7 Connectivity configurations

7.1 Introduction

7.2 VLAN configurations

7.3 Tunneling configurations

7.4 Multicast configurations

7.4.1 Introduction

7.4.2 Multicast based on VLAN and/or MAC group address

7.4.3 Multicast based on combined VLAN and group address with no authorization control

7.4.4 Multicast based on combined VLAN and group address with authorization control

7.4.5 Multicast based on multicast LLID

7.4.5.1 Operation outline

In the multicast mode based on multicast LLID (mLLID), the OLT forwards downstream multicast packets on the mLLID with active group membership.

A multicast group membership may be *client-controlled* or *server-controlled*. A client-controlled group membership (sometimes refered to as a *dynamic multicast session*) is initiated by multicast clients that independently issue requests to join or leave a multicast group. A server-controlled group membership (sometimes refered to as *static multicast session*) is initiated and configured by a multicast server or NMS without any explicit input from multicast clients.

From the OLT perspective, an mLLID represents a logical channel that delivers frames to a set of ONUs. The OLT configures the ONU with the mLLID (see 7.4.5.3.1). The OLT controls the intra-ONU forwarding of downstream multicast packets to specific UNI ports through provisioning of necessary classification and forwarding rules.

In some configurations, the logical channel formed by the mLLID is dedicated to a single multicast session. In such configurations, the mLLID value uniquely identifies an individual multicast session and the ONU classification rules may classify multicast frames solely by the mLLID value.

In other configurations, an mLLID logical channel is allocated for a set of multicast sessions. In such configurations, an individual multicast session is identified by a combination of mLLID value and values of some other fields, typically IP Group DA and/or IP SA. Correspondingly, the ONU classification rules may require multiple fileds to classify frames belonging to individual multicast sessions.

To configure classification and forwarding rules for a specific multicast session, the OLT specifies a set of desitination UNI ports for this sesssion. In scenarios where UNI ports belonging to a given multicast session are known to the OLT, the OLT performs the *port-based multicast control* (see 7.4.5.3.2). However, in some scenarios (e.g., when using the client-controlled multicast), the OLT is only aware of the MAC addresses of the multicast clients. In this case, the OLT uses the *MAC-based multicast control* method (see 7.4.5.3.3).

7.4.5.2 IGMP-based and MLD-based multicast control

When the client-controlled multicast group membership method is utilized, it relies on either IGMP or MLD protocols. The ONU does not proxy or snoop IGMP/MLD messages to track IP multicast group membership and has no IP multicast control protocol awareness. In the upstream direction, the ONU forwards IGMP/MLD control messages received from the multicast clients to the OLT after adding appropriate encapsulation parameters as configured by the OLT. All processing of IGMP/MLD control messages and tracking of IP multicast group membership are centralized and performed by the OLT.

7.4.5.2.1 ONU requirements

In the upstream direction, the ONU shall forward all IGMP/MLD control messages received at the UNI to the ONU_MDI using a provisioned unicast ESP. The Modifier block of the ESP may be configured to add a VLAN Tag to the multicast control frame prior to forwarding the multicast control frame to the ONU_MDI.

In the downstream direction, multicast-group-specific IGMP/MLD control frames are forwarded according to the forwarding rules configured on the ONU. These IGMP/MLD control frames are not tagged.

7.4.5.2.2 OLT requirements

When the OLT receives a *JOIN* request for a specific IP multicast session from a multicast client connected to a specific UNI port, it performs one of the following actions:

a) If OLT does not know the instance of the UNI port to which the multicast client is connected, it shall query the ONU to determine an instance of a UNI port on which the given client's MAC address has been learned.

b) If the ONU is not already configured to receive the requested IP multicast session, the OLT shall provision the mLLID (see 7.4.5.3.1) and shall add a new classification and forwarding rule to forward the requested multicast session to the specific UNI port (see 7.4.5.3.2).

c) If the ONU is already receiving the requested multicast session, but the given UNI port is not configured to receive the multicast session, the OLT shall modify the existing classification and forwarding rule to include the additional UNI port into the existing multicast group. The rule modification involved deleting provisioning of a new rule and then deleting the old rule (see .

d) If the given UNI port is already configured to receive the requested IP multicast session, the OLT takes no action.

