

# **CLOCK SYNCHRONIZATION WITHIN DATACENTER**

Tanjila Ahmed  
Friday Group Meeting.

# OUTLINE

- ❑ Packet Switched Optical Network (PSON) Architecture
- ❑ Why synchronization in Datacenter?
- ❑ Goal of PSON Project
- ❑ Existing Time Synchronization Protocol
- ❑ Precision Time Protocol
- ❑ PTP Mechanism
- ❑ Proposed Protocol
- ❑ Simulation parameters
- ❑ Expected Results

# PSON Architecture

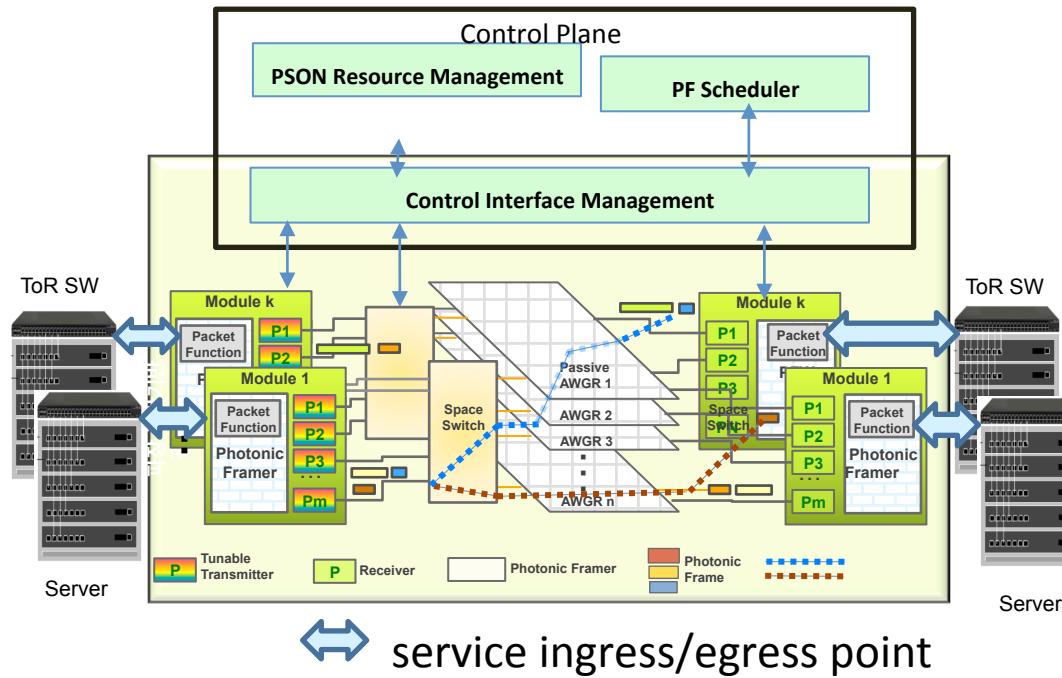
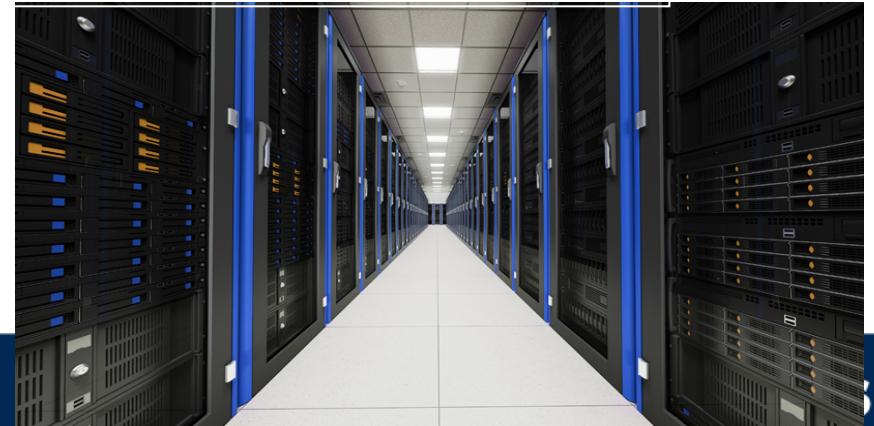


Fig 1: Packet Switched Optical Network Architecture

# Why Synchronization in Datacenter?

1. Cloud based high performance computing
2. Real time data analysis
3. Financial services industry e.g. high frequency trading
4. Server clustering
5. Web services requirements



# Goal of PSON Project

1. Synchronizing the TOR switches (servers)
2. High precision (sub-us, nano second)
3. MAC/Upper layer protocol
4. No overhead
5. Synchronization bits within the size of the frame(no additional bits for it)
6. Controller managed synchronization

# Existing Protocols

Network Time Protocol →  
Precision Time Protocol →  
Global Positioning System →  
Datacenter Time Protocol →  
[1]

	Precision	Scalability	Overhead (pckts)	Extra hardware
NTP	us	Good	Moderate	None
PTP	sub-us	Good	Moderate	PTP-enabled devices
GPS	ns	Bad	None	Timing signal receivers, cables
DTP	ns	Good	None	DTP-enabled devices

Table 1: Comparison between NTP, PTP, GPS, and DTP

[1] K.S. Lee et al., “Globally Synchronized Time via Datacenter Networks”, Proceedings of the 2016 conference on ACM SIGCOMM 2016 Conference, pp. 454-467, 2016.

