

# L2TP Network Server

---

## In This Chapter

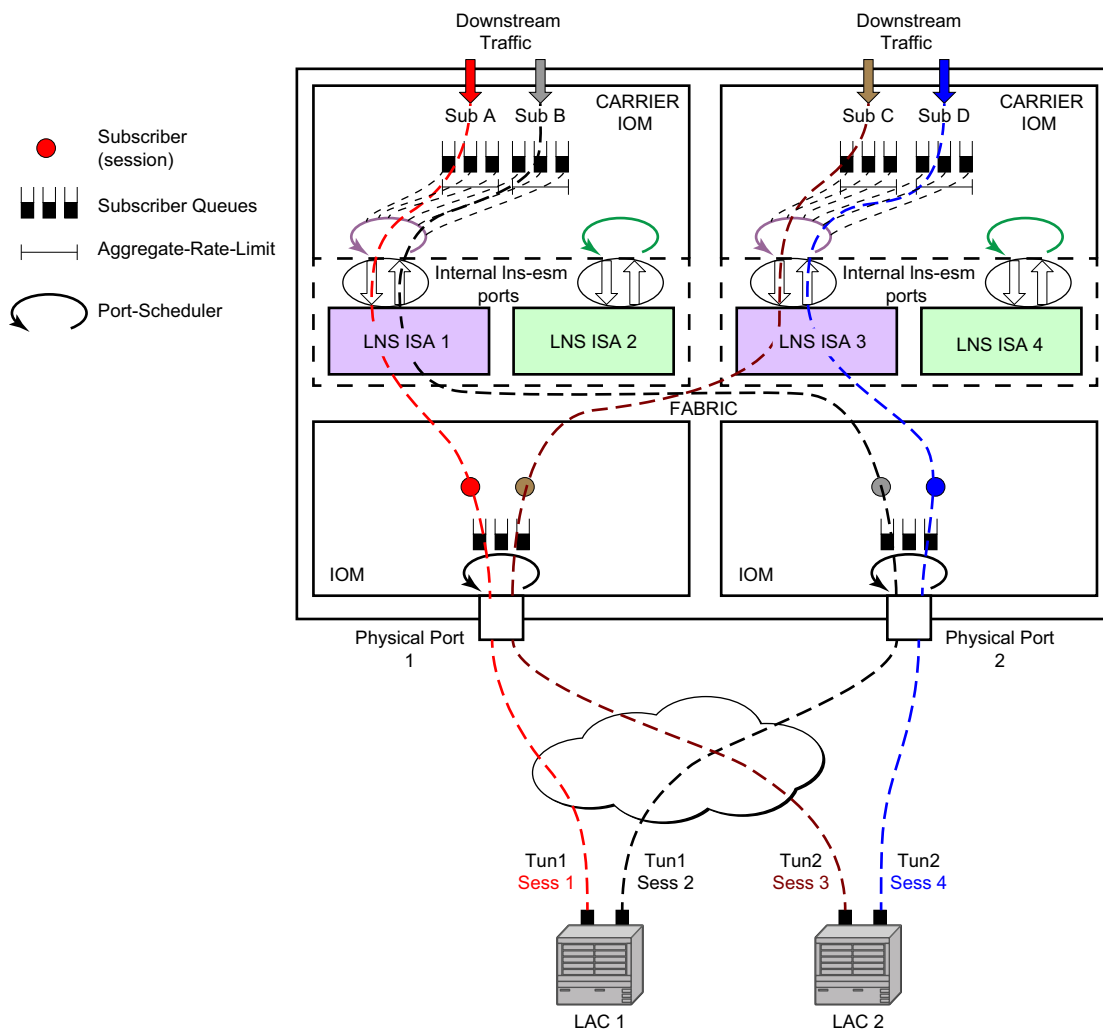
This chapter provides information about L2TP Network Server (LNS) aspects, including configuration process overview, and implementation notes.

Topics in this chapter include:

- [Subscriber agg-rate-limit on LNS on page 728](#)
- [MLPPPoE, MLPPP\(oE\)oA with LFI on LNS on page 735](#)

## Subscriber agg-rate-limit on LNS

In non-LNS ESM environment, the existing **agg-rate-limit** command is applied to the subscriber within the subscriber profile (sub-profile). However, the agg-rate-limit cannot be the highest level in subscriber's HQoS hierarchy. The agg-rate-limit will be only effective if it is applied to a subscriber that is tied to a port-scheduler. In other words, the port-scheduler in subscriber's HQoS hierarchy is a prerequisite for successful operation of agg-rate-limit. On regular MDAs, the port-scheduler is directly applied to a physical port. The port between the carrier IOM and the ISA is an internal port that is not exposed in the CLI. This is shown in Figure 48.



OSSG688

Figure 48: QoS Hierarchy on LNS

The port-scheduler will be applied to the internal lns-esm port in the egress direction. The lns-esm egress port is a port between the carrier IOM and the ISA that is passing traffic from all VRFs that have subscriber L2TP sessions terminated in the corresponding ISA.

The port-scheduler will be applied to each lns-esm port with the following CLI:

```
CLI Syntax: configure
                port-policy <port-policy-name>
                  egress-scheduler-policy <port-scheduler-policy-name>
```

Port-policy at the root CLI level will create a port policy manager that can apply various policies (port scheduler) to hidden, dynamically created ports for WLAN GW/LNS/NAT.

```
CLI Syntax: configure
                isa
                  lns-group <grp-id>
                    mda <card>/<slot>
                    mda <card>/<slot>
                    :
                    port-policy <port-policy-name>
```

The port policy itself will be applied to internal LNS port under the lns-group CLI hierarchy. The port scheduler will automatically be applied to egress lns-esm ports on carrier IOMs towards every LNS ISA in the lns-group. The port schedulers will have the same configuration on every lns-esm port in the lns group but will operate independently on each port.

Additional consideration:

- An ISA can be assigned to a single lns-group. In other words, two or more LNS-groups can not contain the same ISA. However, an ISA can belong simultaneously to an LNS-group and a NAT group. The port scheduler will affect only LNS traffic.
- The port scheduler rates are wire rates that are based on the encapsulation between the carrier IOM and the ISA which is Ethernet QinQ. However, the queue rates, the billing stats and the agg-rate-limit rates can be optionally based on the last mile encapsulation in the same way as they have been supported in non-LNS environment with **queue-frame-based-accounting** and **encap-offset** commands.

The ability to calculate queue rates or the agg-rate-limit based on the last mile encapsulation is referred to as Last Mile Aware Shaping.

For example, the **encap-offset** command will cause the queue rates, the billing stats and the agg-rate-limit to be based on the wire encapsulation in the last mile. For ATM in the last mile, the wire overhead will be calculated per each packet (including ATM cellification overhead and padding). For Ethernet in the first mile, a fixed last mile

encapsulation (defined with the **encap-offset** command or the RFC 5515, *Layer 2 Tunneling Protocol (L2TP) Access Line Information Attribute Value Pair (AVP Extensions)*) wire overhead will be considered in rate calculation. In essence the length of the PPPoE Ethernet QinQ header that is used on the link between the carrier IOM and the ISA will be artificially modified so that it matches the length of the header used in the last mile. The net effect is rate shaping on LNS based on the virtual packet length that is present in the last mile.

The last mile encapsulation information that is used in Last Mile Aware Shaping can be obtained either statically through the explicit value in the **encap-offset** command or dynamically by the RFC 5515 method (AVP 144 in ICRQ). The latter will be the case if the **encap-offset** command does not have any explicitly configured value.

In the absence of the **encap-offset** command, the queue rates, the billing stats and the agg-rate-limit rates will be based on the Ethernet QinQ encapsulation between the carrier IOM and the ISA. Depending on the queue-frame-based-accounting configuration option, those rates can be wire based or data based (Layer 2 encapsulation only).

- The agg-rate-limit is not applicable to ingress direction (LNS or non LNS ESM).
- V-Port is not applicable in LNS configuration.



# LNS Reassembly

---

## Overview

In certain cases PPPoE clients do not honor the negotiated MRU during the LCP phase and consequently they will send packets larger than the negotiated MRU. This applies to control and data packets.

