White paper
Cisco public

The bridge to possible

MLDP with Flex-Algo in Service Provider Networks

Contents

What will you learn?

This paper discusses the MLDP over Flex-Algo solution, which will allow simple traffic-engineered MLDP LSPs that satisfy a given set of constraints. The feature is being developed as per the draft draft-wijnands-mpls-mldp-multi-topology-01, which addresses Multi-Topology Routing with Flex-Algo to enable service differentiation within an IP network. This feature is available starting from Cisco XR Release 7.5.1. This paper discusses how the solution works, along with configuration details and scalability examples in a typical MVPN topology.

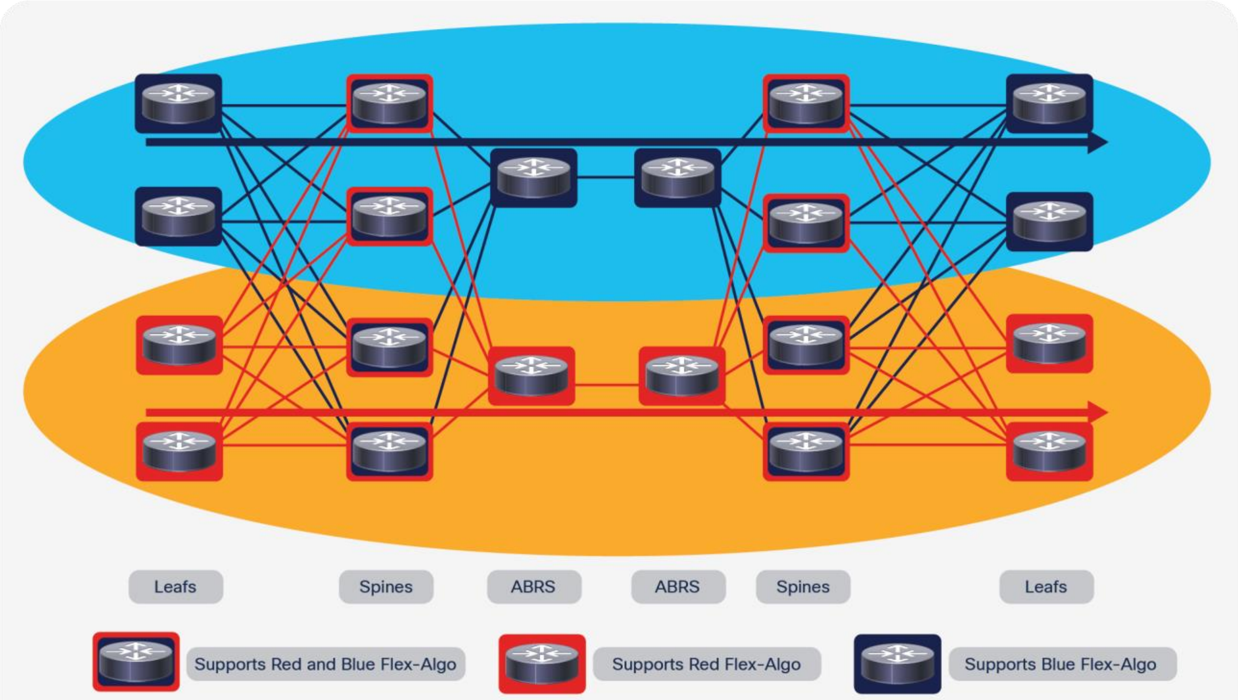
Why is it required?

MLDP P-tunnels are currently only following the shortest path. The MLDP LSP is built from the leaf node toward the root. Hop by hop, the shortest path toward the root address is chosen to send the label mappings and build the tree upstream toward the root.

However, there are cases where some multicast flows may require special treatment. A few of the scenarios include:

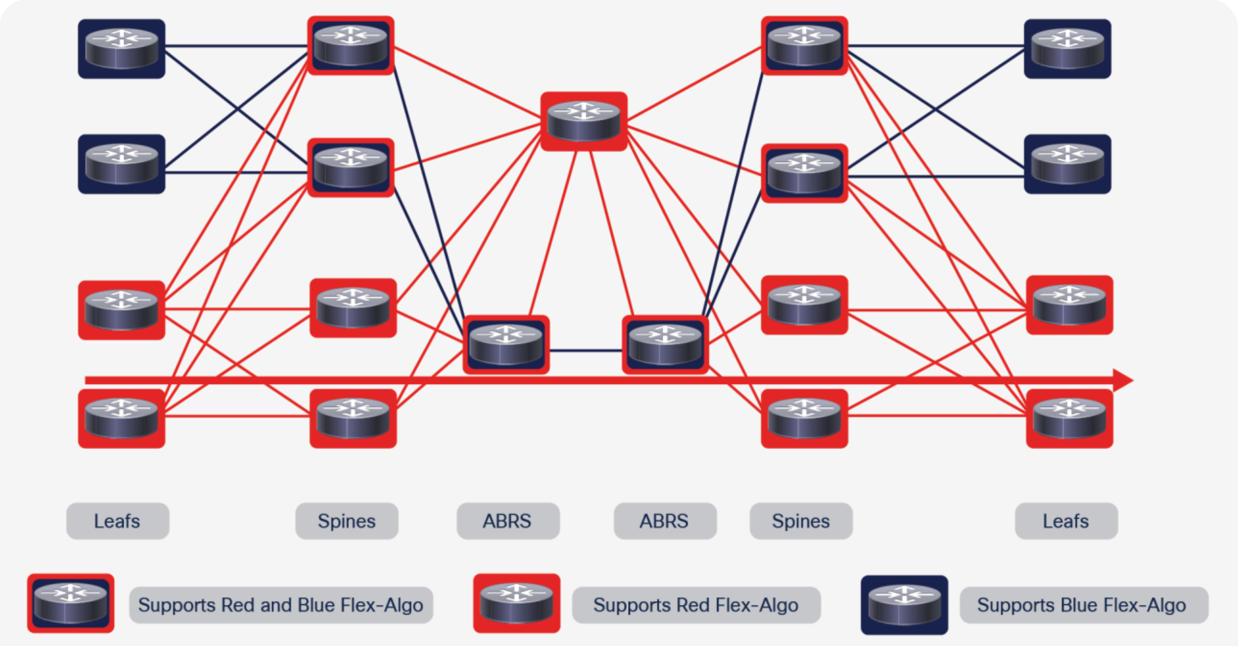
1. Low-latency routing
2. Disjoint paths in case of Live-Live
3. Paths avoiding specific links
4. Scoping/Constraining Multicast Flows to a specific region
5. A very specific routed path

Some customers have stringent high availability requirements for certain applications. For such applications, the customers implement multicast Live-Live, where an application generates two multicast streams for the same feed. Each of the streams must be carried within a separate network topology. Each network topology must be completely disjointed from the other one to prevent the two streams from being impacted at the same time.



