft, S. Bjørnstad, D. R. Hjelme "Time Interleaved Polarization Multiplexing for Polarization Labeling", Proceedings of the 7th International Conference on Transparent Optical Networks (ICTON), July 2005
16. F. Puleio, "TTM4511 Specialization Project – Use of an OpMiGua hybrid network for grid networking", NTNU, December 2009
17. ITU-T Recommendations G.823 (03/2000), "The control of jitter and wander within digital networks which are based on the 2048 kbit/s hierarchy"
18. ITU-T Recommendations G.824 (03/2000), "The control of jitter and wander within digital networks which are based on the 1544 kbit/s hierarchy"
19. ITU-T Recommendations G.825 (03/2000), "The control of jitter and wander within digital networks which are based on the synchronous digital hierarchy (SDH)"
20. ITU-T Recommendations G.8262/Y.1362 (08/2007), "Timing characteristics of synchronous Ethernet equipment slave clock (EEC)"
21. ITU-T Recommendations G.8264/Y.1364 (10/2008), "Distribution of timing information through packet networks"
22. Institute of Electrical and Electronics Engineer, IEEE Std 1588 "Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems",

November 2002

23. J. C. Eidson, "Measurement, Control, and Communication Using IEEE 1588", Springer Editions, January 2006
24. L. Hiley, "Mobile service, costs and a green agenda", Article no.12 of Connect-Word Asia-Pacific I 2010, retrievable from http://www.connect-world.com/articles/recent_article.php?oid=AP_I_2010_12
25. ARINC Research Corporation, "ICD-GPS-200 NavStar GPS Space Segment/Navigation User Interfaces", October 1993
26. TeliaSonera (2010) . "TeliaSonera has selected 4G vendors". Retrieved on May 2010 from http://www.teliasonera.com/News-and-Archive/Press-releases/2010/TeliaSonera-has-selected-4G-vendors/
27. S. Sesia, I. Toufik, M. Baker, "LTE, the UMTS Long Term Evolution. From Theory to Practice", Wiley Editions, 2009, 648 pp.
28. J. Walfisch and H. L. Bertoni, "Theoretical Model of UHF Propagation in Urban Environments", IEEE Trans. Ant. Prop. vol. 36, no. 12, Dec. 1988, pp. 1788-1796
29. T.Otsu, N.Umeda, Y.Yamao, "System architecture for mobile communications systems beyond IMT-2000", GLOBECOM'01, vol.1, pp.538-452, 2001.
30. T.Otsu, N.Umeda, Y.Yamao, I.Otsu, "Network Architecture for Mobile Communications Systems Beyond IMT-2000", NNT DoCoMo, pp.31-37
31. S. Little "Is Microwave Backhal Up To The 4G Task?", IEEE Microwave Magazine vol. August 2009, pp. 67-74
32. S. Chia, M. Gasparroni, P. Brick, "The Next Challenge for Cellular Networks: Backhaul", IEEE Microwave Magazine vol. August 2009, pp. 54-66
33. Z. Becvar, J. Zelenka, "Handovers in the Mobile WiMax", Research in Telecommunication Technology 2006, Brno, 2006
34. D. Wong, T. J. Lim, "Soft Handoffs in CDMA Mobile Systems", IEEE Personal Communications, Vol. 4 Issue 6, December 1997
35. A.Krendzel, Y.Koucheryavy, J.Hariu, "Cost and Reliability Estimation of Radio Access Network Structures for 4G Systems", Tamper University of Technology, pp.2297-2301
36. L. Ruan, D. Du, "Optical Networks – Recent Advances (Network Theory and Application)", Kluwer Academic Publishers, September 2001
37. G.Ellinas, K.Bala, G.Chang, "A Novel Wavelength Assignment Algorithm for 4-fiber WDM Self-Healing Rings", IEEE International Conference on Communications, 1998, Conference Record vol. 1, pp. 197-201
38. X.Lu, S.He, "Wavelength Assignment for WDM ring", IEEE Electronic Letters vol. 39 no. 19, September 2003
39. S. Subramaniam, M. Azizoglu, K. Somani, "All-optical networks with sparse wavelength conversion", IEEE/ACM Trans. Networking vol. 4, pp. 544-557, August 1996
40. S. Arora and S. Subramaniam, "Converter placement in wavelength routing mesh topologies," in *Proc. IEEE ICC*, June 2000, pp.1282–1288
41. H. Harai, M. Murata, H. Myahara, "Heuristic algorithms of allocation of wavelength convertible nodes and routing coordination in all-optical networks," *IEEE/OSA J. Lightwave Technol.*, vol. 17, pp. 535–545, Apr. 1999
42. L. Li and A. K. Somani, "Efficient algorithms for wavelength converter placement," *SPIE/Kluwer Opt. Networks Mag.*, vol. 3, no. 2, pp. 54–62, Mar./Apr. 1999
43. "On the optimal placement of wavelength converters in wavelength-routed networks," in *Proc. IEEE INFOCOM'98*, Apr. 1998, pp. 902–909.