In this case, the LAC will fragment IPv4 packets which will then have to be reassembled in LNS.

In general, reassembly processing applies only to the end nodes that are receiving fragments. In tunneled environment a fragmented packet must be reassembled before it is de-capsulated.

---

## Reassembly Function

LNS reassembly is implemented through a generic IPv4 reassembly function that can be shared across multiple ISAs in a nat-group. The same ISA can be independently part of an lns-group and a nat-group.

Traffic that needs to be reassembled is steered to the nat-group via filters. Once the fragmented traffic is in the nat-group, it will be reassembled and injected back within the same routing context to the lns-group for further L2TP processing.

Configuration steps:

- Configure two isa groups, a nat-group providing generic reassembly function and a lns-group providing the L2TP services. The ISAs can be shared amongst the groups, or they can be separated per each group :

```

CLI Syntax: configure
                isa
                  nat-group 1
                    active-mda-limit 2
                    mda 1/1
                    mda 1/1
                  lns-group 1
                    mda 1/1
                    mda 1/2
  
```

- Configure redirection of the L2TP traffic to the nat-group performing reassembly:

```

CLI Syntax: configure
                filter
  
```

```

ip-filter 10
  entry 5
    match
      dst-ip 10.10.10.10 - traffic classification cri-
                          teria ; in this case LNS tunnel endpoint.
    action reassemble
  default-action forward

```

- Apply 'reassembly' filter on the incoming L2TP traffic:

**CLI Syntax:**

```

configure
router
  interface from-lac
    address 10.0.0.1/24
    port 2/2/2
    ingress
    filter ip 10

```

- Associate the reassembly context with the same service where LNS is configured.

**CLI Syntax:**

```

configure
service
  vprn 10
    reassembly-group 1
      l2tp
        group "lns-vrf-10" create
          ppp
            authentication-policy "lns"
            proxy-authentication
            proxy-lcp
          tunnel "lns-test-tunnel" create
            lns-group 1
            no shutdown

        subscriber-interface "int1" create
          address 10.20.20.254/24
          group-interface "lns-grp-10" lns create
            sap-parameters
            sub-sla-mgmt
            sub-ident-policy "sub-ident"
        dhcp
          server 192.168.1.1
            trusted
            client-applications ppp
            gi-address 10.20.20.1

```

## Load Sharing Between the ISAs

All traffic matching the criteria associated with the filter action reassemble will be forwarded to the reassembly function, regardless of whether the traffic is fragmented or not.

In case that there are multiple ISAs in the NAT-group, traffic is load shared between them based on the source IP address and the incoming service id (routing context).

---

## Inter-chassis ISA Redundancy

In case that an active ISA fails in a nat-group, the standby ISA will take over the reassembly function. However, the switchover is not statefull and consequently traffic destined to the failed ISA will be lost until it is restarted.



## MLPPPoE, MLPPP(oE)oA with LFI on LNS

MLPPPoX is generally used to address bandwidth constraints in the last mile. The following are other uses for MLPPPoX:

- To increase bandwidth in the access network by bundling multiple links/VCs together. For example it is less expensive for a customer with an E1 access to add another E1 link in order to increase the access b/w, rather than to upgrade to the next circuit speed (E3).
- LFI on a single link to prioritize small packet size traffic over traffic with large size packets. This is needed in the upstream and downstream direction.

PPPoE and PPPoEoA/PPPoA v4/v6 host types are supported.

---

## Terminology

The term MLPPPoX is used to reference MLPPP sessions over ATM transport (oA), Ethernet over ATM transport (oEoA) or Ethernet transport (oE). Although MLPPP in subscriber management context is not supported natively over PPP/HDLC links, the terms MLPPP and MLPPPoX terms can be used interchangeably. The reason for this is that link bundling, MLPPP encapsulation, fragmentation and interleaving can be in a broader scope observed independently of the transport in the first mile. However, MLPPPoX terminology will be prevailing in this document in an effort to distinguish MLPPP functionality on ASAP MDA (outside of ESM) and MLPPPoX in LNS (inside of ESM),.

Terms speed and rate are interchangeably used throughout this section. Usually speed refers to the speed of the link in general context (high or low) while rate usually quantitatively describes the link speed and associates it with the specific value in bps.

## LNS MLPPPoX

This functionality is supported through LNS on BB-ISA. LNS MLPPPoX can be used then as a workaround for PTA deployments, whereby LAC and LNS can be run back-to-back in the same system (connected via an external loop or a VSM2 module), and thus locally terminate PPP sessions.

MLPPPoX can:

- Increase bandwidth in the last mile by bundling multiple links together.
- LFI/reassembly over a single MLPPPoX capable link (plain PPP does not support LFI).

## MLPPP Encapsulation

Once the MLPPP bundle is created in the 7750 SR, traffic can be transmitted by using MLPPP encapsulation. However, MLPPP encapsulation is not mandatory over an MLPPP bundle.

MLPPP header is primarily required for sequencing the fragments. But in case that a packet is not fragmented, it can be transmitted over the MLPPP bundle using either plain PPP encapsulation or MLPPP encapsulation.

## MLPPPoX Negotiation

MLPPPoX is negotiated during the LCP session negotiation phase by the presence of the Max-Received-Reconstructed Unit (MRRU) field in the LCP ConfReq. MRRU option is a mandatory field required in MLPPPoX negotiation. It represents the maximum number of octets in the Information field of a reassembled packet. The MRRU value negotiated in the LCP phase must be the same on all member links and it can be greater or lesser than the PPP negotiated MRU value of each member link. This means that the reassembled payload of the PPP packet can be greater than the transmission size limit imposed by individual member links within the MLPPPoX bundle. Packets will always be fragmented so that the fragments are within the MRU size of each member link.

Another field that could be optionally present in an MLPPPoX LCP Conf Req is an Endpoint Discriminator (ED). Along with the authentication information, this field can be used to associate the link with the bundle.

The last MLPPPoX negotiated option is the Short Sequence Number Header Format Option which allows the sequence numbers in MLPPPoX encapsulated frames/fragments to be 12-bit long (instead 24-bit long, by default).

Once the multilink capability is successfully negotiated via LCP, PPP sessions can be bundled together over MLPPPoX capable links.

The basic operational principles are:

- LCP session is negotiated on each physical link with MLPPPoX capabilities between the two nodes.
- Based on the ED and/or the authentication outcome, a bundle is created. A subsequent IPCP negotiation is conveyed over this bundle. User traffic is sent over the bundle.
- If a new link tries to join the bundle by sending a new MLPPPoX LCP Conf Request, the LCP session will be negotiated, authentication performed and the link will be placed under the bundle containing the links with the same ED and/or authentication outcome.
- IPCP/IPv6CP will be in the whole process negotiated only once over the bundle. This negotiation will occur at the beginning, when the first link is established and MLPPPoX bundle created. IPCP and IPv6CP messages are transmitted from the 7750 LNS without MLPPPoX encapsulation, while they can be received as MLPPPoX encapsulated or non-MLPPPoX encapsulated.

## Enabling MLPPPoX

The lowest granularity at which MLPPPoX can be enabled is an L2TP tunnel. An MLPPPoX enabled tunnel is not limited to carrying only MLPPPoX sessions but can carry normal PPP(oE) sessions as well.

In addition to enabling MLPPPoX on the session terminating LNS node, MLPPPoX can also be enabled on the LAC via PPP policy. The purpose of enabling MLPPPoX on the LAC is to negotiate MLPPPoX LCP parameters with the client. Once the LAC receives the MRRU option from the client in the initial LCP ConfReq, it will change its tunnel selection algorithm so that all sessions of an MLPPPoX bundle are mapped into the same tunnel.

The LAC will negotiate MLPPPoX LCP parameters regardless of the transport technology connected to it (ATM or Ethernet). LCP negotiated parameters are passed by the LAC to the LNS via Proxy LCP in ICCN message. In this fashion the LNS has an option to accept the LCP parameters negotiated by the LAC or to reject them and restart the negotiation directly with the client.

The LAC will transparently pass session traffic handed to it by the LNS in the downstream direction and the MLPPPoX client in the upstream direction. The LNS and the MLPPPoX client will perform all data processing functions related to MLPPPoX such as fragmentation and interleaving.

