

Channel Bonding Sub-layer

Steve Shellhammer, Juan Montojo, Andrea Garavaglia,
Patrick Stupar, Nicola Varanese and Christian Pietsch
(Qualcomm)

Supporters

- Saif Rahman (Comcast)
- John Ulm (Motorola)
- Boris Brun (Harmonic)

Multi-channel Motion

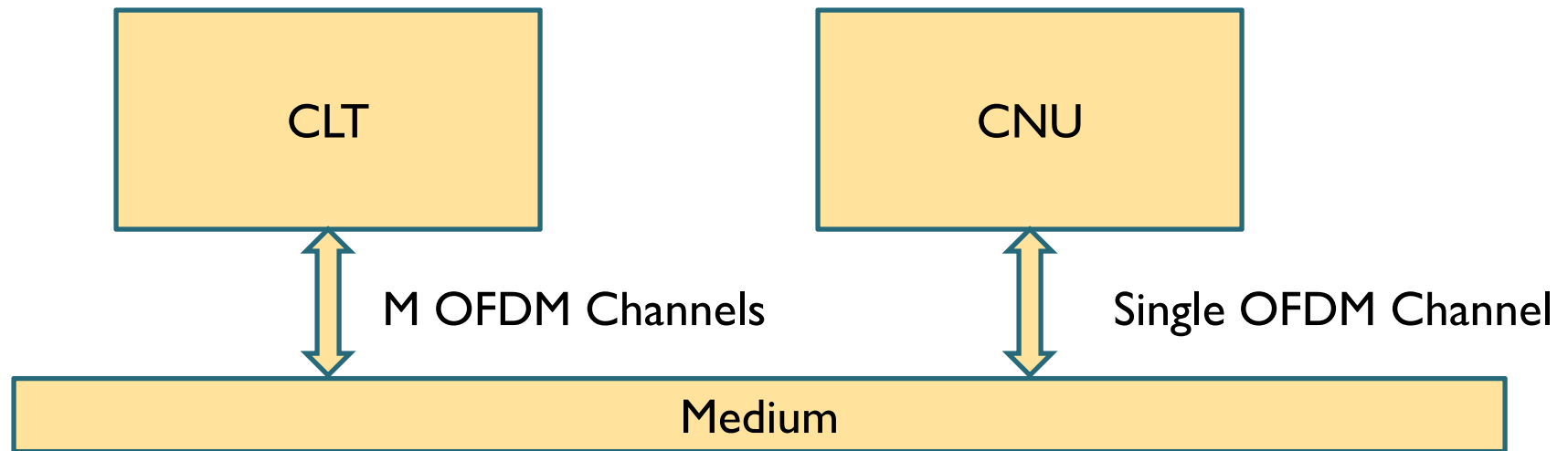
- At the September EPoC meeting the Task Force passed the following motion [1],
 - “The standard shall support the ability for higher capacity by combining multiple 192 MHz OFDM channels.”
- The Task Force needs to develop a method for combining multiple OFDM channels
 - This applies to the FDD downstream and possibly to TDD
 - This may not apply to the FDD upstream
- This presentation proposes a Channel Bonding Sub-layer (CBS) for combining OFDM channels