Disjoint Paths in case of Live-live

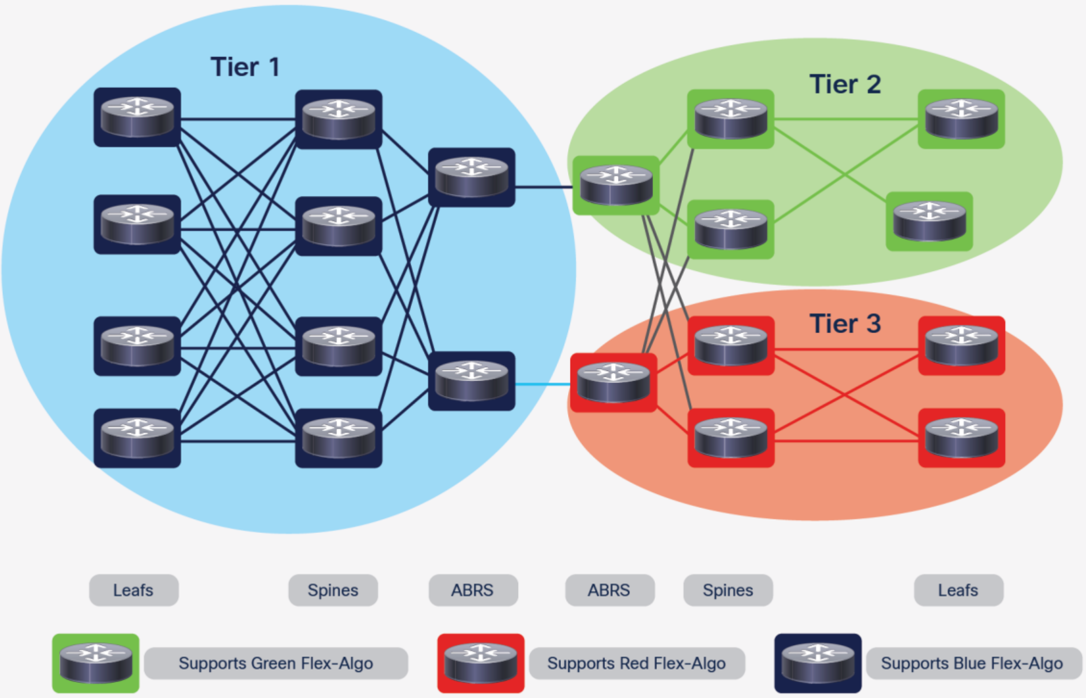
Some customers may want certain MLDP LSPs to take a path other than the usual IGP best path. For example, a certain path may have lower bandwidth but lower latency and hence may be more desirable, though it will not be chosen as the best path by the usual IGP metric. For the MLDP LSP to choose that path, we can use Flex-Algo constraints.



Choosing a non-IGP Path

Similarly, the customer may want the path to avoid certain links or sections of the network. This requirement may be standalone or in addition to the low-latency or Live-Live requirement defined above. Furthermore, a mix of flows is allowed where some flows may take usual non-traffic-engineered paths and some other flows may take Flex-Algo paths.

Finally, the customer may be interested in constraining the scope of certain multicast flows as it may be desirable to limit certain flows by geographical region or a certain administrative boundary. This can be achieved in a very simple and scalable manner using Flex-Algo as compared to using BGP route filters.

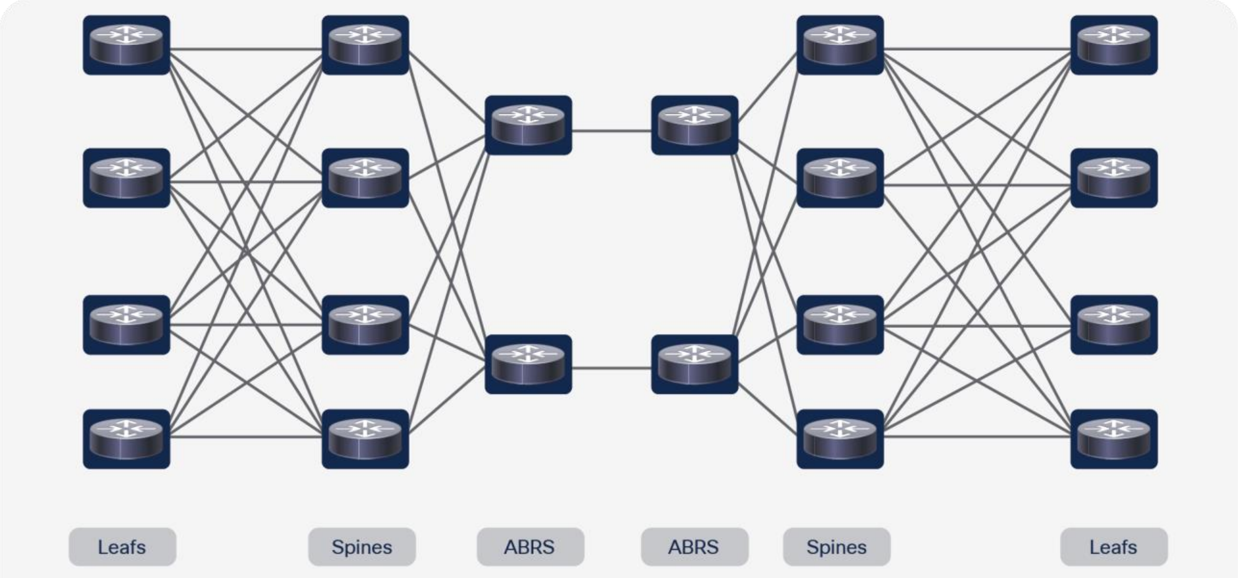


Scoping Flows into regions

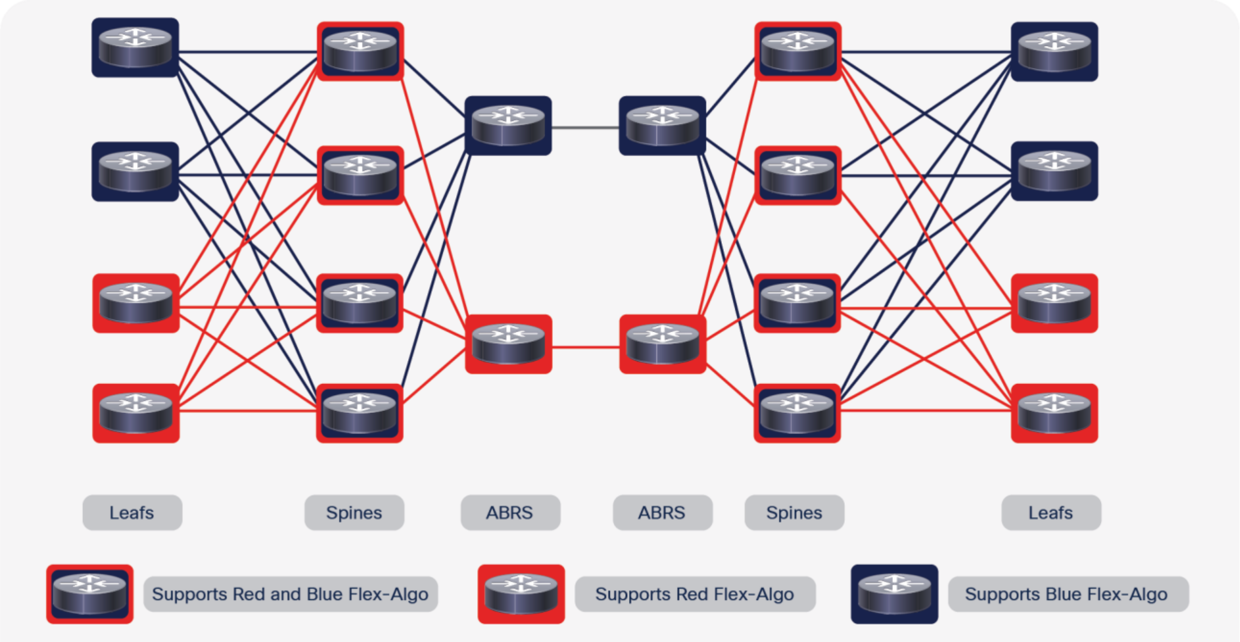
Introduction

What is Flex-Algo

Many possible constraints may be used to compute a path over a network. Some networks are deployed with multiple planes. A simple form of constraint may be to use a particular plane. A more sophisticated form of constraint can include some extended metric, like delay. An even more advanced case could be to restrict the path and avoid links with certain affinities. Combinations of these are also possible. To provide a maximum flexibility, the mapping between the algorithm value and its meaning can be defined by the user. When all the routers in the domain have the common understanding of what the particular algorithm value represents, the computation for such algorithm is consistent and the traffic is not subject to looping. Here, since the meaning of the algorithm is not defined by any standard, but is defined by the user, it is called a Flexible Algorithm.