Once the LCP negotiation is completed and the LCP transition into an open state (configuration ACKs are sent and received), the Authentication phase on the LAC will begin. During the Authentication phase the L2TP parameters will become known (l2tp group, tunnel, etc), and the session will be extended by the LAC to the LNS via L2TP. In case that the Authentication phase does not return L2TP parameters, the session will be terminated because the 7750 does not support directly terminated MLPPPoX sessions.

In the case that MLPPPoX is not enabled on the LAC, the LAC will negotiate plain PPP session with the client. In case that the client accepts plain PPP instead of MLPPPoX as offered by the LAC, when the session is extended to the LNS, the LNS will re-negotiate MLPPPoX LCP with the client on a MLPPPoX enabled tunnel. The LNS will learn about the MLPPPoX capability of the client via Proxy LCP message in ICCN (first Conf Req received from the client is also send in Proxy LCP). If the there is no indication of the MLPPPoX capability of the client, the LNS will establish a plain PPP(oE) session with the client.

Note that there is no dependency between ATM autosensing on LAC and MLPPPoX since autosensing operates on a lower layer than PPP (LCP).

## Link Fragmentation and Interleaving (LFI)

The purpose of LFI is to ensure that short high priority packets are not delayed by the transmission delay of large low priority packets on slow links.

For example it takes ~150ms to transmit a 5000B packet over a 256Kbps link, while the same packet is transmitted in only 40us over a 1G link (~4000 times faster transmission). To avoid the delay of a high priority packet by waiting in the queue while the large packet is being transmitted, the large packet can be segmented into smaller chunks. The high priority packet can be then interleaved with the smaller fragments. This approach can significantly reduce the delay of high priority packets.

The interleaving functionality is only supported on MLPPPoX bundles with a single link. If more than one link is added into a interleaving capable MLPPPoX bundle, then interleaving will be internally disabled and the `tmnxMlpppBundleIndicatorsChange` trap will be generated.

With interleaving enabled on an MLPPPoX enabled tunnel, the following session types are supported:

- Multiple LCP sessions tied into a single MLPPPoX bundle. This scenario assumes multiple physical links on the client side. Theoretically it would be possible to have multiple sessions running over the same physical link in the last mile. For example, two PPPoE sessions going over the same Ethernet link in the last mile, or two ATM VCs on the same last mile link. Whichever the case might be, the LAC/LNS is unaware of the physical topology in the last mile (single or multiple physical links). Interleaving functionality will be internally disabled on such MLPPPoX bundle.
- A single LCP session (including dual stack) over the MLPPPoX bundle. This scenario assumes a single physical link on the client side. Interleaving will be supported on such single session MLPPPoX bundle as long as the conditions for interleaving are met. Those conditions are governed by max-fragment-delay parameter and calculation of the fragment size as described in subsequent sections.
- An LCP session (including dual stack) over a plain PPP/PPPoE session. This type of session is a regular PPP(oE) session outside of any MLPPPoX bundle and therefore its traffic is not MLPPPoX encapsulated.

Packets on an MLPPPoX bundle are MLPPPoX encapsulated unless they are classified as high priority packets when interleaving is enabled.

## MLPPPoX Fragmentation, MRRU and MRU Considerations

A packet of the size greater than the internally calculated fragment length cannot be natively transmitted over an MLPPPoX bundle. Such packet will be MLPPPoX encapsulated and consequently fragmented. This is irrespective of whether the fragmentation is enabled or disabled. The size of the internally calculated fragment length depends on:

- The desired transmission delay in the last mile.
- The fragment “payload to encapsulation overhead” efficiency ratio.
- Various MTU sizes in the 7750 dictated mainly by received MRU, received MRRU and configured PPP MTU under the following hierarchy:
  - configure service vprn l2tp group ppp mtu
  - configure service vprn l2tp group tunnel ppp mtu
  - configure router l2tp group ppp mtu
  - configure router l2tp group tunnel ppp mtu

In cases where MLPPPoX fragmentation is disabled with the `no max-fragment-delay` command, it is expected that packets are not MLPPPoX fragmented but rather only MLPPPoX encapsulated in order to be load balanced over multiple physical links in the last mile. However, even if MLPPPoX fragmentation is disabled, it is possible that fragmentation occurs under certain circumstances. This behavior is related to the calculation of the MTU values on an MLPPPoX bundle.

MLPPPoX in the 7750 is concerned with two MTUs:

- `bundle-mtu` determines the maximum length of the original IP packet that can be transmitted over the entire bundle (collection of links) before any MLPPPoX processing takes place on the transmitting side. This is also the maximum size of the IP packet that the receiving node can accept once it de-encapsulates and assembles received MLPPPoX fragments of the same packet. `Bundle-mtu` is relevant in the context of the collection of links.
- `link-mtu` determines the maximum length of the payload before it is PPP encapsulated and transmitted over an individual link within the bundle. `Link-mtu` is relevant in the context of the single link within the bundle.

Assuming that the CPE advertized MRRU and MRU values are smaller than any configurable `mtu` on MLPPPoX processing modules in the 7750 (carrier IOM and BB-ISA), the `bundle-mtu` and the `link-mtu` will be based on the received MRRU and MRU values, respectively. For example, the `bundle-mtu` will be set to the received MRRU value while `link-bundle` will be set to the MRU value minus the MLPPPoX encapsulation overhead (4 or 6 bytes).

Consider an example where received MRRU value sent by CPE is 1500B while received MRU is 1492B. In this case, our `bundle-mtu` will be set to 1500B and our `link-mtu` will be set to 1488B (or

1486B) to allow for the additional 4/6B of MLPPPoX encapsulation overhead. Consequently, IP payload of 1500B can be transmitted over the bundle but only 1488B can be transmitted over any individual link. In case that an IP packet with the size between 1489B and 1500B needs to be transmitted from the 7750 towards the CPE, this packet would be MLPPPoX fragmented in the 7750 as dictated by the link-mtu. This is irrespective of whether MLPPPoX fragmentation is enabled or disabled (as set by no max-fragment-delay flag).

To entirely avoid MLPPPoX fragmentation in this case, the received MRRU sent by CPE should be lower than the received MRU for the length of the MLPPPoX header (4 or 6 bytes). In this case, for IP packets larger than 1488B, IP fragmentation would occur (assuming that DF flag in the IP header allows it) and MLPPPoX fragmentation would be avoided.

On the 7750 side, it is not possible to set different advertized MRRU and MRU values with the ppp-mtu command. Both MRRU and MRU advertized values adhere to the same configured ppp mtu value.



## LFI Functionality Implemented in LNS

As mentioned in the previous section, LFI on LNS is implemented only on MLPPPoX bundles with a single LCP session.

There are two major tasks associated with LFI<sup>1</sup> on the LNS:

- Executing subscriber QoS in the carrier IOM based on the last mile conditions. The subscriber QoS rates are the last mile on-the-wire rates. Once traffic is QoS conditioned, it is sent to the BB-ISA for further processing.
- Fragmentation and artificial delay (queuing) of the fragments so that high priority packets can be injected in-between low priority fragments (interleaved). This operation is performed by the BB-ISA.

Examine an example to further clarify functionality of LFI. The parameters, conditions and requirements that will be used in our example to describe the desired behavior are the following:

- High priority packets must NOT be delayed for more than 50ms in the last mile due to the transmission delay of the large low priority packets. Considering that tolerated end-to-end VoIP delay must be under 150ms, limiting the transmission delay to 50ms on the last mile link is a reasonable choosing.
- The link between the LNS and LAC is 1Gbps Ethernet.
- The last mile link rate is 256kbps.
- Three packets arrive back-to-back on the network side of the LNS (in the downstream direction). The large 5000B low priority packet P1 arrives first, followed by two smaller high priority packets P2 and P3, each 100B in length. Note that packets P1, P2 and P3 can be originated by independent sources (PCs, servers, etc.) and therefore can theoretically arrive in the LNS from the network side back-to-back at the full network link rate (10Gbps or 100Gbps).
- The transmission time on the internal 10G link between the BB-ISA and the carrier IOM for the large packet (5000B) is 4us while the transmission time for the small packet (100B) is 80ns.
- The transmission time on the 1G link (LNS->LAC) for the large packet (5000B) is 40us while the transmission time for the small packet (100B) is 0.8us.
- The transmission time in the last mile (256kbps) for the large packet is ~150ms while the transmission time for the small packet on the same link is ~3ms.
- Last mile transport is ATM.