Goals

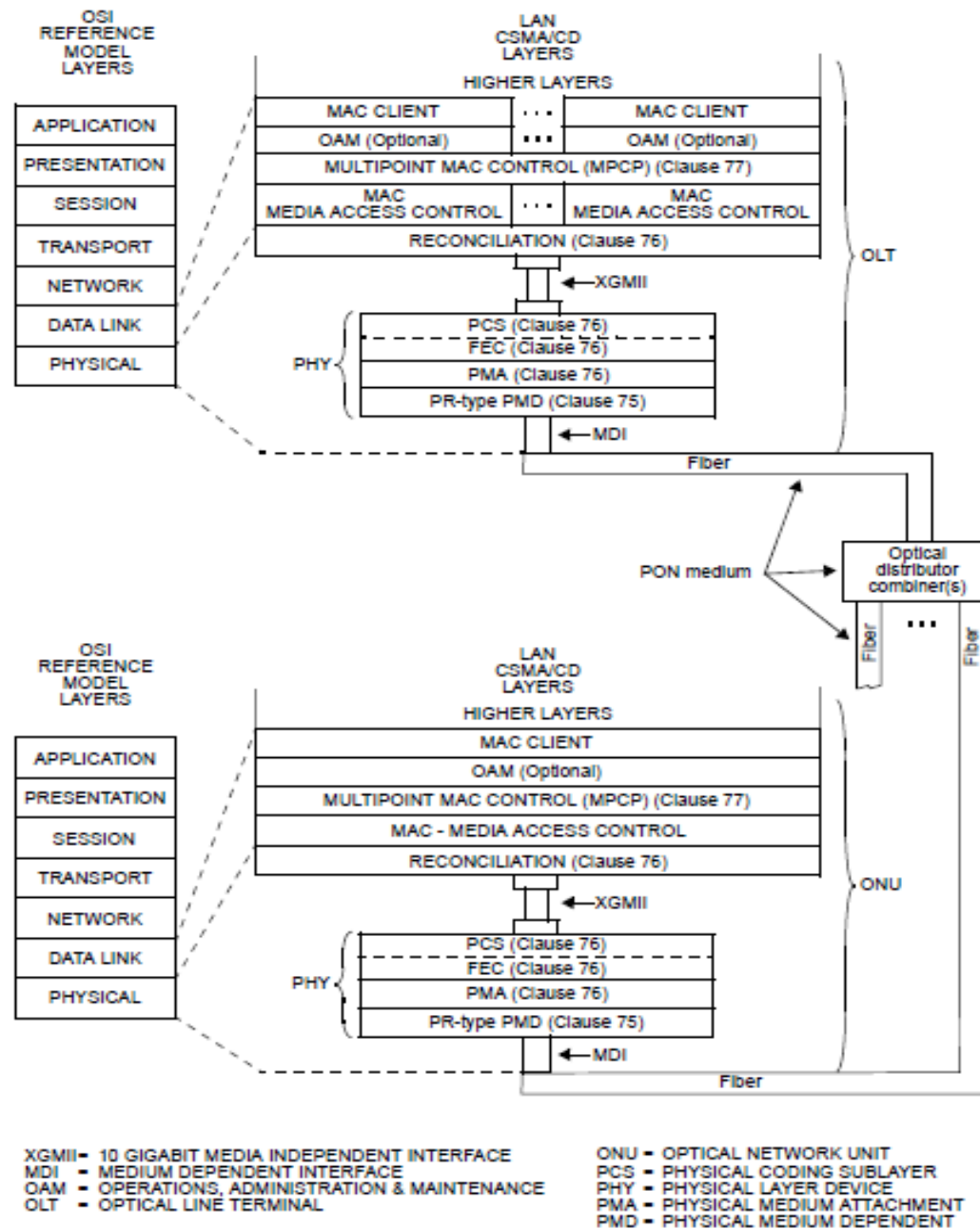
- Need to bond together M OFDM Channels
 - The maximum value of M needs to be specified by the Task Force
 - Four channels provides a total of 768 MHz of spectrum
 - Six channels provides a total of 1152 MHz of spectrum
- It is desirable that these channels be placed independently in the RF spectrum
 - Contiguous RF spectrum may not be available
 - Allows for both very wideband receiver and conventional super-heterodyne receiver
 - The channels should be non-overlapping
- We want to allow the CNU to bond fewer OFDM channels than the CLT, to enable low-cost, low-power consumer CNUs
 - Want to support a first generation single-channel CNU

Single-Channel CNU

- The standard should support a single-channel CNU to enable low-cost consumer devices [2]
 - If the CNU is required to support say 4 channels (e.g. 768 MHz) there will be a significant impact on the CNU cost
- This needs to work with a M channel CLT
 - Typical value of M could be M=4



10G EPON PHY Sub-layer (Reference)

