

# MPLS Data Plane Encapsulation for In-situ OAM Data

*draft-gandhi-mpls-ioam-sr-05*

*Rakesh Gandhi - Cisco Systems ([rgandhi@cisco.com](mailto:rgandhi@cisco.com)) - Presenter*

*Zafar Ali - Cisco Systems ([zali@cisco.com](mailto:zali@cisco.com))*

*Clarence Filsfils - Cisco Systems ([cfilsfil@cisco.com](mailto:cfilsfil@cisco.com))*

*Frank Brockners - Cisco Systems ([fbrockne@cisco.com](mailto:fbrockne@cisco.com))*

*Bin Wen - Comcast ([Bin\\_Wen@cable.comcast.com](mailto:Bin_Wen@cable.comcast.com))*

*Voitek Kozak - Comcast ([Voitek\\_Kozak@comcast.com](mailto:Voitek_Kozak@comcast.com))*

# Agenda

- Requirements and Scope
- Summary
- Next Steps

# Requirements and Scope

## Requirements:

- Transport In-situ OAM (IOAM) data fields with MPLS Encapsulation

## Scope:

- Using IOAM data fields defined in:
  - *draft-ietf-ippm-ioam-data*
  - *draft-ietf-ippm-ioam-direct-export*
  - *draft-ietf-ippm-ioam-flags*
- Edge-to-edge (E2E) IOAM
- Hop-by-hop (HbH) IOAM (that includes E2E)

# MPLS Extensions

# IOAM G-ACh for IOAM Data Fields

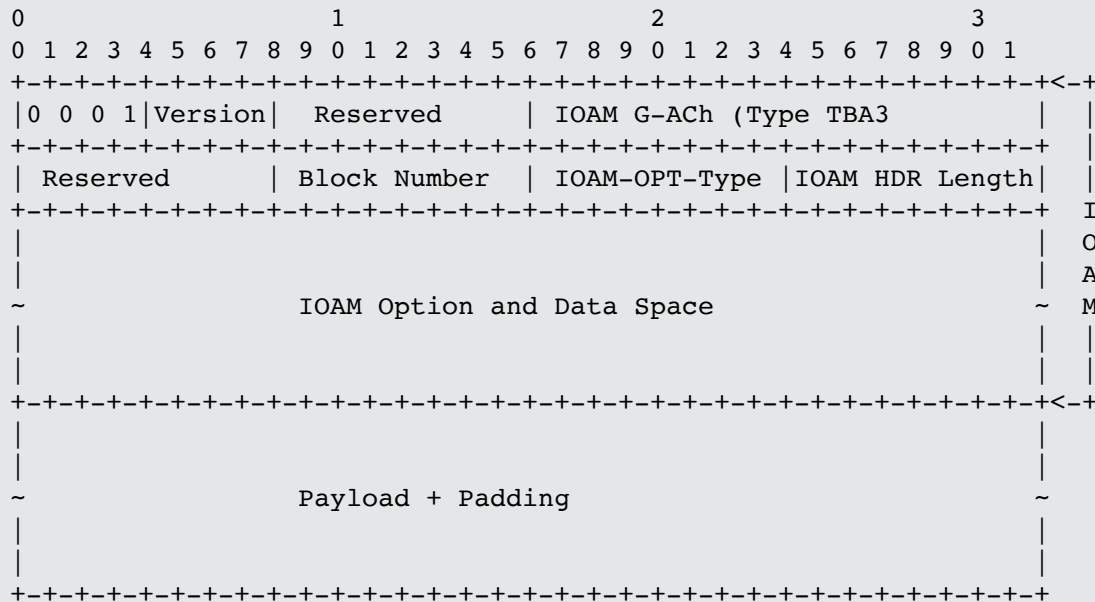


Figure: IOAM G-ACh for IOAM Data Fields

# IOAM G-ACh Header

- New Generic Associated Channel (G-ACh) Type (value **TBA3**) defined for IOAM
- Protocol value *0001b* allows to avoid incorrect IP header based hashing over ECMP paths
- Block Number can be used to:
  - Aggregate IOAM data collected in data plane, e.g. compute measurement metrics for each block of a flow
  - Correlate IOAM data from different nodes

<https://www.iana.org/assignments/g-ach-parameters/g-ach-parameters.xhtml#mpls-g-ach-types>