---

1. Most of this is also applicable to non-lfi case. The only difference between lfi and non-lfi is that there is no artificial delay performed in non-lfi case.

To satisfy the delay requirement for the high priority packets, the large packets will be fragmented into three smaller fragments. The fragments will be carefully sized so that their individual transmission time in the last mile does not exceed 50ms. After the first 50ms interval, there will be window of opportunity to interleave the two smaller high priority packets.

This entire process is further clarified by the five points (1-5) in the packet route from the LNS to the Residential Gateway (RG).

The five points are:

1. [Last Mile QoS Awareness in the LNS on page 745](#)
2. [BB-ISA Processing on page 747](#)
3. [LNS-LAC Link on page 748](#)
4. [AN-RG Link on page 748](#)
5. [Home Link on page 748](#)

## Last Mile QoS Awareness in the LNS

By implementing MLPPPoX in LNS, we are effectively transferring the traffic treatment functions (QoS/LFI) of the last mile to the node (LNS) that is multiple hops away.

The success of this operation depends on the accuracy at which we can simulate the last mile conditions in the LNS. The assumption is that the LNS is aware of the two most important parameters of the last mile:

- The last mile encapsulation — This is needed for the accurate calculation of the overhead associated of the transport medium in the last mile for traffic shaping and interleaving.
- The last mile link rate — This is crucial for the creation of artificial congestion and packet delay in the LNS.

The subscriber QoS in the LNS is implemented in the carrier IOM and is performed on a per packets basis before the packet is handed over to the BB-ISA. Per packet, rather than per fragment QoS processing will ensure a more efficient utilization of network resources in the downstream direction. Discarding fragments in the LNS would have detrimental effects in the RG as the RG would be unable to reconstruct a packet without all of its fragments.

High priority traffic within the bundle is classified into the high priority queue. This type of traffic is not MLPPPoX encapsulated unless its packet size exceeds the link MTU as described in **MLPPPoX Fragmentation, MRRU and MRU Considerations on page 741**. Low priority traffic is classified into a low priority queue and is always MLPPPoX encapsulated. In case that the high priority traffic becomes MLPPPoX encapsulated/fragmented, the MLPPPoX processing module (BB-ISA) will consider it as low-priority. The assumption is that the high priority traffic is small in size and consequently MLPPPoX encapsulation/fragmentation an degradation in priority can be avoided. The aggregate rate of the MLPPPoX bundle is on-the-wire rate of the last mile as shown in Figure 3.

ATM on-the-wire overhead for non-MLPPPoX encapsulated high priority traffic will include:

- ATM encapsulation (VC-MUX, LLC/NLPID, LLC/SNAP).
- AAL5 trailer (8B).
- AAL5 padding to 48B cell boundary (this makes the overhead dependent on the packet size).
- Multiplication by 53/48 to account for the ATM cell headers.

For low priority traffic which is always MLPPPoX encapsulated, an additional overhead related to MLPPPoX encapsulation and possibly fragmentation must be added (blue arrow in Figure 3). In other words, each fragment carries ATM+MLPPPoX overhead.

Note that we can avoid the 48B boundary padding for all fragments except the last one. This can be done by choosing the fragment length so that it is aligned on the 48B boundary (rounded down if based on max-fragment-delay or rounded up if based on the encapsulation/utilization).

For Ethernet in the last mile, our implementation always assures that the fragment size plus the encapsulation overhead is always larger or equal to the minimum Ethernet packet length (64B).

## BB-ISA Processing

MLPPPoX encapsulation, fragmentation and interleaving are performed by the LNS in BB-ISA. If we refer to our example, a large low priority packet (P1) is received by the BB-ISA, immediately followed by the two small high priority packets (P2 and P3). Since our requirement stipulates that there is no more than 50ms of transmission delay in the last mile (including on-the-wire overhead), the large packet must be fragmented into three smaller fragments each of which will not cause more than 50ms of transmission delay.

The BB-ISA would normally send packets/fragments to the carrier IOM at the rate of 10Gbps. In other words, by default the three fragments of the low priority packet would be sent out of the BB-ISA back-to-back at the very high rate before the high priority packets even arrive in the BB-ISA. In order to interleave, the BB-ISA must simulate the last mile conditions by delaying the transmission of the fragments. The fragments will be paced out of the BB-ISA (and out of the box) at the rate of the last mile. High priority packets will get the opportunity to be injected in front of the fragments while the fragments are being delayed.

As shown in [Figure 46](#) (point 2) the first fragment F1 is sent out immediately (transmission delay at 10G is in the 1us range). The transmission of the next fragment F2 is delayed by 50ms. While the transmission of the second fragment F2 is being delayed, the two high priority packets (P1 and P2 in red) are received by the BB-ISA and are immediately transmitted ahead of fragments F2 and F3. This approach relies on the imperfection of the IOM shaper which is releasing traffic in bursts (P2 and P3 right after P1). The burst size is dependent on the depth of the rate token bucket associated with the IOM shaper.

Note that by the time the second fragment F2 is transmitted, the first fragment F1 has traveled a long way (50ms) on high rate links towards the Access Node (assuming that there is no queuing delay along the way), and its transmission on the last mile link has already begun (if not already completed).

This is not applicable for this discussion, but nonetheless worth noticing is that the LNS BB-ISA also adds the L2TP encapsulation to each packet/fragment. The L2TP encapsulation is removed in the LAC before the packet/fragment is transmitted towards the AN.

## LNS-LAC Link

This is the high rate link (1Gbps) on which the first fragment F1 and the two consecutive high priority packets, P2 and P3, are sent back-to-back by the BB-ISA

(BB-ISA->carrier IOM->egress IOM-> out-of-the-LNS).

The remaining fragments (F2 and F3) are still waiting in the BB-ISA to be transmitted. They are artificially delayed by 50ms each.

Additional QoS based on the L2TP header can be performed on the egress port in the LNS towards the LAC. This QoS is based on the classification fields inside of the packet/fragment headers (DSCP, dot1.p, EXP).

Note that the LAC-AN link is not really relevant for the operation of LFI on the LNS. This link can be either Ethernet (in case of PPPoE) or ATM (PPPoE or PPP). The rate of the link between the LAC and the AN is still considered a high speed link compared to the slow last mile link.

---

## AN-RG Link

Finally, this is the slow link of the last mile, the reason why LFI is performed in the first place. Assuming that LFI played its role in the network as designed, by the time the transmission of one fragment on this link is completed, the next fragment arrives just in time for unblocked transmission. In between the two fragments, we can have one or more small high priority packets waiting in the queue for the transmission to complete.

Note on the AN-RG link in [Figure 46](#) that packets P2 and P3 are ahead of fragments F2 and F3. Therefore the delay incurred on this link by the low priority packets is never greater than the transmission delay of the first fragment (50ms). The remaining two fragments, F2 and F3, can be queued and further delayed by the transmission time of packets P2 and P3 (which is normally small, in our example 3ms for each).

Note that if many low priority packets are waiting in the queue, then they would have caused delay and would have further delayed the fragments that are in transit from the LNS to the LAC. This condition is normally caused by bursts and it should clear itself out over time.

---

## Home Link

High priority packets P2 and P3 are transmitted by the RG into the home network ahead of the packet P1 although the fragment F1 has arrived in the RG first. The reason for this is that the RG must wait for the fragments F2 and F3 before it can re-assemble packet P1.

## Optimum Fragment Size Calculation by LNS

Fragmentation in LFI is based on the optimal fragment size. LNS implementation calculates the two optimal fragment sizes, based on two different criteria:

- Optimal fragment size based on the payload efficiency of the fragment given the fragmentation/transportation header overhead associated with the fragment
  - encapsulation based fragment size.
- Optimal fragment size based on the maximum transmission delay of the fragment set by configuration
  - delay based fragment size.

At the end only one optimal fragment size will be selected. The actual fragments length will be of the optimal fragment size.