Network without Flex-Algo



Network with Flex-Algo enabled

MLDP with Flex-Algo

In order to deploy MLDP in a network that supports Flexible Algorithms, MLDP is required to become topology and flexible algorithm aware. New extensions are required for MLDP to support Flexible Algorithms when building Multipoint LSPs so that it can follow a particular algorithm.

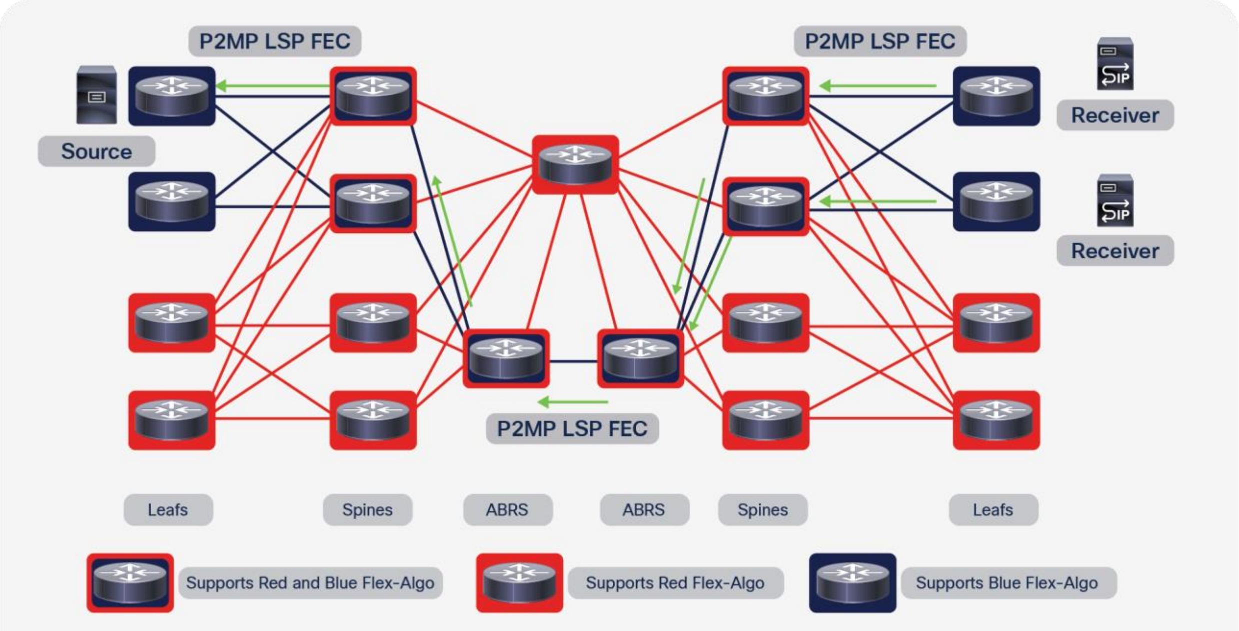
To enable MLDP with Flexible Algorithms, first multi-topology capability must be negotiated with its peers. “MT Multipoint Capability” is a new LDP capability that is to be advertised to its peers by an MLDP speaker to announce its capability to support Multi-Topology Routing. This capability is sent in an initialization message at the session establishment time to its peer.

A P2MP LSP consists of a single root node, zero or more transit nodes, and one or more leaf nodes. Leaf nodes initiate P2MP LSP setup and teardown. Leaf nodes also install forwarding state to deliver the traffic received on a P2MP LSP to wherever it needs to go. Transit nodes install MPLS forwarding state and propagate the P2MP LSP setup (and teardown) toward the root. The root node installs forwarding state to map traffic into the P2MP LSP; the root node determines which traffic should go over the P2MP LSP.

For the setup of a P2MP LSP with MLDP, the P2MP FEC Element is used as a FEC Element in the FEC TLV. The P2MP FEC Element consists of the address of the root of the P2MP LSP and an opaque value. The opaque value consists of one or more MLDP MP opaque value elements. The opaque value is unique within the context of the root node. The combination of (Root Node Address type, Root Node Address, Opaque Value) uniquely identifies a P2MP LSP within the MPLS network.

For enabling P2MP MLDP with Flex-Algo, P2MP FEC elements need to be extended. The Flex-Algo is an entity that is relevant in the context of the root address of the P2MP LSP. The Flex-Algo determines in which sub-topology the root address needs to be resolved. So, the Flex-Algo should be part of the MLDP FEC.

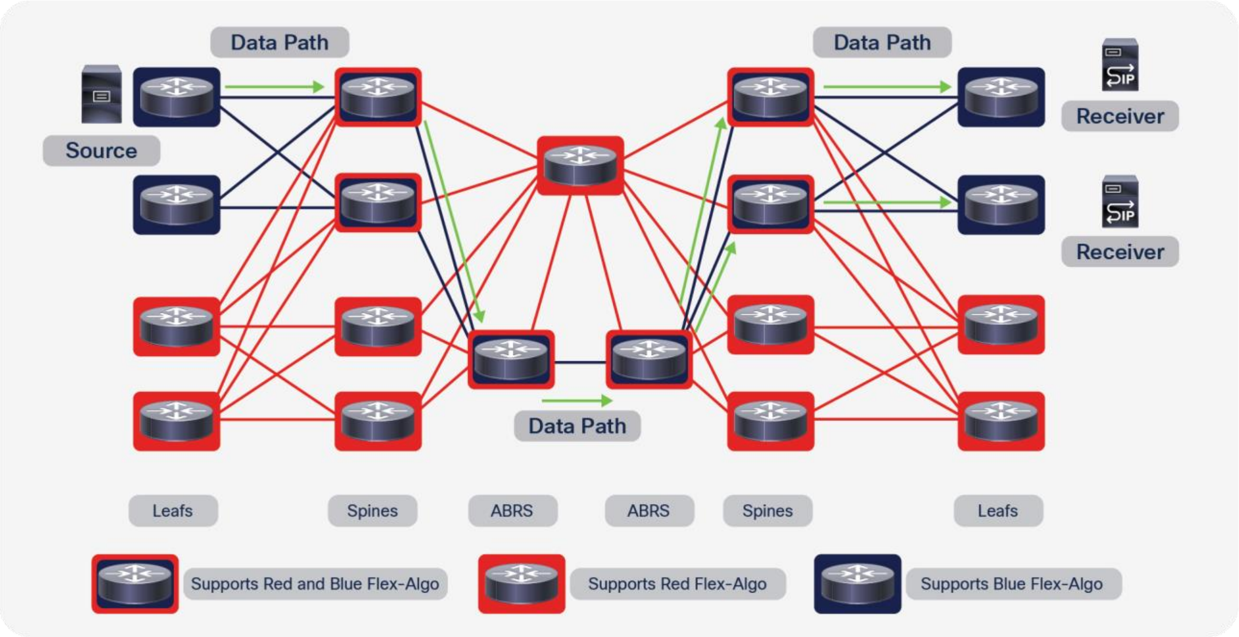
New address families "MT IP" and "MT IPv6" allow the specification of an IP prefix within a flexible algorithm or topology scope.