# IOAM Indicator Labels

- “IOAM Indicator Label” is used to indicate the presence of the IOAM data fields after EOS in the MPLS Encapsulation.
- Separate Indicator Labels are defined for E2E IOAM (for edge nodes) and HbH IOAM (*for edge and intermediate nodes*).
  - The E2E IOAM Label allows to bypass IOAM processing on intermediate nodes in case of E2E IOAM.
- In case of E2E IOAM, the IOAM Option-Type(s) in the data packets are processed on edge nodes only. The intermediate nodes ignore the IOAM Option-Type(s) carried by the data packets. **Hence, only E2E Option-Type is carried in the IOAM data field.**
- In case of HbH IOAM, the IOAM Option-Type(s) in the data packets are processed on intermediate and edge nodes. **Hence, both HbH and E2E Option-Types can be carried in the IOAM data field(s).**

# E2E IOAM



# MPLS Encapsulation with E2E IOAM Data Fields

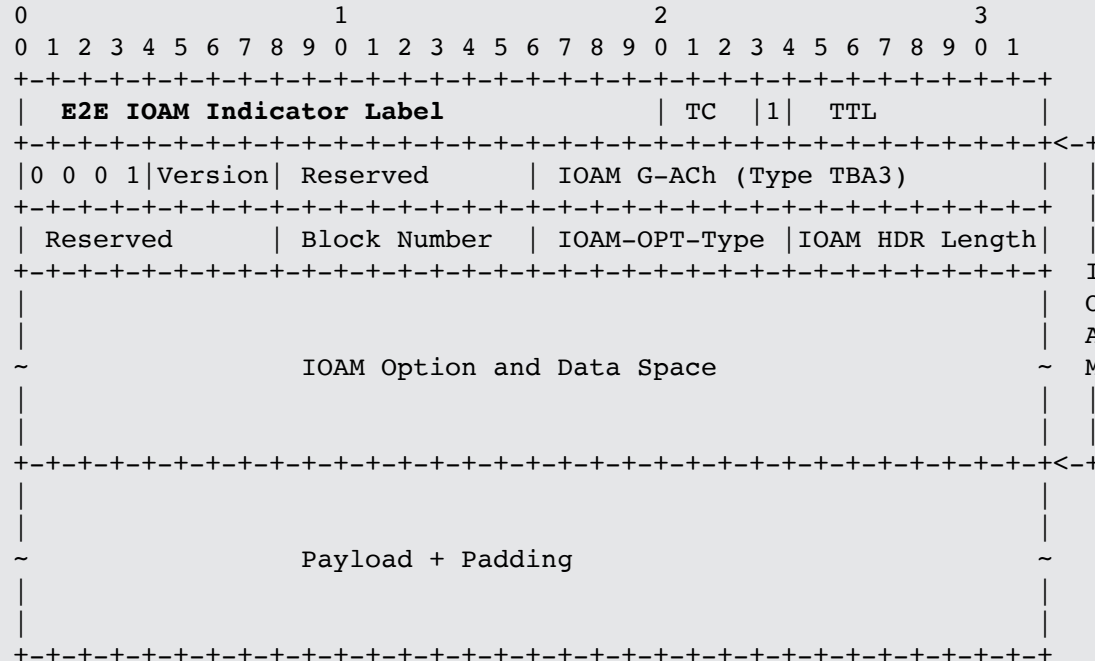


Figure: MPLS Encapsulation with E2E IOAM Data Fields

# E2E IOAM Indicator Label Allocation Methods

1. Extension Label (15) and Label assigned by IANA with value **TBA1**
  - From Extended Special Purpose Labels (eSPL) range
2. Global Label allocated by a controller
  - The controller provisions the label on encapsulating and decapsulating nodes
3. The IOAM Label allocated by the decapsulating node
  - Signaling/advertisement extensions needed to convey the label to all encapsulating nodes (out of scope)

# E2E IOAM Indicator Label - Comparisons

	Method	Extra Label Stack Size (Note 2)	Location on Stack
1	eSPL Labels	+2 (Note 1)	Bottom
2	Global Label	+1	Bottom
3	Signal/Advertise Label	+1 (compared to PHP)	Bottom

1. This is true for any mechanism that we are defining using eSPL
  - SFC: <https://tools.ietf.org/html/rfc8595>
  - E2E: draft-ietf-mpls-inband-pm-encapsulation
2. IOAM data packets may require Entropy label for ECMP to work around hashing issue due to ACH for IP packets