- The parameters required to calculate the optimal fragment sizes are known to the LNS either via configuration or via signaling. These, in-advance known parameters are:
  - Last mile maximum transmission delay (max-fragment-delay obtained via CLI)
  - Last mile ATM Encapsulation (in our example the last mile is ATM but in general it can be Ethernet for MLPPPoE)
  - MLPPP encapsulation length (depending on the fragment sequence number format)
  - The last mile on-the-wire rate for the MLPPPoX bundle

Examine closer each of the two optimal fragment sizes.

---

## Encapsulation Based Fragment Size

One needs to be mindful of the fact that fragmentation may cause low link utilization. In other words, during fragmentation a node may end up transporting mainly overhead bytes in the fragment as opposed to payload bytes. This would only intensify the problem that fragmentation is intended to solve, especially on an ATM access link that tend to carry larger encapsulation overhead.

To reduce the overhead associated with fragmentation, the following is enforced in the 7750:

The minimum fragment payload size will be at least 10times greater than the overhead (MLPPP header, ATM Encapsulation and AAL5 trailer) associated with the fragment.

The optimal fragment length (including the MLPPP header, the ATM Encapsulation and the AAL5 trailer) is a multiple of 48B. Otherwise, the AAL5 layer would add an additional 48B boundary padding to each fragment which would unnecessary expand the overhead associated with fragmentation. By aligning all-but-last fragments to a 48B boundary, only the last fragment will potentially contain the AAL5 48B boundary padding which is no different from a non-

fragmented packet. For future reference we will refer to all fragments except for the last fragment as non-padded fragments. The last fragment will obviously be padded if it is not already natively aligned to a 48B boundary.

As an example, calculate the optimal fragment size based on the encapsulation criteria with the maximum fragment overhead of 22B. To achieve >10x transmission efficiency the fragment payload size must be 220B (10\*22B). To avoid the AAL5 padding, the entire fragment (overhead + payload) will be rounded UP on a 48B boundary. The final fragment size will be 288B [22B + 22B\*10 + 48B\_allignment].

In conclusion, an optimal fragment size was selected that will carry the payload with at least 90% efficiency. The last fragment of the packet cannot be artificially aligned on a 48B boundary (it is a natural remainder), so it will be padded by the AAL5 layer. Therefore the efficiency of the last fragment will probably be less than 90% in our example. In the extreme case, the efficiency of this last fragment may be only 2%.

Note that the fragment size chosen in this manner is purely chosen based on the overhead length. The maximum transmission delay did not play any role in the calculations.

For Ethernet based last mile, the CPM always makes sure that the fragment size plus encapsulation overhead is larger or equal to the minimum Ethernet packet length of 64B.

---

## Fragment size based on the max transmission delay

The first criterion in selecting the optimal fragment size based on the maximum transmission delay mandates that the transmission time for the fragment, including all overheads (MLPPP header, ATM encapsulation header, AAL5 overhead and ATM cell overhead) must be less than the configured max-fragment-delay time.

The second criterion mandates that each fragment, including the MLPPP header, the ATM Encapsulation header, the AAL5 trailer and the ATM cellification overhead be a multiple of 48B. The fragment size is rounded down to the nearest 48B boundary during the calculations in order to minimize the transmission delay. Aligning the fragment on the 48B boundary eliminates the AAL5 padding and therefore reduces the overhead associated with the fragment. The overhead reduction will not only improve the transmission time but it will also increase the efficiency of the fragment.

Given these two criteria along with the configuration parameters (ATM Encapsulation, MLPPP header length, max-fragment-delay time, rate in the last mile), the implementation calculates the optimal non-padded fragment length as well as the transmission time for this optimal fragment length.



## **Selection of the Optimum Fragment Length**

So far the implementation has calculated the two optimum fragment lengths, one based on the length of the MLPPP/transport encapsulation overhead of the fragment, the other one based on the maximum transmission delay of the fragment. Both of them are aligned on a 48B boundary. The larger of the two is chosen and the BB-ISA will perform LFI based on this selected optimal fragment length.

## Upstream Traffic Considerations

Fragmentation and interleaving is implemented on the originating end of the traffic. In other words, in the upstream direction the CPE (or RG) is fragmenting and interleaving traffic. There is no interleaving or fragmentation processing in the upstream direction in the 7750. The 7750 will be on the receiving end and is only concerned with the reassembly of the fragments arriving from the CPE. Fragments will be buffered until the packet can be reconstructed. If all fragments of a packet are not received within a preconfigured timeframe, the received fragments of the partial packet will be discarded (a packet cannot be reconstructed without all of its fragments). This time-out and discard is necessary in order to prevent buffer starvation in the BB-ISA. Two values for the time-out can be configured: 100ms and 1s.

---

## Multiple Links MLPPPoX With No Interleaving

Interleaving over MLPPPoX bundles with multiple links will not be supported. However, fragmentation is supported.

In order to preserve packet order, all packets on an MLPPPoX bundle with multiple links will be MLPPPoX encapsulated (monotonically increased sequence numbers).

We will not support multiclass MLPPP (RFC 2686, *The Multi-Class Extension to Multi-Link PPP*). Multiclass MLPPP would require another level of intelligent queuing in the BB-ISA which we do not have.

---

## MLPPPoX Session Support

The following session types in the last mile will be supported:

- MLPPPoE — Single physical link or multilink. The last mile encapsulation is Ethernet over copper (This could be Ethernet over VDSL or HSDSL). The access rates (especially upstream) are still limited by the xDSL distance limitation and as such interleaving is required on a slow speed single link in the last mile. It is possible that the last mile encapsulation is Ethernet over fiber (FTTH) but in this case, users would not be concerned with the link speed to the point where interleaving and link aggregation is required.

Finally, this is the slow link of the last mile, the reason why LFI is performed in the first place. Assuming that LFI played its role in the network as designed, by the time the transmission of one fragment on this link is completed, the next fragment arrives just in time for unblocked transmission. In between the two fragments, we can have one or more small high priority packets waiting in the queue for the transmission to complete.

We can see on the AN-RG link in Figure 2 that packets P2 and P3 are ahead of fragments F2 and F3. Therefore the delay incurred on this link by the low priority packets is never greater than the transmission delay of the first fragment (50ms). The remaining two fragments, F2 and F3, can be queued and further delayed by the transmission time of packets P2 and P3 (which is normally small, in our example 3ms for each).

Note that if many low priority packets were waiting in the queue, then they would have caused delay for each other and would have further delayed the fragments in transit from the LNS to the LAC. This condition is normally caused by bursts and it should clear itself out over time.

- MLPPP(oEo)A — A single physical link or multilink. The last mile encapsulation is ATM over xDSL.

Some other combinations are also possible (ATM in the LAST mile, Ethernet in the aggregation) but they all come down to one of the above models that are characterized by:

- Ethernet or ATM in the last mile.
- Ethernet or ATM access on the LAC.
- LPPP/PPPoE termination on the LNS

## Session Load Balancing Across Multiple BB-ISAs

PPP/PPPoE sessions are by default load balanced across multiple BB-ISAs (max 6) in the same group. The load balancing algorithm considers the number of active session on each BB-ISA in the same group<sup>2</sup>.

With MLPPPoX, it is important that multiple sessions per bundle be terminated on the same LNS BB-ISA. This can be achieved by per tunnel load balancing mode where all sessions of a tunnel are terminated in the same BB-ISA. Per tunnel load balancing mode is mandatory on LNS BB-ISAs that are in the group that supports MLPPPoX.

On the LAC side, all sessions in an MLPPPoX bundle are automatically assigned to the same tunnel. In other words an MLPPPoX bundle is assigned to the tunnel. There can be multiple tunnels created between the same pair of LAC/LNS nodes.

- 
2. The load balancing algorithm does not take into account the number of queues consumed on the carrier IOM. Therefore a session can be refused if queues are depleted on the carrier IOM even though the BB-ISA may be lightly loaded in terms of the number of sessions that is hosting.

## BB-ISA Hashing Considerations

All downstream traffic on an MLPPPoX bundle with multiple links is always MLPPPoX encapsulated. Some traffic is fragmented and served in a octet oriented round robin fashion over multiple member links. However, fragments are never delayed in case that the bundle contains multiple links.