# Precision Time Protocol (PTP)

## IEEE 1588



*Cisco Nexus 3000*

1. Clock synchronization protocol for distributed and real-time applications
2. Provides sub microsecond/nano second level of accuracy
3. Used in high frequency trading, telecommunication, industrial automation and power system etc
4. Requires master and slave clocks
5. Uses multicast messaging
6. Assumes network to be symmetric

*We are modifying some features of PTP according to our architectural specification.*

[2] M. Levesque and D. Tipper, “ptp++: A Preciosn Time Protocol Simulation Model for OMNeT++/INET”, proceedings of the “OMNeT++ Community Summit 2015.

# PTP Synchronization Mechanism

1. Best master clock algorithm(BMC)
2. Master clock sends its time, *Sync* (TM1) to all slave clocks
3. Slave clock receives it at TS1 time
4. Master clock again send TM1 through *follow\_Up* message
5. Slave node sends a *Delay\_Req* (TS2) and master receives it at TM2 time
6. Master will send *Delay\_Resp*(TM2) message to slave
7. Clock offset,  $\theta = \frac{1}{2} * ((TS1 - TM1) - (TM2 - TS2))$
8. Update local time of slave ,  $T = T - \theta$

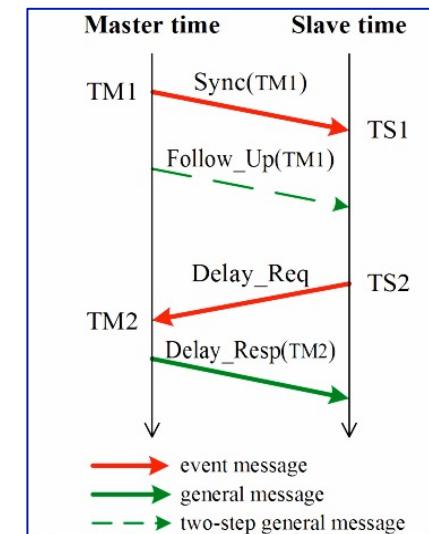


Fig 3: PTP synchronization mechanism [1]

[4] Y.Liu et al., “OMNeT++ Based Modeling and Simulation of the IEEE 1588 PTP Clock”, 2011 International Conference on Electrical and Control Engineering (ICECE), Sept 2011.

# Proposed Protocol

## Assumptions

- (1) Controller has the most accurate time and a time counters
- (2) TOR switch and controller has PTP enabled clocks
- (3) As TOR switches are physically connected to servers, they are synchronized
- (4) There is continuous traffic flow in the network
- (5) Clock bit length is less than padding length so it can replace padding bits requiring no additional bits