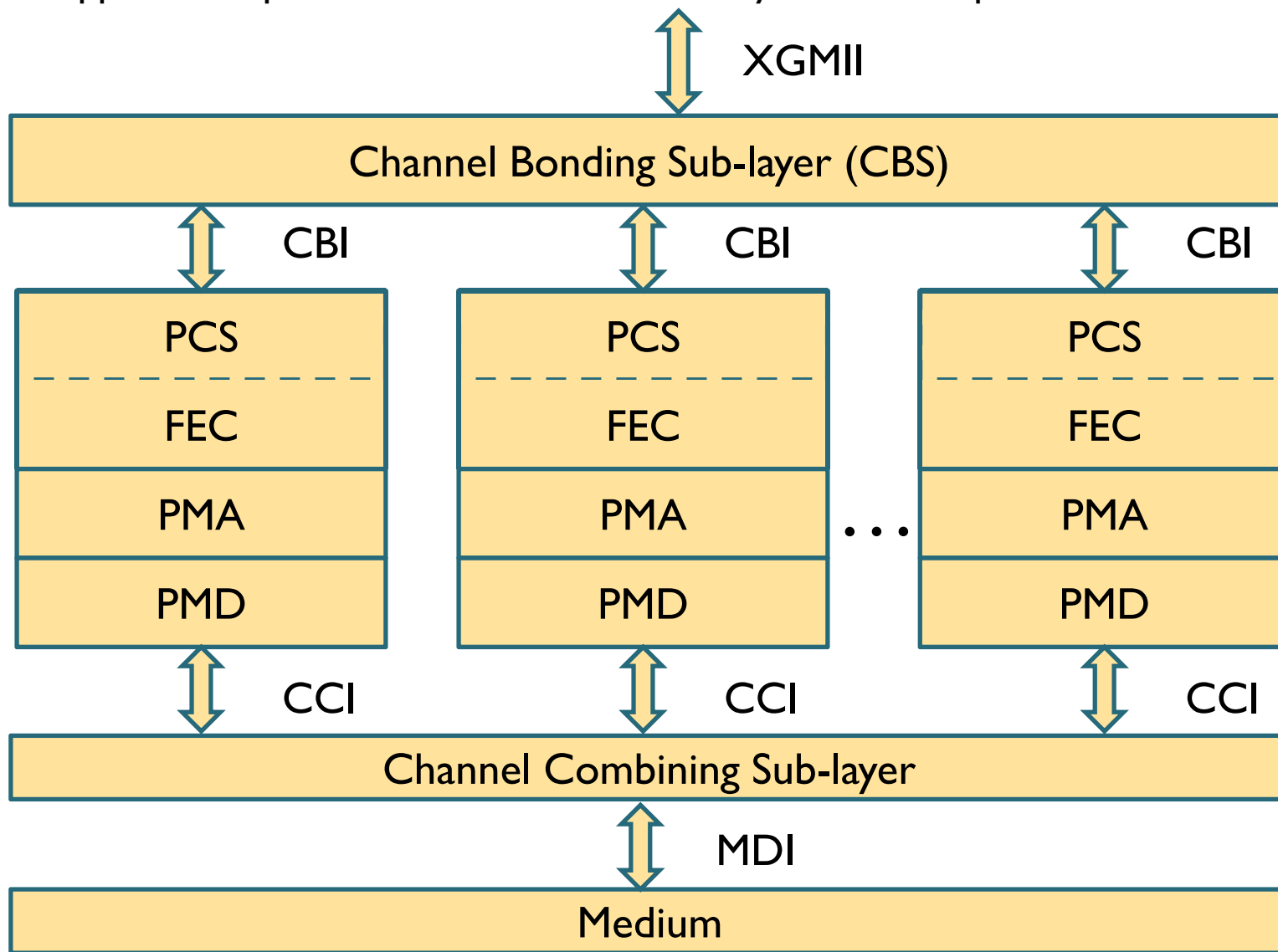


From IEEE 802.3av amendment, Figure 56-3

Figure 56-3—Architectural positioning of EFM:
P2MP 10/10G-EPON architecture (10 Gb/s downstream, 10 Gb/s upstream)