# E2E IOAM Procedure

1. E2E IOAM includes IOAM processing on encapsulating and decapsulating nodes. **The only E2E Option-Type is carried in the IOAM data field.**
2. The encapsulating node inserts an E2E Indicator Label and one or more IOAM data field(s) in the MPLS header.
3. The intermediate (intermediate) nodes do not process IOAM data.
4. The decapsulating node “punts the timestamped copy” of the data packet including IOAM data field(s).
  - a. The decapsulating node processes IOAM data field(s) from the punted packet.
5. The decapsulating node also pops the IOAM Indicator Label and the IOAM data field(s) from the MPLS encapsulation.
  - a. The decapsulating node forwards the data packet downstream.

# HbH IOAM

# MPLS Encapsulation with HbH IOAM Data Fields

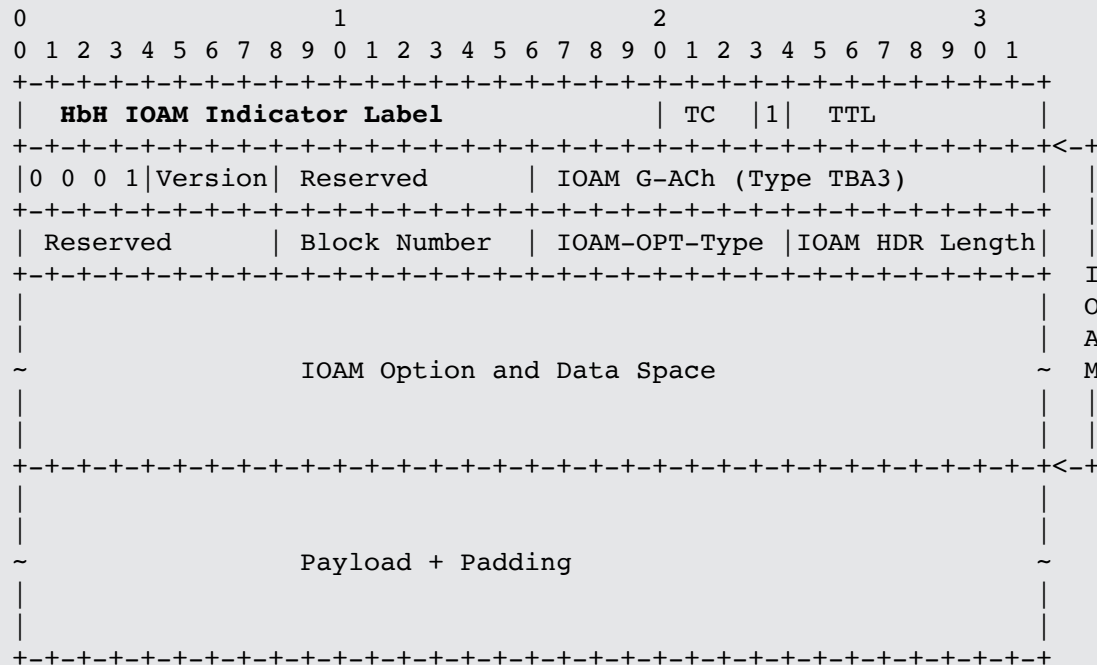


Figure: MPLS Encapsulation with HbH IOAM Data Fields

# HbH IOAM Indicator Label Allocation Methods

1. Extension Label (15) and Label assigned by IANA with value [TBA2](#)
  - From Extended Special Purpose Labels (eSPL) range
2. Global Label allocated by a controller
  - The controller provisions the label on encapsulating, intermediate and decapsulating nodes
3. The IOAM Label allocated by the intermediate and decapsulating nodes
  - Signaling/advertisement extensions needed to convey the label to all encapsulating nodes (out of scope)

# HbH IOAM Indicator Label - Comparisons

	Method	Extra Label Stack Size (Note 4)	Location on Stack	Scan Label Stack (Notes 3)	Different FIB Entry for Local Label
1	eSPL Labels	+2	Bottom (Note 1)	Yes (Note 2)	No
2	Global Label	+1	Bottom	Yes (Note 2)	No
3	Signal/Advertise Label (like SFL)	+0	Incoming Packet with Top Label	No	Yes