P2MP LSP built on blue Flex-Algo Sub-topology

The (MT-ID, Flex-Algo) tuple is part of an MLDP FEC so there is no need to support multiple sub-topologies forwarding tables in MLDP. Each MP LSP will be unique due to the tuple being part of the FEC. There is also no need to have specific label forwarding tables per topology, and each MP LSP will have its own unique local label in the table.

The procedures to select the best path to reach the root (upstream LSR) are already defined for P2MP LSPs in RFC 6388. MLDP with Flex-Algo uses the existing procedures to build the Flex-Algo-enabled MP LSPs. The (MT-ID, Flex-Algo) tuple is signaled as part of the FEC; this tuple is used to select the sub-topology that must be used to find the best path to the root address. Best path selection is done with the existing procedures defined for MP LSPs.



The procedures to select a downstream forwarding interface are also already defined in RFC 6388. In these procedures, any interface leading to the downstream LDP neighbor can be considered as a candidate forwarding interface. With Flex-Algo in the picture, this doesn't hold true. There may be multiple downstream links to the LDP neighbor participating in different Flex-Algos. An interface must only be selected if it is part of the same sub-topology that was signaled in the MLDP FEC element. Apart from this restriction, the rest of the existing procedures are applicable for downstream selection.

TLVs

In order to build the MP LSPs within a sub-topology, the feature introduces one new TLV and modifies one existing TLV. For the centralized mode of operation, an additional two TLVs are defined (MT Multipoint Capability TLV).

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|U|F|  MT Multipoint Cap.(IANA) |            Length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|S| Reserved    |
+-+-+-+-+-+-+-+-+
                      MT Multipoint Capability TLV Format
```

Where:

- U- and F-bits: MUST be 1 and 0, respectively, as per LDP Capabilities in [RFC5561].
- MT Multipoint Capability: TLV type (IANA assigned).
- Length: The length (in octets) of TLV. The value of this field MUST be 1 as there is no Capability-specific data [RFC5561] that follows in the TLV.
- S-bit: Set to 1 to announce and 0 to withdraw the capability (as per [RFC5561].

By using extended MT IP Address Family, the resultant MT MP FEC element is to be encoded as follows:

```
0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| MP FEC type   |  AF (MT IP/ MT IPv6)          |    AF Length  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Root Node Address                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Reserved   |      IPA      |        MT-ID                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Opaque Length              |       Opaque Value            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+                               +
~                                                               ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

IP MT-Scoped MP FEC Element Format

The applicable LDP FECs for MT MLDP include:

- MP FEC Elements:
  - P2MP
  - MP2MP-up
  - MP2MP-down
- Typed Wildcard FEC Element

Only P2MP FEC supports Flex-Algo enablement.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     IPv4 Address                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Reserved   |      IPA      |        MT-ID                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Modified MT IP Address Families Data Format

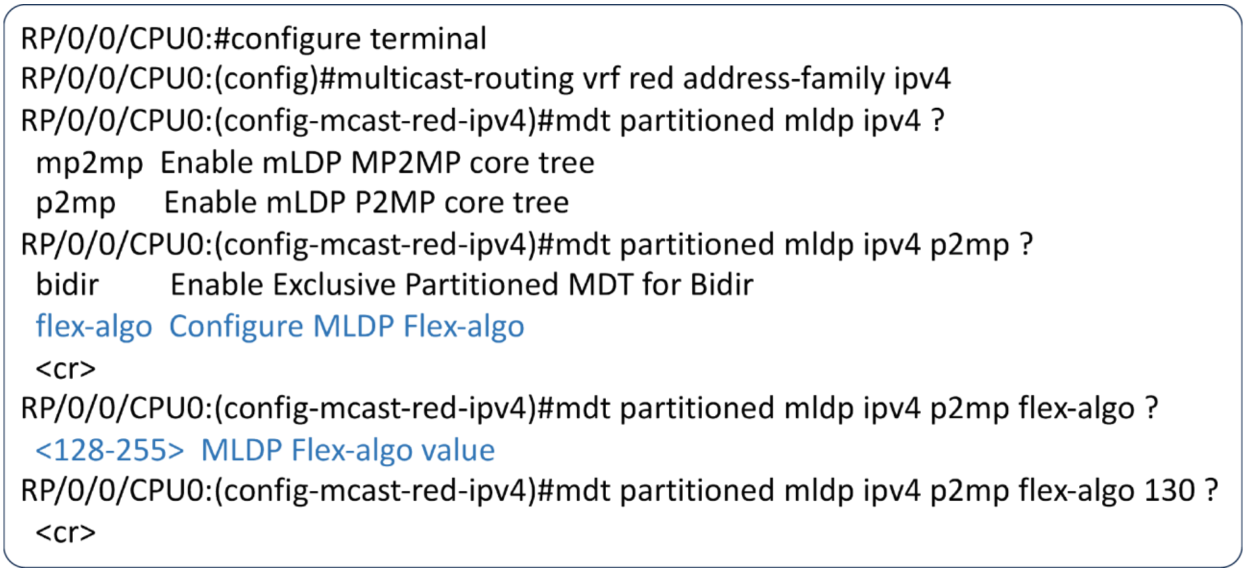
Where:

- IPv4 Address: An IP address corresponding to the " MT IP" address family.
- IPA: The IGP Algorithm; values are from the IGP Algorithm registry.
- Reserved: This 8-bit field MUST be zero. If a message is received that includes an MT address family where the 8-bit Reserved value is not zero, the message must be discarded.