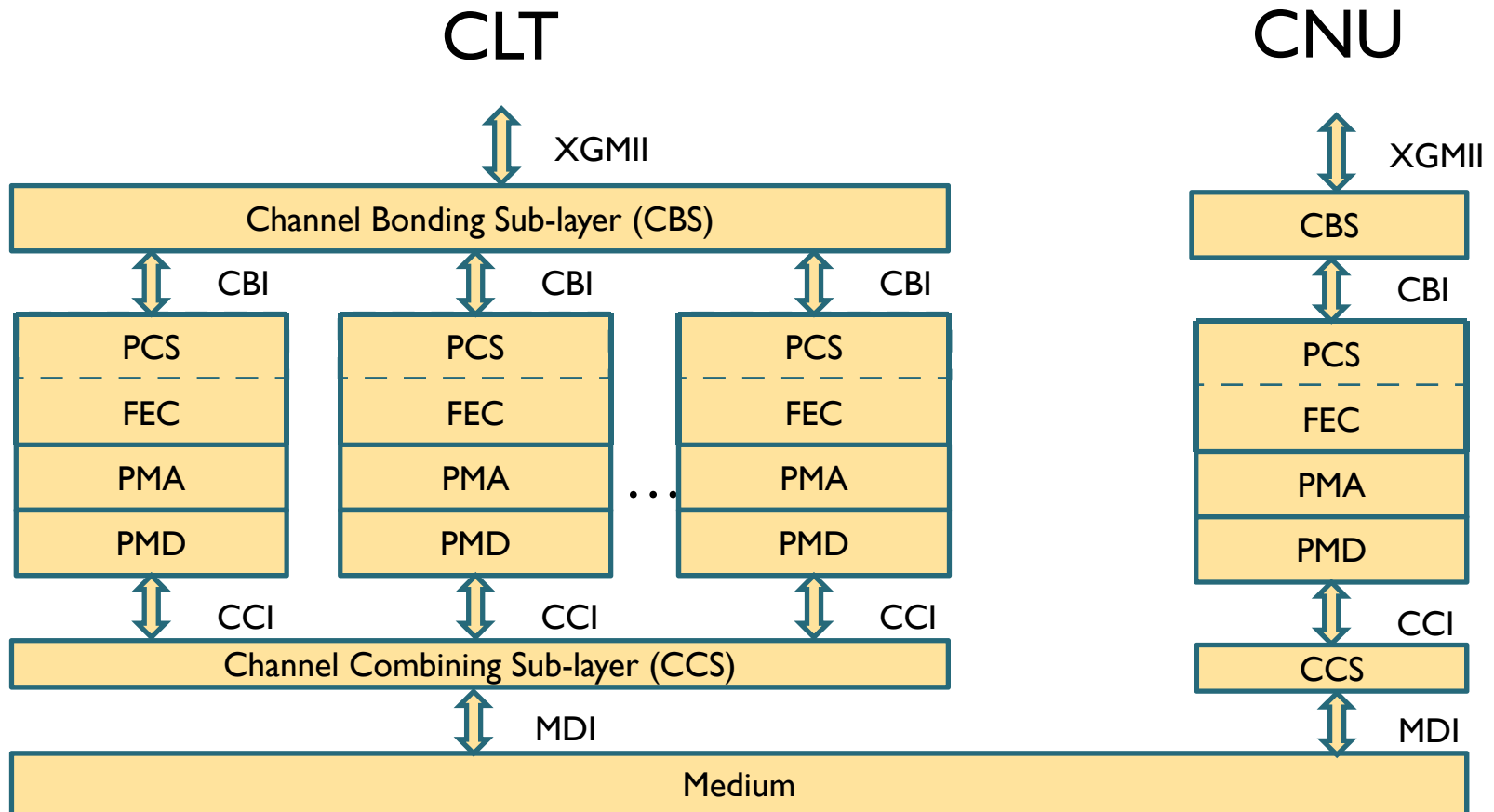
Add Channel Bonding Sub-layer

- Add Channel Bonding Sub-layer (CBS) above the PCS
- Add Channel Bonding Interface between CBS and PCS
- Support multiple PCS/FEC/PMA/PMD sub-layers for multiple OFDM channels



CLT/CNU Illustration

- Example
 - CLT supports three OFDM channels
 - CNU supports one OFDM channel



OFDM Symbol Synchronization

- The OFDM symbols in the OFDM channels are all time synchronized as to when the OFDM symbols begin and end
 - In FDD and TDD downstream this is straightforward since the CLT transmits all the OFDM channels
 - On the TDD upstream this is accomplished by time synchronizing the CNU clocks to the CLT. This is consistent with the motion to use OFDMA on the upstream [1], which requires time synchronization
 - Note: on the upstream the time synchronization is actually at the receive end (CLT) since there is compensation for propagation delay. For our purposes, we can assume synchronization