1. eSPL at top of the label stack breaks MPLS forwarding in heterogenous network environment with and without IOAM capable nodes
2. Entropy Label similarly also requires intermediate nodes to scan label stack, however, entropy label processing is optional whereas IOAM processing is not optional
3. A intermediate node may have a limit on how many labels it can scan. However, with any indicator scheme, the node will have to look past EOS into the packet to find the IOAM data that needs to be processed
4. IOAM data packets may require Entropy label for ECMP to work around hashing issue due to ACH for IP packets



# HbH IOAM Procedure

1. HbH IOAM includes IOAM processing on encapsulating, intermediate and decapsulating nodes. **Both HbH and E2E Option-Types can be carried in the IOAM data field(s).**
2. The encapsulating node inserts a HbH Indicator Label and one or more IOAM data field(s) in the MPLS encapsulation.
3. The intermediate (intermediate) nodes process HbH IOAM data field(s) and forward the data packet including updated IOAM data field(s).
  - a. The intermediate (intermediate) nodes may punt the timestamped copy of the data packet for further IOAM processing.
4. The decapsulating node "punts the timestamped copy" of the data packet including IOAM data field(s).
  - a. The decapsulating node processes IOAM data field(s) from the punted packet.
5. The decapsulating node also pops the IOAM Indicator Label and the IOAM data field(s) from the MPLS encapsulation.
  - a. The decapsulating node forwards the data packet downstream.

# Next Steps

- Welcome your comments and suggestions
- Requesting MPLS WG adoption

# Thank you

# Example MPLS Encapsulations for IOAM Data Fields

## Example SR-MPLS Encapsulation with IOAM Data Fields

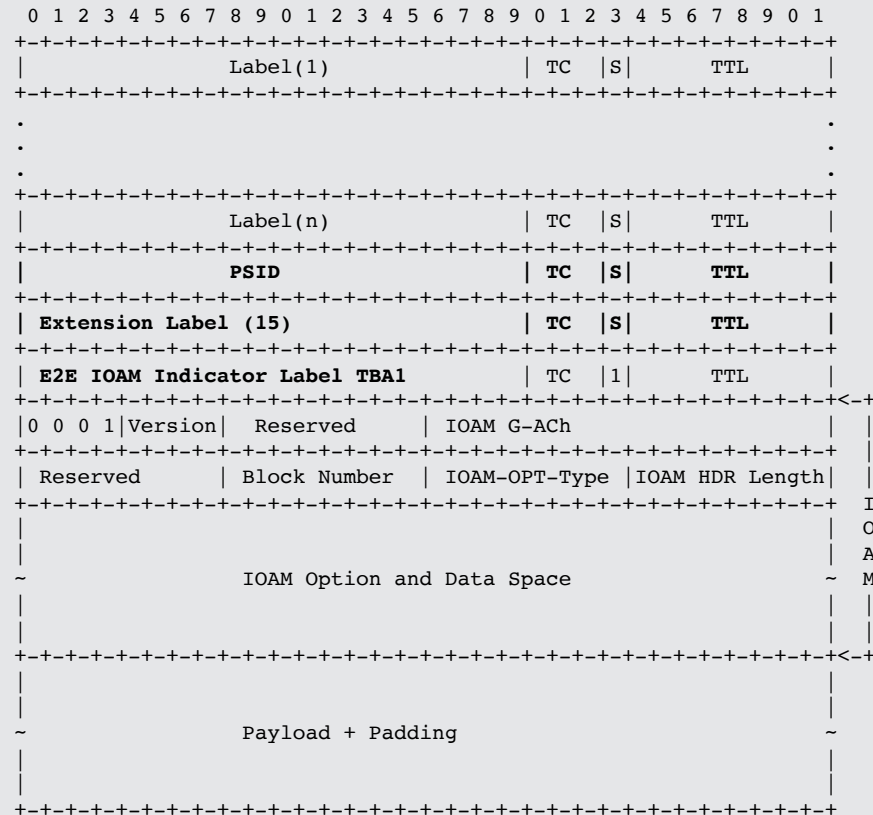


Figure: Example SR-MPLS Encapsulation with IOAM Data Fields

# IOAM Data Fields with Other Control Words/ACHs

- IOAM Data Fields are part of the MPLS encapsulation after the EOS, any other control word / ACH is added after the IOAM Data Fields in the data packet.
- The intermediate nodes can easily process the IOAM data field(s) after the EOS in the data packets.
- The decapsulating node removes the MPLS encapsulation including the IOAM Data Fields and then processes the other control word /ACH following it.
- *IOAM HDR Length* allows to find the Control word /ACH after the IOAM Data Fields.