In some implementations, when the OLT receives the first *JOIN* request for a specific IP multicast session from a multicast client connected to a specific UNI port, the OLT verifies whether this UNI port is authorized to receive the requested IP multicast session. In such scenario, the OLT provisions the mLLID and the necessary classification and forwarding rules in the ONU only if the UNI port is authorized to receive the multicast session. The method used to authorize the UNI ports is outside of scope of this standard.

If the IP multicast session requested by a client does not exist in the OLT (i.e., the requested multicast stream is not being currently forwarded to any multicast clients on the given EPON), the OLT shall provision a local multicast-bearing ESP that forwards multicast traffic identified by the requested IP multicast address to the same mLLID that is provisioned on the ONUs to receive this multicast stream.

When the OLT determines that there are no multicast clients for an IP multicast session connected to an ONU UNI port, the OLT shall modify the associated classification and forwarding rules at the ONU to stop forwarding the indicated multicast session to the UNI port (see 7.4.5.3.3).

When the OLT determines that there are no multicast clients for an IP multicast session connected to any of the UNI ports on an ONU, the OLT shall delete the associated classification and forwarding rule (see 7.4.5.3.3).

When the OLT determines that there are no multicast clients connected to any of the UNI ports on an ONU for any of IP multicast sessions being delivered on a specific mLLID, in addition to deleting the classification and forwarding rules associated with these IP multicast sessions, the OLT shall configure the ONU to delete the mLLID used to deliver these IP multicast sessions (see 7.4.5.3.1).

7.4.5.3 Provisioning of multicast forwarding

The OLT provisions multicast forwarding either in response to multicast clients' requests to join a specific multicast group, in case of client-controlled multicast group membership, or in response to a NMS request, in case of server-controlled multicast group membership. In either case, provisioning of LLID-based multicast forwarding involves the following two steps:

— Configuring inter-ONU multicast per 7.4.5.3.1, and

— Configuring intra-ONU multicast per 7.4.5.3.2 or 7.4.5.3.3

7.4.5.3.1 Provisioning of inter-ONU multicast based on mLLID

The inter-ONU multicast is provisioned using the *acConfigMulticastLlid* (0xD9/0x01-07) action (see 14.4.5.2.7). Using this action, the OLT may add a new mLLID to an ONU, delete a specific mLLID from the ONU, or delete all mLLIDs from the ONU.

Deleting one or all mLLIDs from the ONU shall not modify or delete any of the rules provisioned into Classifier/Modifier using the *aRuleSetConfig* (0xD7/0x05-01) attribute.

The OLT may retrieve a list of all registered mLLIDs in the ONU using the attribute *aOnuMulticastLlid* (0xD7/0x01-10) (see 14.4.3.2.15).

7.4.5.3.2 Provisioning of intra-ONU multicast using port-based multicast group control

To add a port to a specific multicast group, the OLT uses the attribute *aRuleSetConfig* (0xD7/0x05-01) (see 14.4.3.6.1). To replicate a multicast frame to multiple UNI ports, the *aRuleSetConfig* attribute includes multiple *sResult* sub-attributes with the *sFrameAction* set to *QUEUE* (see 14.4.3.6.1.2).

To add the *first* multicast port to a given multicast group, the OLT shall generate the *aRuleSetConfig* attribute that includes:

— One or more *sClause* sub-attributes necessary to match frames belonging to specific multicast flow

— A single *sResult* sub-attribute with the action set to *QUEUE*, directing traffic to a specific queue associated with a specific UNI port instance.