Channel Bonding Sub-layer

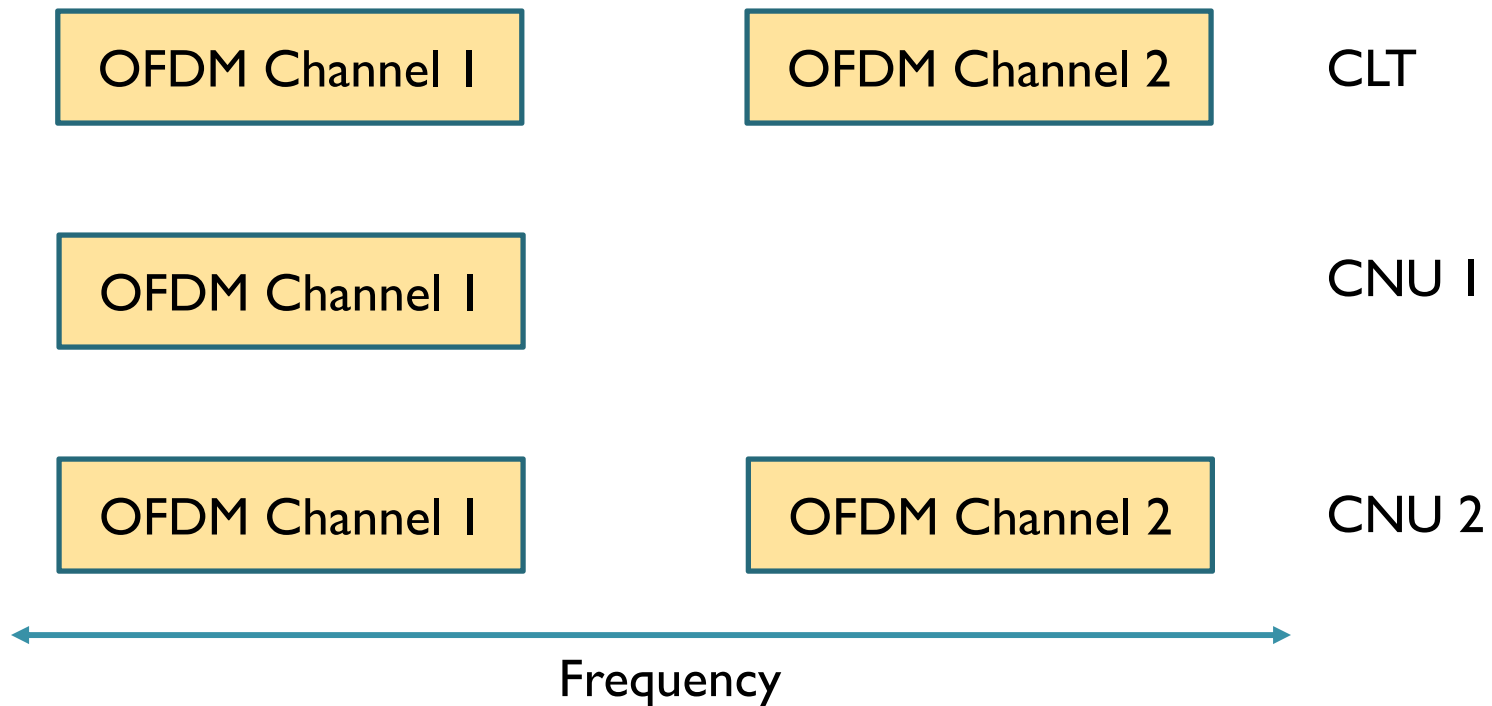
- For TX the CBS must distribute Ethernet Frames to the different channel bonding interfaces (CBIs)
- If the CNU has fewer OFDM channels than the CLT, then the CNU will be connected to a subset of the CLT's OFDM channels.
- In that case, the CBS must distribute the Ethernet Frames for that CNU to one of the CBIs that correspond to those OFDM Channels
- Also, the CBS must maintain Ethernet Frame order within an LLID

Channel Combining Sub-layer (CCS)

- The M channels must be combined on the transmit side
- The CCI connects to the medium through a single MDI (includes connector)
- On the receive side the CCI selects channels through filtering (analog or digital) and passes the channels to the PMD

Illustration

- CLT supports OFDM Channels 1 and 2
- CNU 1 supports OFDM Channels 1
- CNU 2 supports OFDM Channels 1 and 2



- Ethernet Frames to CNU 1 must be sent over CBI 1
- Ethernet Frames to CNU 2 can be sent over either CBI 1 or CBI 2, but cannot be fragmented and sent partially over CBI 1 and partially over CBI 2

CBS in CLT for Transmit

- The XGMII consists of a stream of Ethernet Frames and Idles
- The CBS must parse the XGMII stream and extract the Ethernet Frames, based on LLID
- The CBS must direct an Ethernet Frame to a specific CBI based on the LLID