In a per fragment/packet load sharing algorithm, there is always the possibility that there is uneven load utilization between the member links. A single link overload will most likely go unnoticed in the network all the way to the Access Node. The access node is the only node in the network that actually has multiple physical links connected to it. All other session-aware nodes<sup>3</sup> (LAC and LNS) only see MLPPPoX as a bundle with multiple sessions without any mechanism to shape traffic per physical link.

If one of the member sessions is perpetually overloaded by the LNS, traffic will be dropped in the last mile since the corresponding physical link cannot absorb traffic beyond its physical capabilities. This would have detrimental effects on the whole operation of the MLPPPoX bundle. To prevent this perpetual overloading of the member links that can be caused by per packet/fragment load balancing scheme, the load balancing scheme that takes into account the number of octets transmitted over each member link. The octet counter of a new link will be initialized to the lowest value of any existing link counter. Otherwise the load balancing mechanism would show significant bias towards the new link until the byte counter catches up with the rest of the links.

---

## Last Mile Rate and Encapsulation Parameters

The last mile rate information along with the encapsulation information is used for fragmentation (to determine the maximum fragment length) and interleaving (delaying fragments in the BB-ISA). In addition, the aggregate subscriber rate (aggregate-rate-limit) on the LNS is automatically adjusted based on the last mile link rate and the number of links in the MLPPPoX bundle.

### Downstream Data Rate in the Last Mile

The subscriber aggregate rates (agg-rate-limit) used in (H)QoS on the carrier IOM and in the BB-ISA (for interleaving) must be wire based in the last mile. This rule applies equally to both, the LAC and LNS.

The last mile on-the-wire rates of the subscriber can be submitted to the LAC and the LNS via various means. Here is the break down on how the last mile wire rates will be passed to each entity:

---

3. Other nodes in this case being 7750s. Other vendors may have the ability to condition (shape) traffic per session.

## LAC

The last mile link rate is taken via the following methods in the order of listed priority:

- LUDB — rate-down command under the host hierarchy in LUDB.
- RADIUS Alc-Access-Loop-Rate-Down VSA. Although this VSA is stored in the state of plain PPP(oE) sessions (MLPPPoX bundled or not), it is applicable only to MLPPPoX bundles.
- PPPoE tags — Vendor Specific Tags (RFC 2516, *A Method for Transmitting PPP Over Ethernet (PPPoE)*); tag type 0x0105; tag value is Enterprise Number 3561 followed by the TLV sub-options as specified in TR-101 -> Actual Data Rate Downstream 0x82)

As long as the link rate information is available in the LAC, it will always be passed to the LNS in the ICRQ message using the standard L2TP encoding. This cannot be disabled.

In addition, an option is available to control the source of the rate information can be conveyed to the LNS via TX Connect Speed AVP in the ICCN message. This can be used for compatibility reasons with other vendors that can only use TX Connect Speed to pass the link rate information to the LNS. By default, the maximum port speed (or the sum of the maximum speeds of all member ports in the LAG) will be reported in TX Connect Speed. Unlike the rate conveyed in ICRQ message, The TX Connect Speed content is configurable via the following command:

```
config>subscriber-management>
  sla-profile <name>
    egress
      report-rate agg-rate-limit | scheduler <scheduler-name> | pppoe-actual-rate
    | rfc5515-actual-rate
```

The report-rate configuration option will dictate which rate will be reported in the TX Connect Speed as follows:

- agg-rate-limit => statically configured agg-rate-limit value or RADIUS QoS override will be reported
- scheduler <scheduler-name> => virtual schedulers are not supported in MLPPPoX
- pppoe-actual-rate => rate taken from PPPoE Tags will be reported. Note that rate reported via RFC5515 can still be different if the source for both methods is not the same.
- rfc5515-actual-speed => the rate is taken from RFC5515.

The RFC 5515 relies on the same encoding as PPPoE tags (vendor id is ADSL Forum and the type for Actual Data Rate Downstream is 0x82). Note that the two methods of passing the line rate to the LNS are using different message types (ICRQ and ICCN).

The LAC on the 7750 is not aware of MLPPPoX bundles. As such, the aggregate subscriber bandwidth on the LAC is configured statically via usual means (sub-profile, scheduler-policy) or dynamically modified via RADIUS. The aggregate subscriber (or MLPPPoX bundle) bandwidth on the LAC is not automatically adjusted according to the rates of the individual links in the

bundle and the number of the links in the bundle. As such, an operator must ensure that the statically provided rate value for aggregate-rate-limit is the sum of the bandwidth of each member link in the MLPPPoX bundle. The number of member links and their bandwidth must be therefore known in advance. The alternative is to have the aggregate rate of the MLPPPoX bundle set to a high value and rely on the QoS treatment performed on the LNS.

### LNS

The sources of information for the last mile link rate on the LNS will be taken in the following order:

- LUDB (during user authentication phase, same as in LAC)
- RADIUS (same as in LAC)
- ICRQ message — Actual Data Downstream Rate (RFC 5515)
- ICCN message — TX Connect Speed

There will be no configuration option to determine the priority of the source of information for the last mile link rate. TX Connect Speed in ICCN message will only be taken into consideration as a last resort in absence of any other source of last mile rate information.

Once the last mile rate information is obtained, the subscriber aggregate rate (aggregate-rate-limit) will be automatically adjusted to the minimum value of:

- The smallest link speed in the MLPPPoX bundle multiplied by the number of links in the bundle.
- Statically configured aggregate-rate-limit

The link speed of each link in the bundle must be the same, i.e. different link speeds within the bundle are not supported. In the case that we receive different link speed values for last mile links within the bundle, we will adopt the minimum received speed and apply it to all links.

In case that the obtained rate information from the last mile for a session within the MLPPP bundle is out of bounds (1Kbps to 100Mbps), the session within the bundle will be terminated.

### Encapsulation

Wire-rates are dependent on the encapsulation of the link to which they apply. The last mile encapsulation information can be extracted via various means.

### LAC

- Static configuration via LUDB.
- RADIUS — Alc-Access\_Loop-Encap-Offset VSA.

- PPPoE tags — Vendor Specific Tags (RFC 2516; tag type 0x0105; tag value is Enterprise Number 3561 followed by the TLV sub-options as specified in TR-101 -> Actual Data Rate Downstream 0x82).

The LAC will pass the line encapsulation information to the LNS via ICRQ message using the encoding defined in the RFC 5515.

LNS

The LNS will extract the encapsulation information in the following order:

- Static configuration via LUDB.
- RADIUS — Alc-Access-Loop-Encap-Offset VSA.
- ICRQ message (RFC 5515)

In case that the encapsulation information is not provided by any of the existing means (LUDB, RADIUS, AVP signaling, PPPoE Tags), then by default pppoa-null encapsulation will be in effect. This applies to LAC and LNS.

---

## Link Failure Detection

The link failure in the last mile is detected via the expiration of session keepalives (LCP). The LNS will tear down the session over the failed link and notify the LAC via a CDN message.

---

## CoA Support

CoA request for the subscriber aggregate-rate-limit change is honored on the LAC and the LNS.

CoA for the rate change of an individual link within the bundle is supported through the same VSA that can be used to initially assign the rate parameter to each member link. This is supported only on LNS. The rate override via CoA is applied to all active link members within the bundle.

Change of the access link parameters via CoA is supported in the following fashion:

- Change of access loop encap: refused (NAK)
- Change of access loop rate down:
- On L2TP LAC session: refused (NAK). On LAC the access loop rate down is not locally used for any rate limiting function but instead it is just passed to the LNS at the beginning when the session is first established. Mid-session changes on LAC via CoA are not propagated to the LNS.



- On L2TP LNS session:
  - Plain session: ignored. The rate is stored in the MIB table but no rate limiting action is taken. In other words, this parameter is internally excluded from rate calculations and advertisements. However, it is shown in the output of the relevant show commands.
- Bundle session: applied on all link sessions. The aggregate rate limit of the bundle is set to the minimum of the:
  - CoA obtained local loop down rate multiplied by the number of links in the bundle
  - The aggregate rate limit configured statically or obtained via CoA.
- Fragment length will be affected by this change. In case that interleaving is enabled on a single link bundle, the interleave interval will be affected.
- Non-L2TP: ignored. The rate is stored in the MIB table but no rate limiting action is taken. In other words, this parameter is internally excluded from rate calculations and advertisements. However, it will be shown in the output of the relevant show commands.