To add another UNI port to a multicast group already existing in the ONU, the OLT shall generate a new *aRuleSetConfig* attribute, that contains an additional *sResult* sub-attribute with the action set to *QUEUE*, but is otherwise identical to the existing rule for the given multicast group.

The OLT shall not generate a rule with multiple *sResult* sub-attributes pointing to the same instance of a UNI port. The ONU shall reject a rule with multiple *sResult* sub-attributes pointing to the same instance of a UNI port.

To ensure that the multicast session to the existing multicast clients is not interrupted, the OLT shall generate the new *aRuleSetConfig* attribute first. Upon receiving a response from the ONU that the new rule was configured successfully, the OLT shall delete the old *aRuleSetConfig* attribute.

To delete a UNI port from an existing multicast group in the given ONU, the OLT shall generate a new *aRuleSetConfig* attribute, that does not contain the *sResult* sub-attribute forwarding traffic to the port being deleted, but is otherwise identical to the existing rule for the given multicast group. Here as well, the OLT shall generate the new *aRuleSetConfig* attribute first and delete the old *aRuleSetConfig* attribute upon receiving a response from the ONU.

When a new rule is added at the ONU and the old rule is deleted after that, and if the new rule contains the same *sClause* sub-attributes and some of the *sResult* clauses forwarding traffic to the same queues as the old rule, the Classifier at the ONU shall not discard any frames destined to these queues, i.e., the multicast flows to the existing and remaining multicast clients are not interrupted when other clients are added or deleted.

To delete all UNI ports from an existing multicast group in the given ONU, the OLT shall delete the associated *aRuleSetConfig* attribute entirely. This effectively deletes the entire multicast group in the given ONU.

7.4.5.3.3 Provisioning of intra-ONU multicast using MAC-based multicast group control

The MAC-based multicast group control is used in situations where only the MAC addresses of multicast clients are known to the OLT. The MAC-based multicast group control is a two-step process:

1) The OLT shall query the ONU to find out an instance of a UNI port on which the given MAC address has been learned.

2) The OLT shall add this instance of UNI port to the given multicast group using the method described in 7.4.5.3.2.

To retrieve the instance of the UNI port on which the given MAC address has been learned, the OLT shall use the *aUniMacLearned* (0xD7/0x01-11) attribute (see 14.4.3.2.16). If the sub-attribute *sUniPort* contains the value 0xFF, indicating that the given MAC address has not been learned on any of UNI ports, the OLT shall not perform step 2).

To delete a multicast client from a given multicast group under the MAC-based multicast group control method, the OLT may query the ONU again to find out the instance of the UNI port of a given multicast client. Alternatively, the OLT may retain the association of MAC clients and UNI ports when each new multicast client is added, and the OLT may proceed to modify forwarding rules (i.e., add a new rule and/or delete an old rule) without additional querying of the ONU.

7.4.5.4 Multicast forwarding based on mLLID and IP group address

This subclause defines OLT and ONU multicast forwarding process based on mLLID and IP group address. Other configurations, while not explicitly described here, are also possible. For example, multicast forwarding can be based on mLLID only, on mLLID and L2 DA and/or SA, etc.

7.4.5.4.1 OLT forwarding behavior

The OLT Classifier shall be able to forward frames based on IP destination addresses and also based on a combination of IP destination and source addresses. Multicast frames arriving to the NNI are classified using either IP destination address or a combination of IP destination address and IP source address and are then associated with an mLLID for forwarding across the PON. This is achieved by provisioning an ESP Classifier rule that includes minimally the destination IP multicast address, and may additionally include the source IP address, to determine the appropriate mLLID on which to transport the frames.

In the simplest case, there is a one-to-one association between an IP multicast group and an mLLID. A more complex case exists wherein an mLLID carries frames from more than one multicast group. The set of multicast groups that may be aggregated to use the same mLLID for transport across the PON is determined by operator provisioning.

In the multicast mode based on combined LLID and IP group address, in the downstream direction, the OLT shall apply rules and actions as illustrated in Table 7-40. Each rule (row) in the given table represents a separate multicast group.