Example Method for Mapping LLIDs to CBIs

- The CBS is configured with a table
 - For each LLID there is a list of which CBIs connected to that LLID
 - The number of CBIs in the table is equal to the number of OFDM channels connected to the CNU
- Configuration is done at a management layer
- The CBS layer selects one of the possible CBIs and maps the Ethernet frame to that CBI

LLID	CBI List
1 (CNU1)	1
2 (CNU2)	1,2

Example Table using Previous OFDM Channel Example

Note: In this simple example, there is only one LLID for each CNU

Buffering

- The CBS needs to buffer up a symbol worth of Ethernet Frames in order to select the CBI for each LLID
- In any channel bonding system there will need to be some buffering. It is either is in the PHY or in another layer.

Virtual Ordering

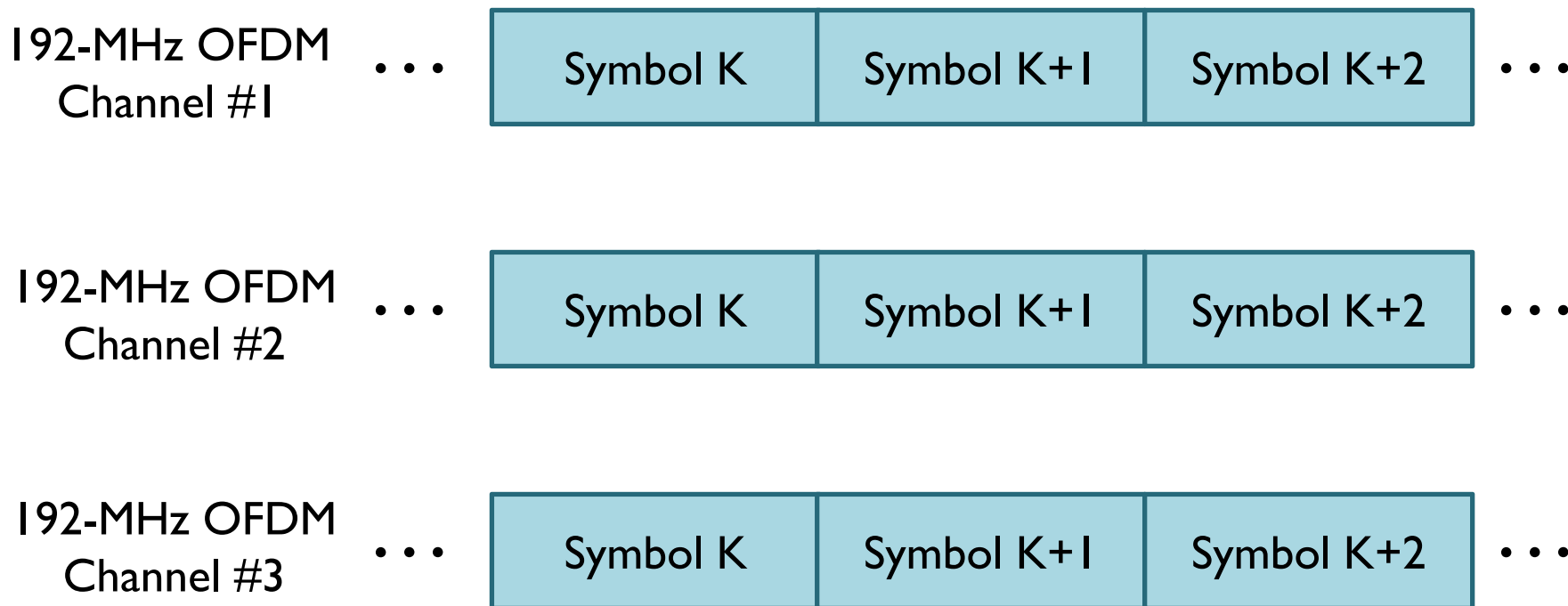
- Ethernet Frames with the same LLID may be sent to different CBIs
- It is necessary to ensure that the Ethernet Frames to the same LLID are received over the XGMII at the receiver in the same order that they are transmitted over the XGMII on at the transmitter
- It is desirable that Ethernet Frames to different LLIDs do not swap order, since they may be sent in that order to meet a QoS requirement; however, if they do swap order it does not cause a failure at a higher layer
- To meet this requirement we introduce Virtual Ordering