Configuration

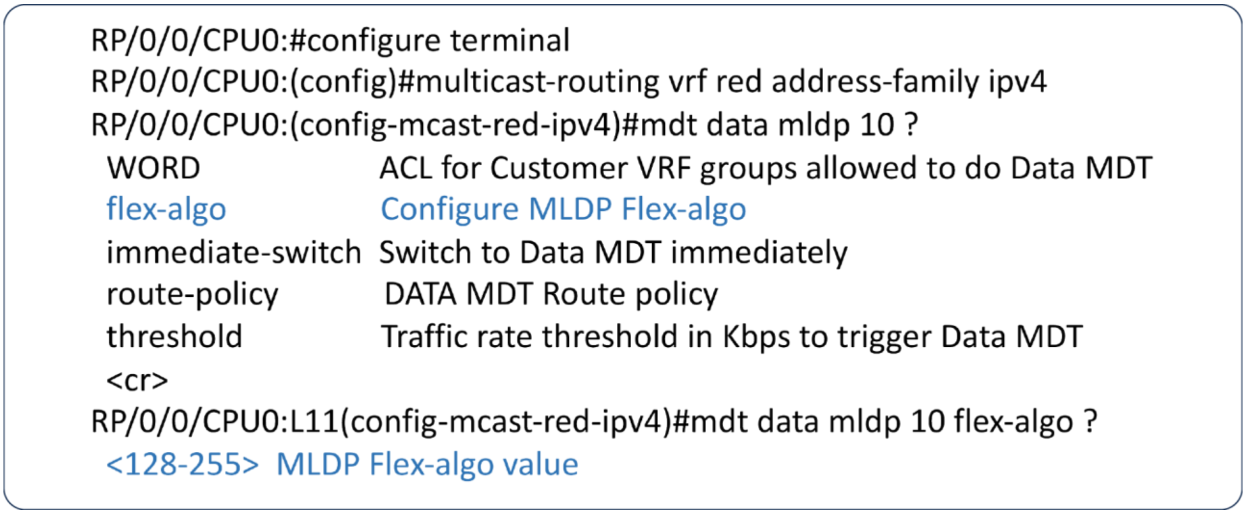
Note: The configuration commands and examples are based on the initial implementation of MLDP Flex-Algo in XR Release 7.5.1.

We can enable MLDP Flex-Algo in the partitioned MDT as follows:



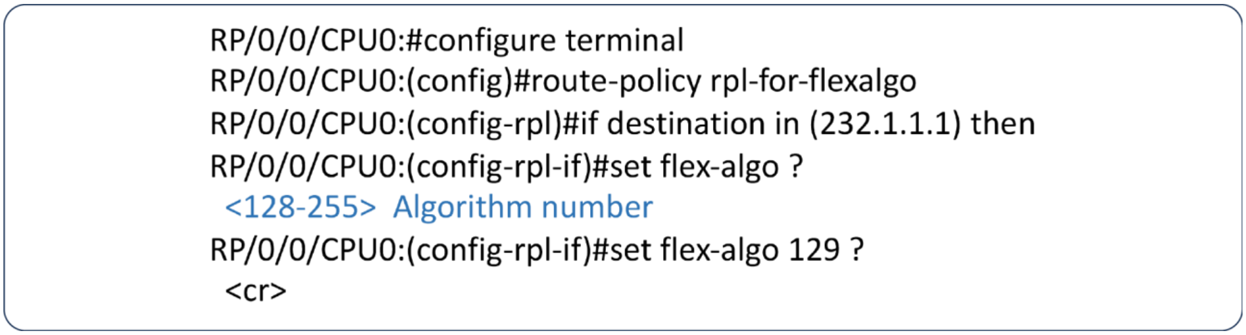
```
RP/0/0/CPU0:#configure terminal
RP/0/0/CPU0:(config)#multicast-routing vrf red address-family ipv4
RP/0/0/CPU0:(config-mcast-red-ipv4)#mdt partitioned mldp ipv4 ?
 mp2mp  Enable mLDP MP2MP core tree
 p2mp   Enable mLDP P2MP core tree
RP/0/0/CPU0:(config-mcast-red-ipv4)#mdt partitioned mldp ipv4 p2mp ?
 bidir      Enable Exclusive Partitioned MDT for Bidir
 flex-algo  Configure MLDP Flex-algo
 <cr>
RP/0/0/CPU0:(config-mcast-red-ipv4)#mdt partitioned mldp ipv4 p2mp flex-algo ?
 <128-255>  MLDP Flex-algo value
RP/0/0/CPU0:(config-mcast-red-ipv4)#mdt partitioned mldp ipv4 p2mp flex-algo 130 ?
 <cr>
```

This will cause the partitioned MDT to be created in the sub-topology defined by the configured Flex-Algo. Similarly, we can enable MLDP Flex-Algo in the Data MDT as follows:



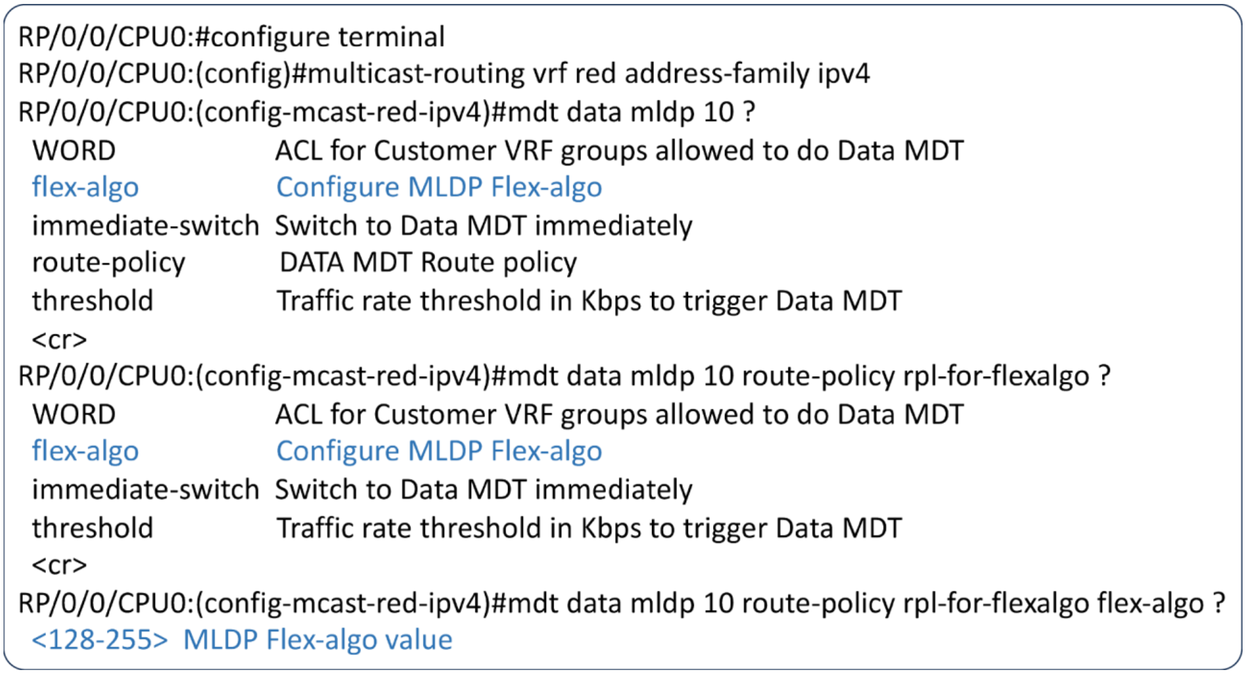
```
RP/0/0/CPU0:#configure terminal
RP/0/0/CPU0:(config)#multicast-routing vrf red address-family ipv4
RP/0/0/CPU0:(config-mcast-red-ipv4)#mdt data mldp 10 ?
 WORD              ACL for Customer VRF groups allowed to do Data MDT
 flex-algo         Configure MLDP Flex-algo
 immediate-switch  Switch to Data MDT immediately
 route-policy       DATA MDT Route policy
 threshold         Traffic rate threshold in Kbps to trigger Data MDT
 <cr>
RP/0/0/CPU0:L11(config-mcast-red-ipv4)#mdt data mldp 10 flex-algo ?
 <128-255>  MLDP Flex-algo value
```