Table 7-1—Classifier rules and Modifier actions for downstream ESP in the OLT multicast filtering mode based on mLLID and IP group address [a–f]

Classifier rules	Modifier actions	Description
IF (EXISTS(IPv4_HEADER) AND IPv4_DA == $IP4GA_1$) THEN <m_0,…,x_1>	$[m_0]$: none	These rules are provisioned when multicast filtering is performed based on IPv4 multicast destination address. If a frame's IPv4_DA field matches the provisioned IPv4 group address $IP4GA_n$, the frame is forwarded to the CrossConnect entry (x_n) that forwards the frame further to an output port associated with the proper mLLID.
…		
IF (EXISTS(IPv4_HEADER) AND IPv4_DA == $IP4GA_n$) THEN <m_0,…,x_n>		
IF (EXISTS(IPv4_HEADER) AND IPv4_DA == $IP4GA_1$ AND IPv4_SA == $IP4SA_1$) THEN <m_0,…,x_1>	$[m_0]$: none	These rules are provisioned when multicast filtering is performed based on IPv4 destination and source addresses. If a frame's IPv4_DA field matches the provisioned IPv4 group address $IP4GA_n$ and IPv4_SA field matches the provisioned IPv4 source address $IP4SA_n$, the frame is forwarded to the CrossConnect entry (x_n) that forwards the frame further to an output port associated with the proper mLLID.
…		
IF (EXISTS(IPv4_HEADER) AND IPv4_DA == $IP4GA_n$ AND IPv4_SA == $IP4SA_n$) THEN <m_0,…,x_n>		
IF (EXISTS(IPv6_HEADER) AND IPv6_DA == $IP6GA_1$) THEN <m_0,…,x_1>	$[m_0]$: none	These rules are provisioned when multicast filtering is performed based on IPv6 multicast destination address. If a frame's IPv6_DA field matches the provisioned IPv6 group address $IP6GA_n$, the frame is forwarded to the CrossConnect entry (x_n) that forwards the frame further to an output port associated with the proper mLLID.
…		
IF (EXISTS(IPv6_HEADER) AND IPv6_DA == $IP6GA_n$) THEN <m_0,…,x_n>		
IF (EXISTS(IPv6_HEADER) AND IPv6_DA == $IP6GA_1$ AND IPv6_SA == $IP6SA_1$) THEN <m_0,…,x_1>	$[m_0]$: none	These rules are provisioned when multicast filtering is performed based on IPv6 destination and source addresses. If a frame's IPv6_DA field matches
…		

Classifier rules	Modifier actions	Description
IF (EXISTS(IPv6_HEADER) AND IPv6_DA == $IP6GA_n$ AND IPv6_SA == $IP6SA_n$) THEN <m_0,…,x_n>		the provisioned IPv6 group address $IP6GA_n$ and IPv6_SA field matches the provisioned IPv6 source address $IP6SA_n$, the frame is forwarded to the CrossConnect entry (x_n) that forwards the frame further to an output port associated with the proper mLLID.

[a] $IP4GA_1$–$IP4GA_n$ represent provisioned IPv4 destination group addresses.

[b] $IP4SA_1$–$IP4SA_n$ represent provisioned IPv4 source addresses.

[c] $IP6GA_1$–$IP6GA_n$ represent provisioned IPv6 destination group addresses.

[d] $IP6SA_1$–$IP6SA_n$ represent provisioned IPv6 source addresses.

[e] When both source and destination addresses are used for matching multicast frames, the same values of destination group addresses may be combined with different values of the source addresses, and the same value of the source address may be combined with different values of the destination addresses. A unique combination of source and destination addresses identifies a unique multicast group.

[f] x_1–x_n represent the CrossConnect entry that forwards the frame to an output port associated with the proper mLLID. When the OLT supports 1 Gb/s and 10 Gb/s downstream channels, the x_i entry is provisioned to duplicate frames to two output ports: one associated with 1 Gb/s mLLID and the other associated with 10 Gb/s mLLID. Both mLLIDs may have the same or different numerical values.