Virtual Ordering

- We will begin by addressing virtual ordering within an OFDM symbol and then add in channel bonding
- This is necessary even without channel bonding since the OFDM PHY transmits multiple Ethernet Frames at the same time within an OFDM symbol
- There must be some method of indicating the order of the Ethernet Frames within an OFDM symbol
- This concept was in some way introduced during the Study Group phase of the project [3]

Virtual Ordering

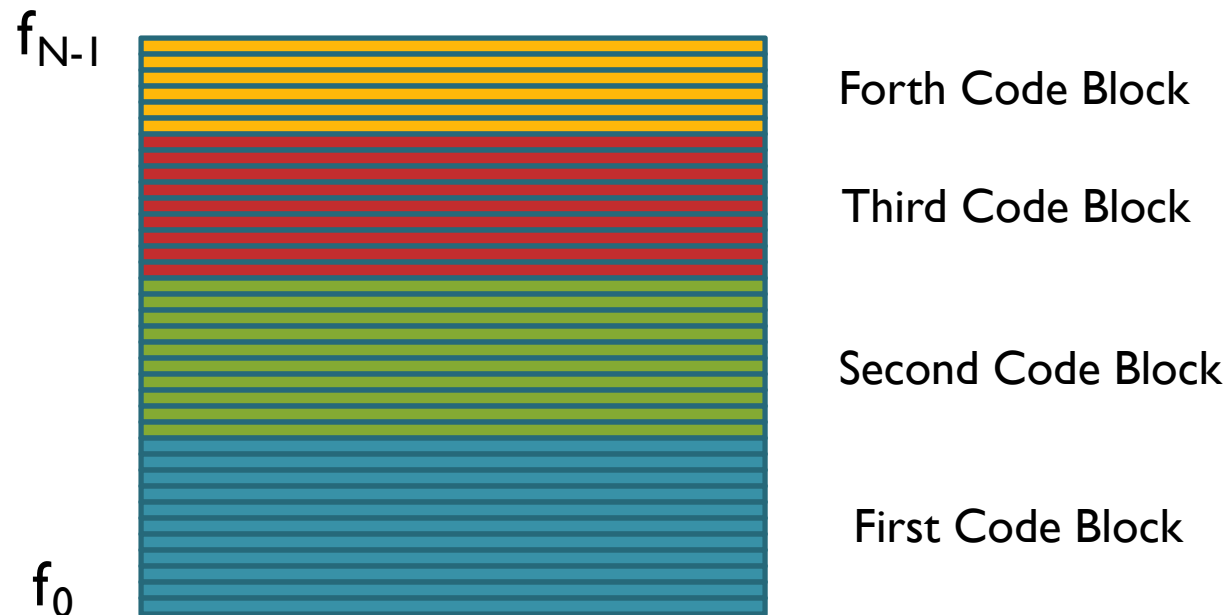
- Synchronous OFDM Symbols



- Ethernet Frames in Symbol K are before Ethernet Frames in Symbol K+1, even in different channels
- What if they are in the same symbol?

Virtual Ordering within an OFDM Symbol

- An OFDM symbol consists of N subcarriers
- Each Ethernet Frame (after FEC) is encapsulated within a subset of subcarriers.
- In [2] these FEC blocks containing the Ethernet Frames are placed in sequence of the OFDM subcarriers, beginning at the lowest frequency and progressing to the higher frequencies



Virtual Ordering within an OFDM Symbol

- We define the virtual ordering within an OFDM symbol using the following ordering rule
 - Given two code blocks, the one with the lowest subcarrier frequency is the first code block passed
- If code blocks are placed in order into the OFDM symbol according to this rule at the transmitter and read out of the OFDM symbol according to this rule then the order of the code blocks will be maintained
- If the order of the code blocks are maintained then the order of the Ethernet Frames will be maintained at the XGMII interface

Virtual Ordering with Channel Bonding

- Now we will address how to handle virtual ordering with channel bonding
- For two different channels, the code block with the lowest channel index is the first code block

Virtual Ordering Summary

Three Rules for Ordering Code Blocks

1. If two code blocks are in different symbol periods the one in the first symbol is the first code block
2. If two code blocks are in the same symbol period then the one with the lowest channel index is first
3. If two code blocks are in the same symbol period and have the same channel index, then the one with the lowest first subcarrier index is the first code block