This will cause the Data MDT to be created in the sub-topology defined by the configured Flex-Algo. Furthermore, we can also create a policy for MLDP Flex-Algo as follows:



```
RP/0/0/CPU0:#configure terminal
RP/0/0/CPU0:(config)#route-policy rpl-for-flexalgo
RP/0/0/CPU0:(config-rpl)#if destination in (232.1.1.1) then
RP/0/0/CPU0:(config-rpl-if)#set flex-algo ?
 <128-255>  Algorithm number
RP/0/0/CPU0:(config-rpl-if)#set flex-algo 129 ?
 <cr>
```

We can then associate the policy to the Data MDT as follows:

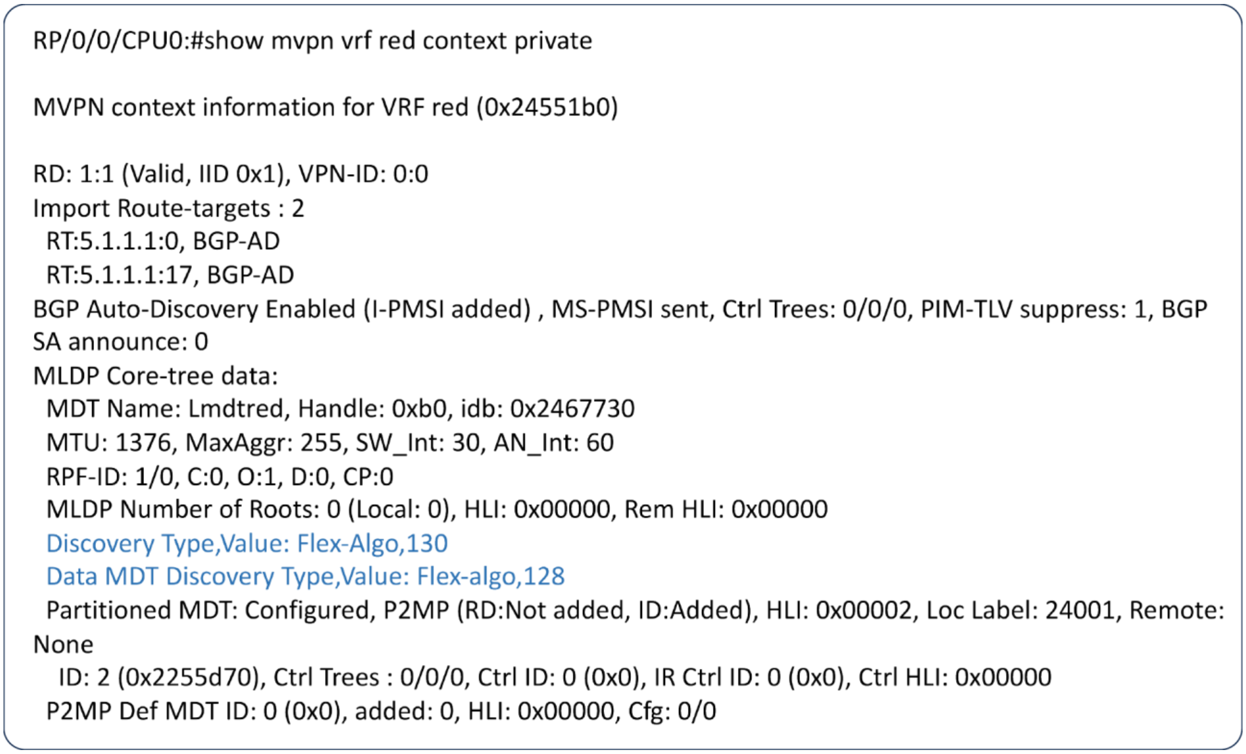


```
RP/0/0/CPU0:#configure terminal
RP/0/0/CPU0:(config)#multicast-routing vrf red address-family ipv4
RP/0/0/CPU0:(config-mcast-red-ipv4)#mdt data mldp 10 ?
 WORD              ACL for Customer VRF groups allowed to do Data MDT
 flex-algo         Configure MLDP Flex-algo
 immediate-switch  Switch to Data MDT immediately
 route-policy      DATA MDT Route policy
 threshold         Traffic rate threshold in Kbps to trigger Data MDT
 <cr>
RP/0/0/CPU0:(config-mcast-red-ipv4)#mdt data mldp 10 route-policy rpl-for-flexalgo ?
 WORD              ACL for Customer VRF groups allowed to do Data MDT
 flex-algo         Configure MLDP Flex-algo
 immediate-switch  Switch to Data MDT immediately
 threshold         Traffic rate threshold in Kbps to trigger Data MDT
 <cr>
RP/0/0/CPU0:(config-mcast-red-ipv4)#mdt data mldp 10 route-policy rpl-for-flexalgo flex-algo ?
 <128-255>  MLDP Flex-algo value
```

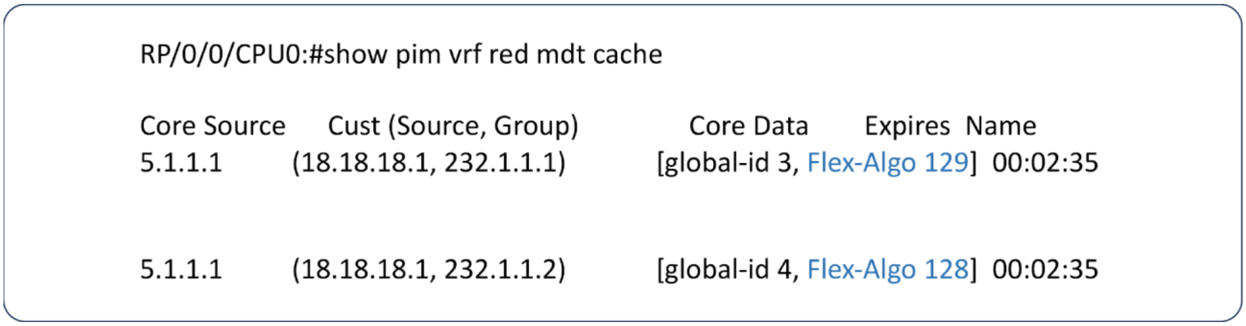
In the event all the three options for specifying a Flex-Algo are configured, one of the following Flex-Algo (in the order of precedence) will be applied:

1. If a stream matches an RPL policy, the stream will be associated with the Flex-Algo attached to that policy.
2. If there is a Flex-Algo configured under the data-mdt command, the stream will be associated with the Flex-Algo specified there.
3. If there is a Flex-Algo configured under the partitioned-mdt command, the stream will be associated with the Flex-Algo specified there.
4. The stream will be associated with the default topology with no Flex-Algo.