Similar behavior is exhibited if at midsession, the parameters are changed via LUDB with the exception of the rate-down parameter in LAC. If this parameter is changed on the LAC, all sessions are disconnected.

---

## Accounting

Accounting counters on the LNS include all packet overhead (wire overhead from the last mile). There is only one accounting session per bundle.

On the LAC, there is one accounting session per pppoe session (link).

In tunnel-accounting mode there is one accounting session per link.

On LNS only the stop-link of the last link of the bundle will carry all accounting data for the bundle.

---

## Filters and Mirroring

Filters and mirrors (LI) are not supported on an MLPPPoX bundle on LAC. However, filters and ip-only mirror type are supported on the LNS.

## PTA Considerations

Locally terminated MLPPPoX (PTA) solution is offered based on the LAC and the LNS hosted in the same system. An external loop (or VSM2) is used to connect the LAC to the LNS within the same box. The subscribers will be terminated on the LNS.

---

## QoS Considerations

---

### Dual-Pass

HQoS and LFI are performed in two stages that involve double traversal (dual-pass) of traffic through the carrier IOM and the BB-ISA. The following are the functions performed in each pass:

- In the first pass through the carrier IOM, traffic is marked (dot1p bits) as high or low priority. This will play crucial role in the execution of LFI in the BB-ISA.
  - In the first pass through the BB-ISA this prioritization from the 1st step, will be an indication (along with the internally calculated fragment size) of whether the traffic will be interleaved (non MLPPP encapsulated) or not (MLPPP encapsulated). Consequently the BB-ISA will add the necessary padding related to last mile wire overhead to each packet. This padding will be used in the second pass on the carrier IOM to perform last mile wire based QoS functions.
  - In the second pass through the carrier IOM, the last mile wire based HQoS will be performed based on the padding added in the first pass through the BB-ISA.
  - In the second pass through the BB-ISA, previously added overhead will be stripped off and LFI/MLPPP encapsulation functions will be performed.
- 

### Traffic Prioritization in LFI

The delivery of high priority traffic within predefined delay bounds on a slow speed last mile link is ensured by proper QoS classification and prioritization. High priority traffic will be interleaved with low priority fragments on a single link MLPPPoX bundle with LFI enabled. The classification of traffic into proper (high or low priority) forwarding class is performed on the downstream ingress interface. However, traffic can be re-classified (re-mapped into another forwarding class) on the egress access interface of the carrier IOM, just before packets are transmitted to the BB-ISA for MLPPPoX processing. This can be achieved via QoS sap-egress policy referenced in the LNS sla-profile.

The priority of the forwarding class in regular QoS (on IOM) is determined by the properties<sup>4</sup> of the queue to which the forwarding class is mapped. In contracts, traffic prioritization in LFI domain (in BB-ISA) is determined by the outer dot1p bits that are set by the carrier IOM while transmitting packets towards the BB-ISA. The outer dot1p bits are marked based on the forwarding class information determined by classification/re-classification on ingress/carrier IOM. This marking of outer dot1p bits in the Ethernet header between the carrier IOM and the BB-ISA is fixed and defined in the default sap-egress LNS ESM policy 65537. The marking definition is as follows:

```
FC be -> dot1p 0
FC l2 -> dot1p 1
FC af -> dot1p 2
FC l1 -> dot1p 3
FC h2 -> dot1p 4
FC ef -> dot1p 5
FC h1 -> dot1p 6
FC nc -> dot1p 7
```

In LFI (on BB-ISA), dot1p bits [0,1,2 and 3] are considered low priority while dot1p bits (4,5,6 and 7) are considered high priority. Consequently, forwarding classes BE, L2, AF and L1 are considered low priority while forwarding classes H2, EF, H1 and NC are considered high priority. High priority traffic<sup>5</sup> will be interleaved with low priority traffic.

The following describes the reference points in traffic prioritization for the purpose of LFI in the 7750:

- Classification on downstream ingress interface (entrance point into the 7750) - packets can be classified into one of the following eight forwarding classes: be, l2, af, l1, h2, ef, h1 and nc. Depending on the type of the ingress interface (access or network), traffic can be classified based on dot1p, exp, DSCP, TOS bits or ip-match criteria (dscp, dst-ip, dst-port, fragment, src-ip, src-port and protocol-id).
- Re-classification on downstream access egress interface between the carrier IOM and the BB-ISA - in the carrier IOM, downstream traffic can be re-classified into another forwarding class, just before it is forwarded to the BB-ISA. Re-classification on access egress is based on the same fields as on ingress except for the dot1p and exp bits since Ethernet or MPLS headers from ingress are not carried from ingress to egress.
- Marking on downstream access egress interface between the carrier IOM and the BB-ISA - once the forwarding class is available on the carrier IOM in the egress direction (towards BB-ISA), it will be used to mark outer dot1p bits in the new Ethernet header that will be used to transport the frame from the carrier IOM to the BB-ISA. The marking of the dot1p bits on the egress SAP between the carrier IOM and the BB-ISA cannot be changed for MLPPPoX even if the no qos-marking-from-sap command is configured under the sla-profile on egress.

---

4. Expedited, non-expedited queue type, CIR and PIR rates.

5. Assuming that the packet size does not exceed maximum fragment size.

## Shaping Based on the Last Mile Wire Rates

Accurate QoS, amongst other things, require that the subscriber rates in the first mile on an MLPPPoX bundle be properly represented in the LNS. In other words, the rate limiting functions in the LNS must account for the last mile on-the-wire encapsulation overhead. The last mile encapsulation can be Ethernet or ATM.

For ATM in the last mile, the LNS will account for the following per fragment overhead:

- PID
- MLPPP encapsulation header
- ATM Fixed overhead (ATM encap + fixed AAL5 trailer)
- 48B boundary padding as part of AAL5 trailer
- 5B per each 48B of data in ATM cell.

In case of Ethernet encapsulation in the last mile, the overhead will be:

- PID
- MLPPP header per fragment
- Ethernet Header + FCS per fragment
- Preamble + IPG overhead per fragment

The encap-offset command under the sub-profile egress CLI node will be ignored in case of MLPPPoX. MLPPPoX rate calculation will be by default always based on the last mile wire overhead.

The HQoS rates (port-scheduler, aggregate-rate-limit and scheduler) on LNS are based on the wire overhead of the entity to which the HQoS is applied. For example, if the port-scheduler is managing bandwidth on the link between the BB-ISA and the carrier IOM, then the rate of such scheduler will account for the q-in-q Ethernet encapsulation on that link along with the preamble and inter packet gap (20B).

While virtual schedulers (attached via sub-profile) are supported on LNS for plain PPPoX sessions, they are not supported for MLPPPoX bundles. Only aggregate- rate-limit along with the port-scheduler can be used in MLPPPoX deployments.

## Downstream Bandwidth Management on Egress Port

Bandwidth management on the egress physical ports (Physical Port 1 and Physical Port 2 in Figure 8) is performed at the egress port itself on the egress IOM instead on the carrier IOM. By default, the forwarding class (FC) information is preserved from network ingress to network egress. However, this can be changed via QoS configuration applied to the egress SAP of the carrier IOM towards the BB-ISA.

L2TP traffic originated locally in LNS can be marked via the router/service vprn->sgt-qos hierarchy.

---

## Sub/Sla-Profile Considerations

### Sub-profile

In the MLPPPoX case on LNS, multiple sessions are tied into the same subscriber aggregate-rate-limit via a sub-profile. The consequence is that the aggregate rate of the subscriber can be adjusted dynamically depending on the advertized link speed in the last mile and the number of links in the bundle. Note that shaping in the LNS is performed per the entire MLPPPoX bundle (subscriber) rather than per individual member links within the bundle. The exception is obviously a MLPPPoX bundle with the single member link (interleaving case) where the relationship between the session and the MLPPPoX bundle is 1:1.