## Example 1 - PW Encap Control Word [RFC4385] with IOAM Data Fields

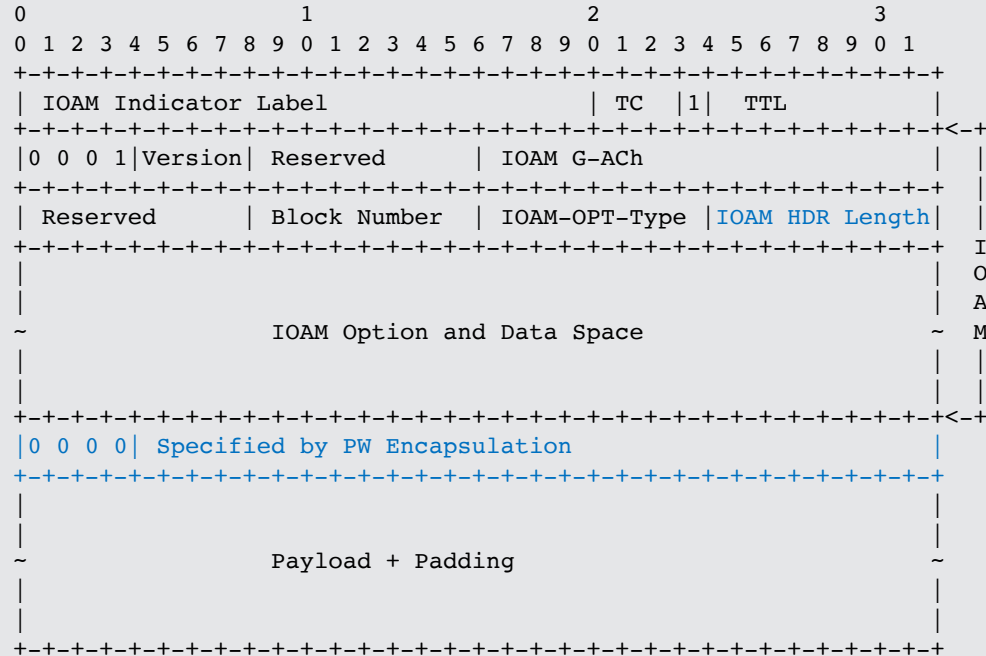


Figure: PW Encapsulation Control Word with IOAM Data Fields

## Example 2 - DetNet Control Word [RFC8964] with IOAM Data Fields

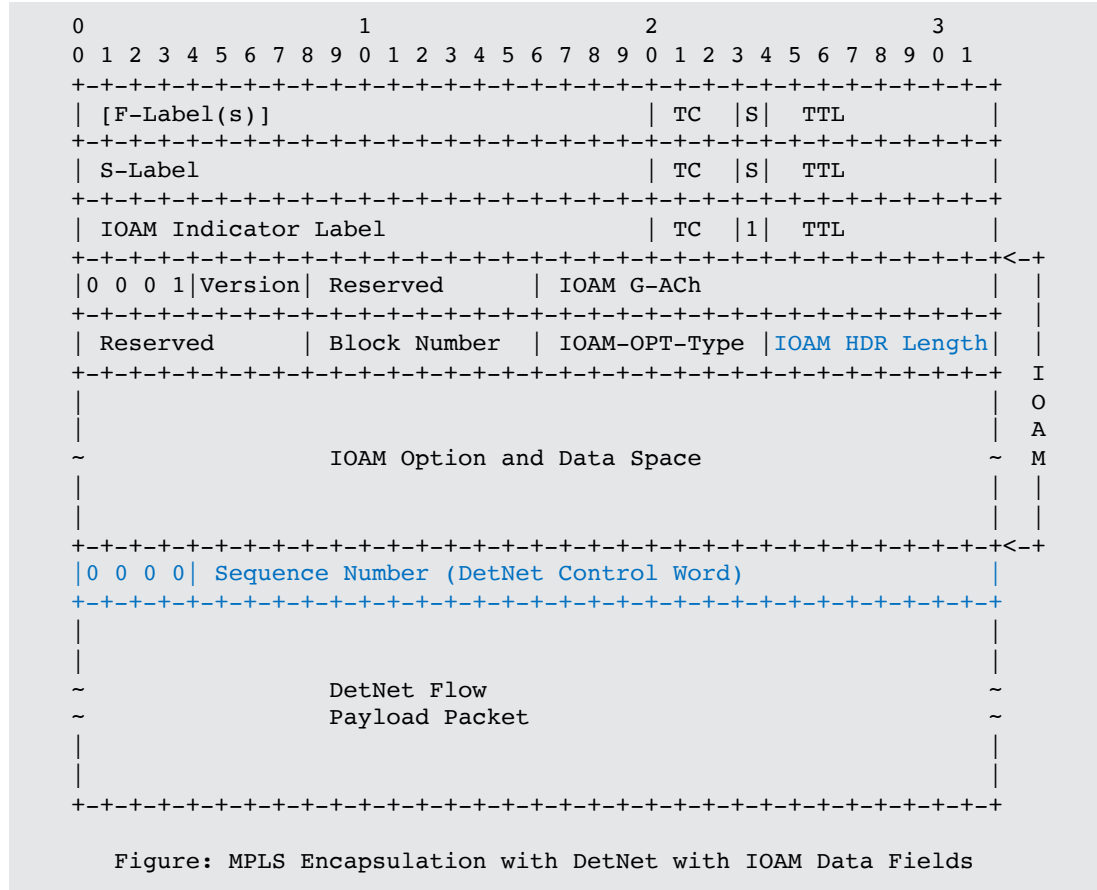