To verify the state, we can display the Flex-Algo associated to the partitioned MDT as well as the Flex-Algo associated to the various S, G streams in PIM as follows:

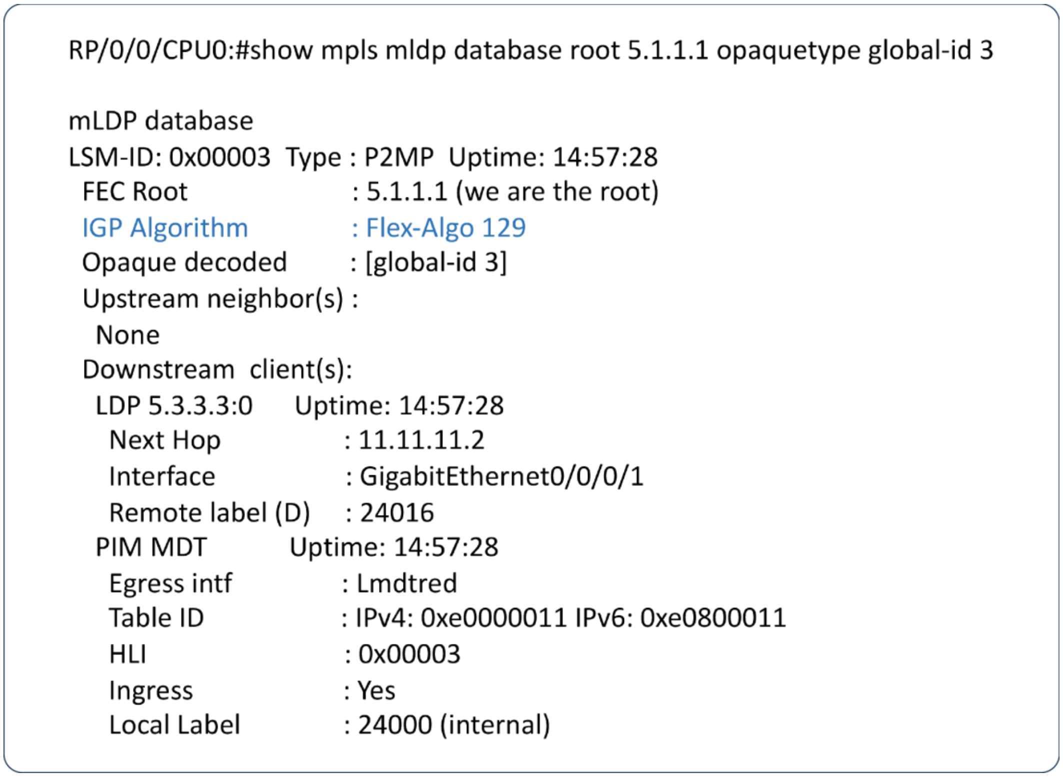


```
RP/0/0/CPU0:#show mvpn vrf red context private

MVPN context information for VRF red (0x24551b0)

RD: 1:1 (Valid, IID 0x1), VPN-ID: 0:0
Import Route-targets : 2
  RT:5.1.1.1:0, BGP-AD
  RT:5.1.1.1:17, BGP-AD
BGP Auto-Discovery Enabled (I-PMSI added) , MS-PMSI sent, Ctrl Trees: 0/0/0, PIM-TLV suppress: 1, BGP
SA announce: 0
MLDP Core-tree data:
  MDT Name: Lmdtred, Handle: 0xb0, idb: 0x2467730
  MTU: 1376, MaxAggr: 255, SW_Int: 30, AN_Int: 60
  RPF-ID: 1/0, C:0, O:1, D:0, CP:0
  MLDP Number of Roots: 0 (Local: 0), HLI: 0x00000, Rem HLI: 0x00000
  Discovery Type,Value: Flex-Algo,130
  Data MDT Discovery Type,Value: Flex-algo,128
  Partitioned MDT: Configured, P2MP (RD:Not added, ID:Added), HLI: 0x00002, Loc Label: 24001, Remote:
None
    ID: 2 (0x2255d70), Ctrl Trees : 0/0/0, Ctrl ID: 0 (0x0), IR Ctrl ID: 0 (0x0), Ctrl HLI: 0x00000
  P2MP Def MDT ID: 0 (0x0), added: 0, HLI: 0x00000, Cfg: 0/0
```

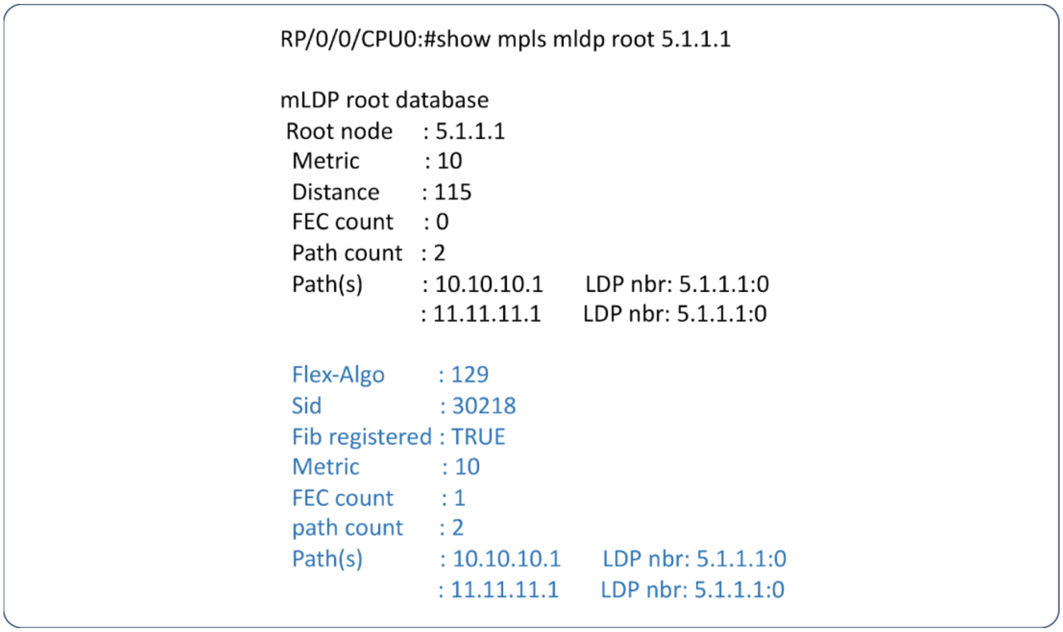


```
RP/0/0/CPU0:#show pim vrf red mdt cache

Core Source    Cust (Source, Group)          Core Data       Expires  Name
5.1.1.1        (18.18.18.1, 232.1.1.1)       [global-id 3, Flex-Algo 129]  00:02:35


5.1.1.1        (18.18.18.1, 232.1.1.2)       [global-id 4, Flex-Algo 128]  00:02:35
```