7.4.5.4.2 ONU forwarding behavior

At the ONU, multicast sessions may be identified by any of the following combinations of fields:

— IP DA

— IP DA and IP SA

— mLLID value and IP DA

— mLLID value, IP DA, and IP SA

In the multicast mode based on combined LLID and IP group address, in the downstream direction, the ONU shall apply rules and actions as illustrated in Table 7-40. Each rule (row) in the given table represents a separate multicast session.

Table 7-2—Classifier rules and Modifier actions for downstream ESP in the ONU multicast filtering mode based on mLLID and IP group address [a–g]

Classifier rules	Modifier actions	Description
IF (EXISTS(IPv4_HEADER) AND IPv4_DA == $IP4GA_1$) THEN <m_0,…,x_1>	$[m_0]$: none	These rules are provisioned when multicast filtering is performed based on IPv4 multicast destination address. If a frame's IPv4_DA field matches the provisioned IPv4 group address $IP4GA_n$, the frame is forwarded to the CrossConnect entry (x_n) that forwards the frame further to a set of output ports associated with the given IP multicast session.
…		
IF (EXISTS(IPv4_HEADER) AND IPv4_DA == $IP4GA_n$) THEN <m_0,…,x_n>		
IF (EXISTS(IPv4_HEADER) AND IPv4_DA == $IP4GA_1$ AND IPv4_SA == $IP4SA_1$) THEN <m_0,…,x_1>	$[m_0]$: none	These rules are provisioned when multicast filtering is performed based on IPv4 destination and source addresses. If a frame's IPv4_DA field matches the provisioned IPv4 group address $IP4GA_n$ and IPv4_SA field matches the provisioned IPv4 source address $IP4SA_n$, the frame is forwarded to the CrossConnect entry (x_n) that forwards the frame further to a set of output ports associated with the given IP multicast session.
…		
IF (EXISTS(IPv4_HEADER) AND IPv4_DA == $IP4GA_n$ AND IPv4_SA == $IP4SA_n$) THEN <m_0,…,x_n>		
IF (EXISTS(IPv6_HEADER) AND IPv6_DA == $IP6GA_1$) THEN <m_0,…,x_1>	$[m_0]$: none	These rules are provisioned when multicast filtering is performed based on IPv6 multicast destination address. If a frame's IPv6_DA field matches the provisioned IPv6 group address $IP6GA_n$, the frame is forwarded to the CrossConnect entry (x_n) that forwards the frame further to a set of output ports associated with the given IP multicast session.
…		
IF (EXISTS(IPv6_HEADER) AND IPv6_DA == $IP6GA_n$) THEN <m_0,…,x_n>		
IF (EXISTS(IPv6_HEADER) AND IPv6_DA == $IP6GA_1$ AND IPv6_SA == $IP6SA_1$) THEN <m_0,…,x_1>	$[m_0]$: none	These rules are provisioned when multicast filtering is performed based on IPv6 destination and source addresses. If a frame's IPv6_DA field matches
…		