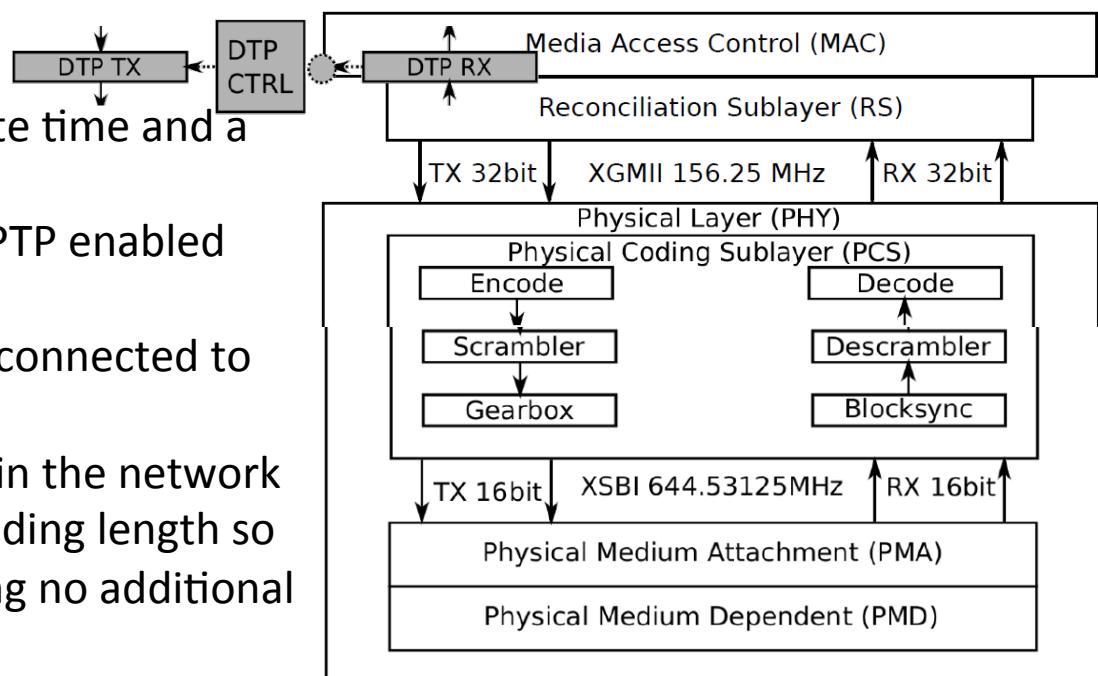


Fig 2: Clock synchronization Protocol in  
**UCDAVIS**

# Methodology

- Instead of using Best Master Clock(BMC) algorithm in PTP we assume the controller to be the master clock
- The TOR switches will be the slave clocks
- As PTP synchronization mechanism, controller-TOR switches will exchange messages as in Fig 3
- Maximum length of this control messages is 432 bits
- Padding/reserve bits of the frame will be used to transmit this messages
- Resynchronization time should be optimum in order to avoid over workload or under precision

# Flowchart

1. Synchronization starts from the controller
2. Resync\_Counter >0, synchronization process will not start
3. Resync\_Counter=0 , the controller sends Sync messages to all the framer
4. Framer will add this message in the padding bits of the deframer and send to TOR switches
5. TOR switches will calculate clock offset by exchanging 4 clock messages
6. When framer receives *Delay-Req* from TOR, controller reads it and sends *Delay\_Resp*
7. After sending *Delay\_Resp* message controller will reset its counter to resynchronization interval (eg: 100)

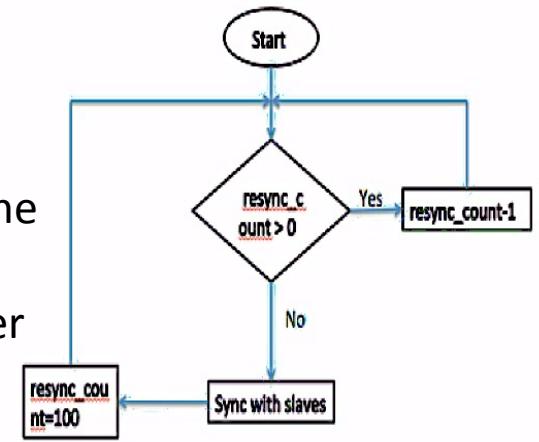


Fig 4: Flowchart of the proposed synchronization process

# Simulation Parameters

## Dataset :

1. Lognormal traffic distribution[4]
2. ON period/OFF period/ Inter arrival : lognormal[4]
3. No of TOR switches : 80
4. Control messages : 432 bits
5. Data packets : 200-1400 bytes[5]
6. Edge link usage : 57.52%[5]

[4] A. Roy et al., "Inside the Social Network's(Datacenter) Network," Proceedings of the 2015 on ACM SIGCOMM 2015 special Interest group on Data Communication, pp. 123-137, 2015.

[5] T. Benson et al., "Understanding data center traffic characteristics," ACM SIGCOMM Computer Commun., vol. 40, no. 1, pp. 92-99, 2010.

# Expected Result

1. In 5 minutes how many TOR switches are synchronized(*Delay\_Resp*)  
Maintaining separate counter for each TOR switch. Transmitting the control messages in padding/reserve bits only
2. Similar case as above. However, maintaining separate counter for each TOR switch. Transmitting control bits independent of data arrival
3. Comparing above two scenarios
4. Observing the synchronization pattern by changing input load
5. Observing synchronization pattern by changing resync frequency for changing load

# Reference

- [1] K.S. Lee, H. Wang , V. Shrivastav and H. Weatherspoon, “*Globally Synchronized Time via Datacenter Networks,*” Proceedings of the 2016 conference on ACM SIGCOMM 2016 Conference, pp. 454-467, 2016.
- [2] M. Levesque and D. Tipper, “*ptp++: A Preciosn Time Protocol Simulation Model for OMNeT++/INET*”, proceedings of the “OMNeT++ Community Summit 2015”.
- [3] Y.Liu and C. Yang, “*OMNeT++ Based Modeling and Simulation of the IEEE 1588 PTP Clock*”, 2011 International Conference on Electrical and Control Engineering (ICECE), Sept 2011.
- [4] A. Roy, H. Zeng, J. Bagga, G. Porter, and A. C. Snoeren,”*Inside the Social Network’s(Datacenter) Network,*” Proceedings of the 2015 on ACM SIGCOMM 2015 special Interest group on Data Communication, pp. 123-137, 2015.
- [5] T. Benson et al., “*Understanding data center traffic characteristics,*” ACM SIGCOMM Computer Commun., vol. 40, no. 1, pp. 92-99, 2010.