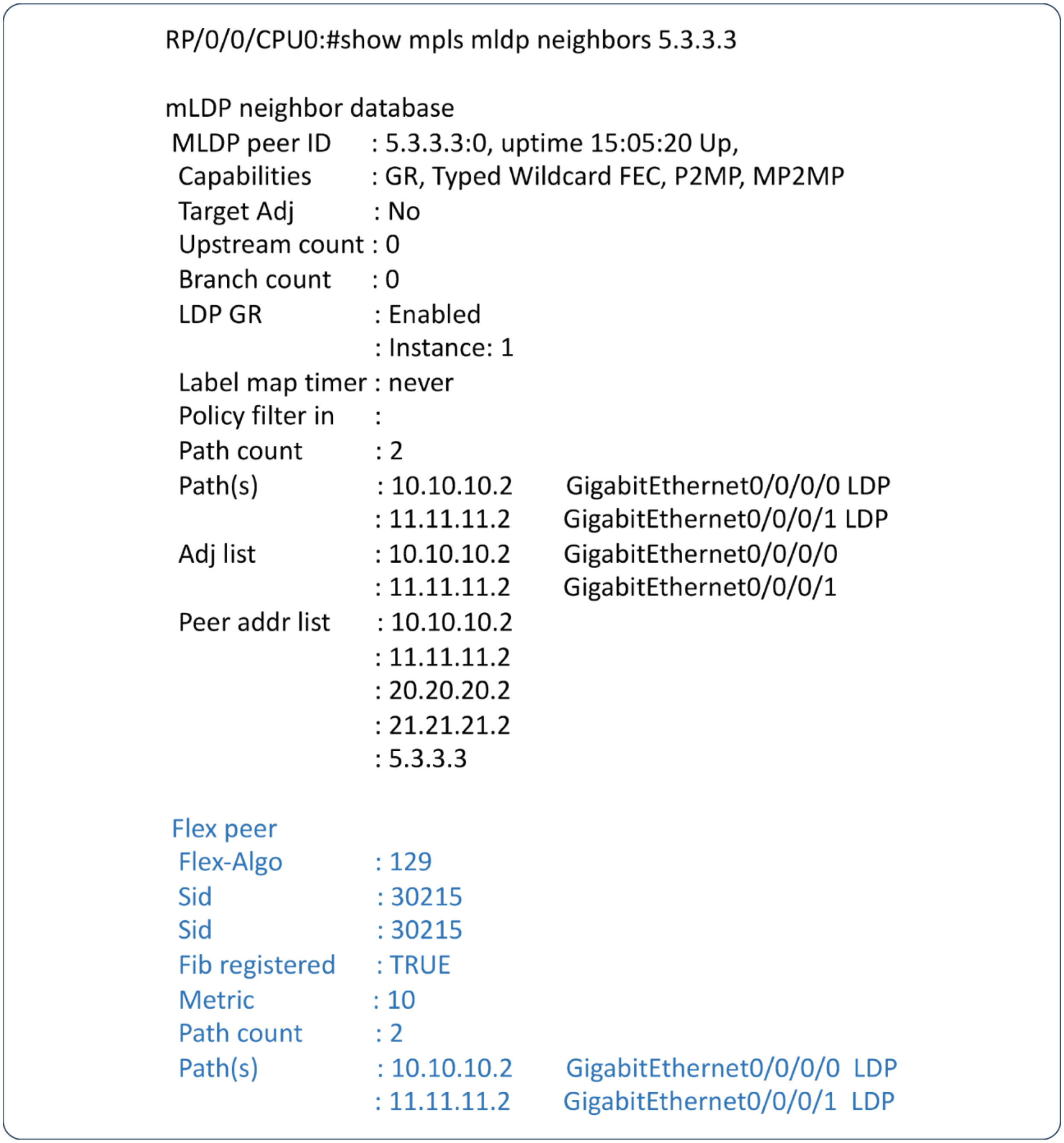
Following are the MLDP show commands to check the Flex-Algo state on a per-FEC basis. The show output also displays the SID label associated with each Flex-Algo and the paths that are learnt for a given Flex-Algo topology.



```
RP/0/0/CPU0:#show mpls mldp database root 5.1.1.1 opaquetype global-id 3

mLDP database
LSM-ID: 0x00003  Type : P2MP  Uptime: 14:57:28
 FEC Root                  : 5.1.1.1 (we are the root)
 IGP Algorithm             : Flex-Algo 129
 Opaque decoded            : [global-id 3]
 Upstream neighbor(s) :
  None
 Downstream  client(s):
  LDP 5.3.3.3:0      Uptime: 14:57:28
   Next Hop                : 11.11.11.2
   Interface               : GigabitEthernet0/0/0/1
   Remote label (D)        : 24016
  PIM MDT           Uptime: 14:57:28
   Egress intf             : Lmdtred
   Table ID                : IPv4: 0xe0000011 IPv6: 0xe0800011
   HLI                     : 0x00003
   Ingress                 : Yes
   Local Label             : 24000 (internal)
```



```
RP/0/0/CPU0:#show mpls mldp root 5.1.1.1

mLDP root database
 Root node    : 5.1.1.1
  Metric      : 10
  Distance    : 115
  FEC count   : 0
  Path count  : 2
  Path(s)     : 10.10.10.1     LDP nbr: 5.1.1.1:0
              : 11.11.11.1     LDP nbr: 5.1.1.1:0

  Flex-Algo        : 129
  Sid              : 30218
  Fib registered : TRUE
  Metric           : 10
  FEC count        : 1
  path count       : 2
  Path(s)          : 10.10.10.1     LDP nbr: 5.1.1.1:0
                   : 11.11.11.1     LDP nbr: 5.1.1.1:0
```



```
RP/0/0/CPU0:#show mpls mldp neighbors 5.3.3.3

mLDP neighbor database
 MLDP peer ID      : 5.3.3.3:0, uptime 15:05:20 Up,
  Capabilities     : GR, Typed Wildcard FEC, P2MP, MP2MP
  Target Adj       : No
  Upstream count : 0
  Branch count     : 0
  LDP GR           : Enabled
                   : Instance: 1
  Label map timer : never
  Policy filter in   :
  Path count       : 2
  Path(s)          : 10.10.10.2       GigabitEthernet0/0/0/0 LDP
                   : 11.11.11.2       GigabitEthernet0/0/0/1 LDP
  Adj list         : 10.10.10.2       GigabitEthernet0/0/0/0
                   : 11.11.11.2       GigabitEthernet0/0/0/1
  Peer addr list   : 10.10.10.2
                   : 11.11.11.2
                   : 20.20.20.2
                   : 21.21.21.2
                   : 5.3.3.3

 Flex peer
  Flex-Algo        : 129
  Sid              : 30215
  Sid              : 30215
  Fib registered   : TRUE
  Metric           : 10
  Path count       : 2
  Path(s)          : 10.10.10.2       GigabitEthernet0/0/0/0  LDP
                   : 11.11.11.2       GigabitEthernet0/0/0/1  LDP
```

Interoperability

There may be devices that have Flex-Algo enabled for unicast reachability but do not support it in MLDP. Devices that do not negotiate “MT Multipoint Capability” won’t be participating in Flex-Algo-enabled LSPs. These devices will continue to participate in building MP LSPs without Flex-Algo as usual.

For upstream or downstream LSR selection, devices not supporting multi-topology shouldn’t be chosen as LSR for Flex-Algo-enabled MP LSPs.

Caveats and limitations

The initial implementation having the following restrictions:

- Default MDT with Flex-Algo is not supported (coming in 7.5.2). Only Partitioned and Data MDTs support Flex-Algo enablement.
- FRR is not supported on MP LSPs enabled with Flex-Algo (coming in 7.5.2).
- PIM as Customer multicast signaling is not supported with Flex-Algo.
- MVPN Inter-AS with Flex-Algo is not supported.
- MVPN Extranet with Flex-Algo is not supported.
- Carrier supporting carrier (CSC) with Flex-Algo is not supported.
- Change of Flex-Algo in multicast configuration is not supported. First, the old Flex-Algo configuration needs to be removed, and then configuration with updated Flex-Algo should be applied.

Summary

Flexible Algorithm allows operators to customize IGP shortest path computation according to their own needs. Flexible Algorithm provides a traffic-engineered path automatically computed by the IGP to any destination reachable by the IGP. MLDP uses Flexible Algorithm to build traffic-engineered MP LSPs. This is used for use cases like low-latency routing, Live-Live disjoint paths, or constraining Multicast Flows to a specific region.

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV Amsterdam,
The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at **https://www.cisco.com/go/offices**.

Printed in USA C11-2785708-00 03/22