In the LAC, the subscriber aggregate rate cannot be dynamically changed based on the number of links in the bundle and their rate. The LAC has no notion of MLPPPoX bundles. However, multiple sessions that in reality belong to an MLPPPoX bundle under the subscriber are shaped as an aggregate (agg-rate-limit under the sub-profile). This in essence yields the same shaping behavior as on LNS.

### Sla-profile

Sessions within the MLPPPoX bundle in LNS share a single sla-profile instances (queues).

In the LAC, as long as the sessions within the subscriber6 are on the same SAP, they can also share the same sla-profile. This will be the case in MLPPPoX.

The manner in which sub/sla-profile are applied to MLPPPoX bundles and the individual sessions within results in aggregate shaping per MLPPPoX bundle as well as allocation of unique set of queues per MLPPPoX bundle. This is valid irrespective of the location where shaping is executed (LAC or LNS). Other vendors may have implemented shaping per session within the bundle and this is something that needs to be taken into consideration during the migration process.

## Example of MLPPPoX Session Setup Flow

### LAC behavior

- A new PPP(oEoA) session request will arrive to the LAC (PADI or LCP Conf Req).
- The LAC will negotiate PADx session if applicable.
- The LAC may negotiate MLPPPoX LCP phase with its own endpoint discriminator, or it may reject MLPPPoX specific options in LCP if MLPPPoX on the LAC is disabled (i.e. no accept-mrru in the LAC's ppp-policy). If MLPPPoX options (seq num header format, ED, MRRU) are rejected, the assumption is that the client will renegotiate plain PPP(oEoA) session with the LAC.
- Once LCP (MLPPPoX capable or not) is negotiated, the session will be authenticated (PAP/CHAP).
- Upon successful authentication, an L2TP tunnel will be identified to which the session belongs.
- If the session is a non-L2TP session (PTA MLPPPoX capable session for which the tunnel cannot be determined), the session will be terminated.
- Otherwise, the QoS constructs will be created for the subscriber hosts: the session will be assigned to a sub/sla-profiles.
- The session LCP parameters will be sent to the LNS via call management messages.
- Note that if another LCP session is requested on the same bundle, the LAC will create a new LCP session and join this session to the existing subscriber as another host. In other words, the LAC is bundle agnostic and the two sessions will appear as two hosts under the same subscriber.

The following assumes that MLPPPoX is configured on the LNS under the L2TP group or the tunnel hierarchy.

### LNS behavior

- The LNS have the option to accept the LCP parameters or to reject them and start renegotiating LCP parameters directly with the client.
- If the LNS choose to renegotiate LCP parameters with the client directly, this renegotiation will be completely transparent to the LAC by the means of a T-bit (control vs. data) in the L2TP header. LCP will be renegotiated on the LNS with all the options necessary to support MLPPPoX. Note that Endpoint Discriminator is not mandatory in the MLPPPoX negotiation. If the client rejects it, the LNS must still be able to negotiate MLPPPoX capable session (same is valid for the LAC). If the client's endpoint discriminator is invalid (bad format, invalid class, etc.), the 7750 will not negotiate MLPPPoX and instead a plain PPP session will be created.
- If the LNS is configured to accept the LCP Proxy parameters, the LNS will determine the capability of the client.

If there is no indication of MLPPPoX capability in the Proxy LCP (not even in the original ConfReq), the LNS may accept plain (non MLPPPoX capable) LCP session or renegotiate from scratch the non MLPPPoX capable session.

If there is an indication of MLPPPoX capability in the Proxy LCP (either completely negotiated on the LAC or at least attempted from the client), the LNS will try to either accept the MLPPPoX negotiated session by the LAC or renegotiate the MLPPPoX capable session directly with the client.

If the LCP Proxy parameters with MLPPPoX capability are accepted by the LNS, then the endpoint as negotiated on the LAC will also be accepted.

- Once the MLPPPoX capable LCP session is negotiated or accepted, authentication can be performed on the LNS. Authentication on the LNS can be restarted (CHAP challenge/response with the client), or accepted (chap challenge/response accepted and verified by the LNS via RADIUS).
- If the authentication is successful, depending on the evaluation of the parameters negotiated up to this point a new MLPPPoX bundle will be created or an existing MLPPPoX bundle will be joined. In case that a new bundle is established, the QoS constructs for the subscriber(-host) will be created (sub/sla-profile). Session negotiation will advance to IPCP phase.
- The decision whether a new session should join an existing MLPPPoX bundle, or trigger creation of a new one is governed by RFC 1990, *The PPP Multilink Protocol (MP)*, section 5.1.3, page 16, cases 1,2,3, and 4.
- Note that interleaving is supported only on MLPPPoX bundles with single session in them.

## Other Considerations

- IPv6 is supported.
- AA is supported at LNS where full IP packets can be redirected via AA policies.
- Intra-chassis redundancy is supported:
  - CPM statefull failover
  - BB-ISA — non-stateful failover



## Configuration Notes

MLPPP in subscriber management context is supported only over ATM, Ethernet over ATM or plain Ethernet transport (MLPPPoX). Native MLPPP over PPP/HDLC links is supported outside of the subscriber management context on the ASAP MDA.

MLPPPoX is supported only on LNS.

Interleaving is supported only on MLPPPoX bundles with a single member link. If more than one link is present in an MLPPPoX bundle, the interleaving will be automatically disabled and a SNMP trap will be generated. The MIB for this even is defined as `tmnxMlpppBundleIndicatorsChange`.

If MLPPPoX is enabled on LNS, the load balancing mode between the BB-ISAs within the group should be set to per tunnel. This will ensure that all sessions of the same MLPPPoX bundle are terminated on the same BB-ISA. On the LAC, sessions of the same bundle are setup in the same tunnel.

Virtual schedulers are not supported on MLPPPoX tunnels on LNS. However, aggregate-rate-limit is supported.

The aggregate-rate-limit on LNS will be automatically adjusted to the minimum value of:

- configured aggregate-rate-limit
- minimum last mile rate (obtained via LUDB, RADIUS or PPPoE tags) multiplied by the number of links in the bundle.

The aggregate-rate-limit on the LAC is not adjusted automatically. Therefore, if configured it should be set to a high value and thus the traffic treatment should rely on QoS performed on the LNS.

The rate (rate-down information) of the member links within the bundle must be the same. Otherwise the lowest rate is selected and applied to all member links.

A single CoA for a rate change (Alc-Access-Loop-Rate-Down) of an individual link in an MLPPPoX bundle will modify rates of all links in the bundle. This is applicable on LNS only.

The range of supported last mile rate (rate-down information) for the member links on an MLPPPoX session is 1kbps — 100mbps. On the LNS the last mile rate can be obtained:

- From the LAC via Tx-Connect-Speed AVP or by standard L2TP encoding as described in the RFC 5515, *Layer 2 Tunneling Protocol (L2TP) Access Line Information Attribute Value Pair (AVP) Extensions*.
- From the LAC via LUDB or RADIUS
- Directly on the LNS via LUDB or RADIUS.

The session will fail to come up if the obtained rate-down information is outside of the allowable range (1kbps — 100mbps).

A session within the MLPPPoX bundle will be terminated if the rate-down information for the session is out of bounds (1Kbps — 100Mbps).

If a member link in the last mile fails, traffic will be blackholed until the LNS is notified of this failure. The failure detection in the LNS relies on PPP keepalives.

Shaping is performed per MLPPPoX bundle and not individually per member links.

If encapsulation overhead associated with fragmentation is too large in comparison to payload, the fragments will be sized based on the encapsulation overhead (to increase link efficiency) rather than on maximum transmission delay.

There can be only a single MLPPPoX bundle per subscriber.

MLPPPoX bundles and non-MLPPPoX (plain L2TP PPPoE) sessions cannot coexist under the same subscriber.

Filters and mirrors (LI) are not supported on MLPPPoX bundles on LAC.

**ip-only** type mirrors are supported on MLPPPoX bundles.

In MLPPP scenario, downstream traffic is traversing Carrier IOM and BB-ISA twice. This is referred to as dual-pass and effectively cuts the throughput for MLPPP in half (for example, 5Gbps of MLPPP traffic on a 10Gbps capable BB-ISA).