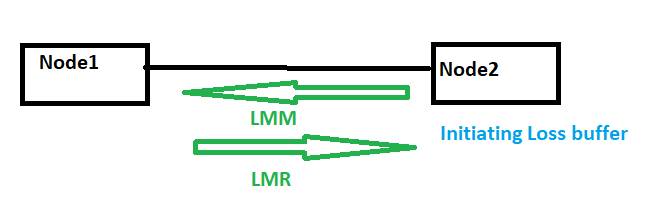
LOSS MEASUREMENT/ DELAY MEASUREMENT



Performance Monitoring(Loss and Delay)

It enables you to identify network problems before customers are impacted by network defects.

Loss Measurement is performed by sending frames with ETH-LM information to peer MEP and receiving frames with ETH-LM information from peer MEP.

* Dual-Ended ETH-LM, using CCM frame with ETH-LM information
* Single-Ended ETH-LM, using LMM and LMR frame.
* The software registers with the hardware to receive LMRs. When the software receives an LMR, the software compares the values of the transmitted and received counters. For each received LMR, the software records the following information – {time-stamp, time interval for this measurement, number of frames lost, port, peer MEP MAC address}. This information is stored in a rolling buffer for an operator to view whenever required. The size of the buffer is configurable by the operator.

Delay measurements are performed by sending periodic frames with ETH-DM information to the peer MEP and receiving frames with ETH-DM information from the peer MEP.

Frame Delay Measurement can be performed in the following ways.

* One-Way DM, using 1DM frame.
* Two-Way DM, using DMM and DMR frame.

The software registers with the hardware to receive DMRs. When the software receives a DMR, the software compares the values of the transmitted and received time-stamps and calculates the frame delay. For each received DMR, the software records the following information – {time-stamp of this measurement, time delay calculated, port, peer MEP MAC address}. This information is stored in a rolling buffer for an operator to view whenever required. The size of the buffer is configurable by the operator.

**Initiating loss buffer:**

* EcfmCcClntLmInitiator->EcfmLmInitXmitLmmPdu->EcfmCcCtrlTxTransmitPkt.

**Reception of LMM in peer node:**

* EcfmCcPktQueueHandler->EcfmCcClntProcessLmm->>EcfmLmResXmitLmrPdu()

**Reception of LMR/DMR:**

* EcfmCcClntProcessLmr->EcfmCcClntParseLmr->EcfmCcClntCalcFrameLoss()
* EcfmLbltClntProcessDmr🡪EcfmDmInrCalcDiffInTimeRep🡪EcfmLbLtCtrlRxOpCodeDeMux🡪

N1---P:9--N2

Loss calculation Flow of ECFM:

EcfmGetPacketCounters🡪VSiGetPortStats -🡪Calculates stats for physical port.

Loss calculation flow of MPLS-OAM:

FsMplsHwGetVcLblBasedStats🡪 NPAPI calculates loss for MPLS LIF.

Tx/RX flow :EcfmCcPktQueueHandler->EcfmProcessMpTpPdu->EcfmMpTpCcProcessPdu

CALCULATION OF LOSS

TxFCf: Value of the local counter TxFCl at the time of transmission of the CCM frame.   
 RxFCb: Value of the local counter RxFCl at the time of reception of the last CCM frame from the peer MEP.   
 TxFCb: Value of TxFCf in the last received CCM frame from the peer MEP.

LMR:

00 02 02 03 04 0A 00 03 02 03 04 0A 81 00 E0 64 89 02 80 2A 00 0C 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 02 02 03 04 0A 00 03 02 03 04 0A 81 00 E0 64 89 02 80 2B 00 0C 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

LMM:

00 02 02 03 04 0A 00 03 02 03 04 0A 81 00 E0 64 89 02 80 2B 00 0C 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 02 02 03 04 0A 00 03 02 03 04 0A 81 00 E0 64 89 02 80 2B 00 0C 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

Ethernrt cfm offload-----

Filter Dump- CFM Offload:

Entry 157

Flags = 00000007

+USED +IN\_HW +WANT\_HW -upd -chg -new -sta

Group = 10

Priority = 0

Prev/Next = 156 / 158

HW entry ID = 00000066

HW priority = 00000000

Qualifiers:

EtherType (29) -> ethertype (14)

8902/FFFF

0000000000008902/000000000000FFFF (expected)

0000000000008902/000000000000FFFF (actual)

002A00/00FF00 ------->ECFM\_OPCODE\_LMR

0000000000002A00/000000000000FF00 (expected)

0000000000002A00/000000000000FF00 (actual)

BCM actions:

Trap (251)

1600702A,00000000

PPD actions:

trap (3)

0000072A

Entry 158

Flags = 00000007

+USED +IN\_HW +WANT\_HW -upd -chg -new -sta

Group = 10

Priority = 0

Prev/Next = 157 / 65536

HW entry ID = 00000067

HW priority = 00000000

Qualifiers:

EtherType (29) -> ethertype (14)

8902/FFFF

0000000000008902/000000000000FFFF (expected)

0000000000008902/000000000000FFFF (actual)

002B00/00FF00 ----------->ECFM\_OPCODE\_LMM

0000000000002B00/000000000000FF00 (expected)

0000000000002B00/000000000000FF00 (actual)

ECFM Opcode:

ECFM\_OPCODE\_LMR,  42   0x2a  
ECFM\_OPCODE\_LMM, 43   0x2b

ECFM\_OPCODE\_DMR, 46  2e  
ECFM\_OPCODE\_DMM, 47   2f

NpFsEcfmOffloadStatus🡪FsNpUpdateRxFilter(installs for LM and DM opcodes)