Virtual Ordering Rules

Transmit

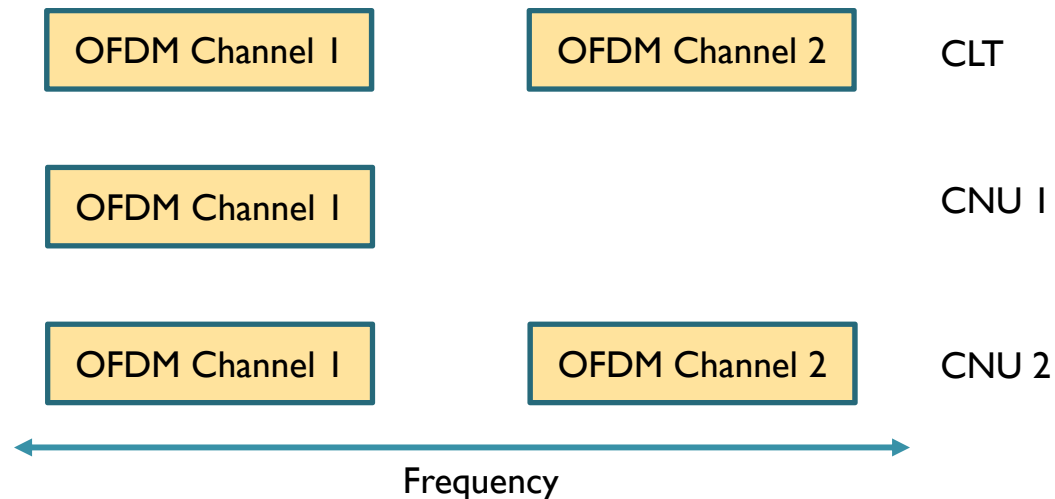
- At the CBS Ethernet Frames with the same LLID are sent to the lowest CBI first and then to increasing CBIs. Within a given CBI they are sent in the order they were received over the XGMII
- Code blocks must be placed in the OFDM symbol from lowest subcarrier index to highest

Receive

- Code blocks are sent up to the FEC from the lowest subcarrier index to the highest
- At the CBS Ethernet Frames are sent to the XGMII interface beginning with the lowest channel index and then within a channel index according to the order they were received over the CBI

Broadcast/Multicast Ethernet Frames

- If CNUs share channels, those shared channels can be used to transmit broadcast/multicast Ethernet Frames.
- This way, the broadcast/multicast Ethernet Frames do not need to be sent on all channels
- Example



- Broadcast/Multicast can be sent only on Channel 1

Broadcast/Multicast Ethernet Frames

- It is possible for a CNU to receive multiple copies of a multicast Ethernet Frame
- Example
 - CNU 1 is on OFDM Channel 1
 - CNU 2 is on OFDM Channel 2
 - CNU 3 is on OFDM Channel 1 and 2
 - Then CNU 3 will receive two copies of the multicast Ethernet Frame
- This can be solved by specifying for CNU 3 that its primary channel is Channel 1.
- It will only accept broadcast/multicast Ethernet Frames on its primary channel, and will discard broadcast/multicast Ethernet Frames on other channels.

Conclusions

- Described channel bonding goal to support low-cost single-channel CNU's
- Introduced a channel bonding sub-layer (CBS) and the channel bonding interface (CBI)
- Described a method for directing each Ethernet Frame over one CBI according to the LLID
- Introduced the virtual ordering rules to ensure that Ethernet frames to the same LLID maintain their order
- Provided virtual ordering rules to maintain Ethernet Frame order at the XGMII

Motions

- The standard shall support multiple generations of CNU from an entry level CNU that operates on a single 192-MHz OFDM channel and higher capacity CNU supporting multiple OFDM channels.
- The standard shall support a CLT that aggregates multiple 192-MHz OFDM channels. It should be possible for a single-channel CNU to receive on different OFDM channels.
- The standard shall support a Channel Bonding Sub-layer above the PCS for aggregating multiple OFDM channels together.

References

1. Duane Remein, “Unconfirmed Minutes of IEEE P802.3bn EPoC PHY Task Force Meeting,” September 27-28, 2012, ITU, Geneva, Switzerland
2. Steve Shellhammer, Juan Montojo, Masoud Zargari and Andrea Garavaglia, “EPoC FDD Downstream RF Bandwidth Proposal,” September 2012
3. Ed Boyd, Eugene Dai, Matt Schmitt and Mark Laubach, “EPOC Upstream TDMA Slot Mapping,” IEEE 802.3 EPoC Study Group, March 2012