Classifier rules	Modifier actions	Description
IF (EXISTS(IPv6_HEADER) AND IPv6_DA == IP6GA$_n$ AND IPv6_SA == IP6SA$_n$) THEN <m_0,…,x_n>		the provisioned IPv6 group address IP6GA$_n$ and IPv6_SA field matches the provisioned IPv6 source address IP6SA$_n$, the frame is forwarded to the CrossConnect entry (x_n) that forwards the frame further to a set of output ports associated with the given IP multicast session.
IF (LLID_VALUE == MLLID AND EXISTS(IPv4_HEADER) AND IPv4_DA == IP4GA$_1$) THEN <m_0,…,x_1>	[m_0]: none	These rules are provisioned when multicast filtering is performed based on mLLID value and IPv4 multicast destination address. If a frame's LLID_VALUE field matches the provisioned MLLID value and IPv4_DA field matches the provisioned IPv4 group address IP4GA$_n$, the frame is forwarded to the CrossConnect entry (x_n) that forwards the frame further to a set of output ports associated with the given IP multicast session.
…		
IF (LLID_VALUE == MLLID AND EXISTS(IPv4_HEADER) AND IPv4_DA == IP4GA$_n$) THEN <m_0,…,x_n>		
IF (LLID_VALUE == MLLID AND EXISTS(IPv4_HEADER) AND IPv4_DA == IP4GA$_1$ AND IPv4_SA == IP4SA$_1$) THEN <m_0,…,x_1>	[m_0]: none	These rules are provisioned when multicast filtering is performed based on mLLID value and IPv4 destination and source addresses. If a frame's LLID_VALUE field matches the provisioned MLLID value and IPv4_DA field matches the provisioned IPv4 group address IP4GA$_n$ and IPv4_SA field matches the provisioned IPv4 source address IP4SA$_n$, the frame is forwarded to the CrossConnect entry (x_n) that forwards the frame further to a set of output ports associated with the given IP multicast session.
…		
IF (LLID_VALUE == MLLID AND EXISTS(IPv4_HEADER) AND IPv4_DA == IP4GA$_n$ AND IPv4_SA == IP4SA$_n$) THEN <m_0,…,x_n>		
IF (LLID_VALUE == MLLID AND EXISTS(IPv6_HEADER) AND IPv6_DA == IP6GA$_1$) THEN <m_0,…,x_1>	[m_0]: none	These rules are provisioned when multicast filtering is performed based on mLLID value and IPv6 multicast destination address. If a frame's LLID_VALUE field matches the provisioned MLLID value and IPv6_DA field matches the
…		

Classifier rules	Modifier actions	Description
IF (LLID_VALUE == MLLID AND EXISTS(IPv6_HEADER) AND IPv6_DA == IP6GA$_n$) THEN <m_0,…,x_n>		provisioned IPv6 group address IP6GA$_n$, the frame is forwarded to the CrossConnect entry (x_n) that forwards the frame further to a set of output ports associated with the given IP multicast session.
IF (LLID_VALUE == MLLID AND EXISTS(IPv6_HEADER) AND IPv6_DA == IP6GA$_1$ AND IPv6_SA == IP6SA$_1$) THEN <m_0,…,x_1>	[m_0]: none	These rules are provisioned when multicast filtering is performed based on mLLID value and IPv6 destination and source addresses. If a frame's LLID_VALUE field matches the provisioned MLLID value and IPv6_DA field matches the provisioned IPv6 group address IP6GA$_n$ and IPv6_SA field matches the provisioned IPv6 source address IP6SA$_n$, the frame is forwarded to the CrossConnect entry (x_n) that forwards the frame further to a set of output ports associated with the given IP multicast session.
…		
IF (LLID_VALUE == MLLID AND EXISTS(IPv6_HEADER) AND IPv6_DA == IP6GA$_n$ AND IPv6_SA == IP6SA$_n$) THEN <m_0,…,x_n>		

[a] IP4GA$_1$–IP4GA$_n$ represent provisioned IPv4 destination group addresses.

[b] IP4SA$_1$–IP4SA$_n$ represent provisioned IPv4 source addresses.

[c] IP6GA$_1$–IP6GA$_n$ represent provisioned IPv6 destination group addresses.

[d] IP6SA$_1$–IP6SA$_n$ represent provisioned IPv6 source addresses.

[e] MLLID represents provisioned mLLID value

[f] When both source and destination addresses are used for matching multicast frames, the same values of destination group addresses may be combined with different values of the source addresses, and the same value of the source address may be combined with different values of the destination addresses. A unique combination of source and destination addresses identifies a unique multicast group.

[g] x_1–x_n represent the CrossConnect entry that forwards the frame to a set of output port associated with the the given IP multicast session.