Figure: MPLS Encapsulation with DetNet with IOAM Data Fields



### Example 3 - DetNet Control Word [RFC8964] with IOAM Data Fields

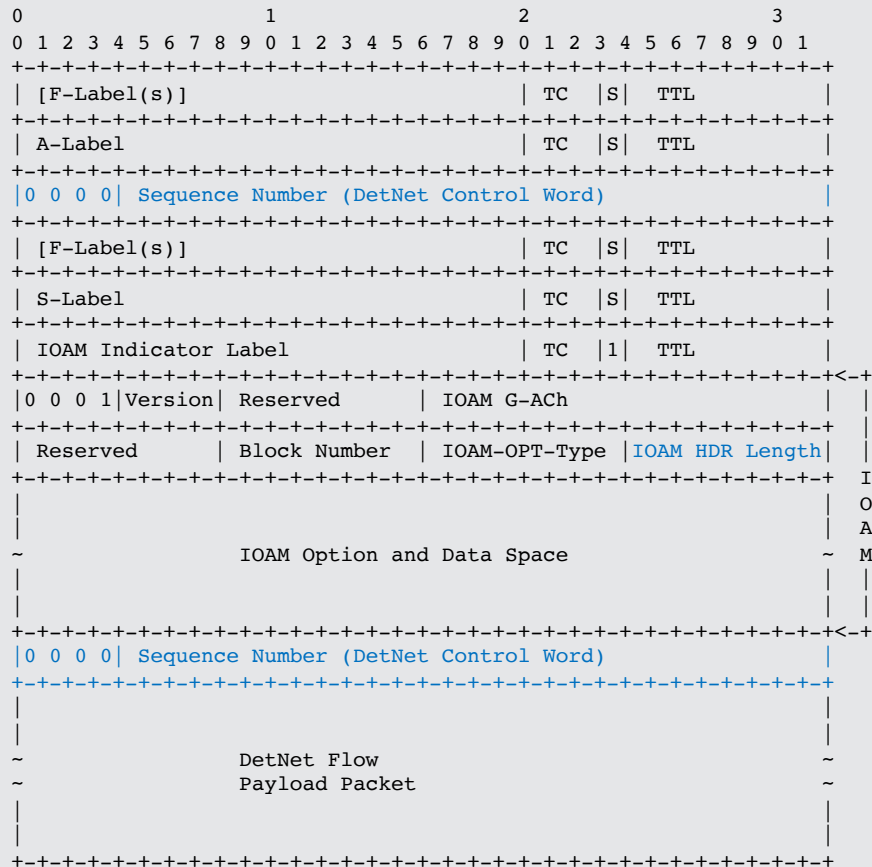


Figure: MPLS Encapsulation with DetNet with IOAM Data Fields

# Example 4 - Generic Delivery Function Encap with IOAM Data Fields

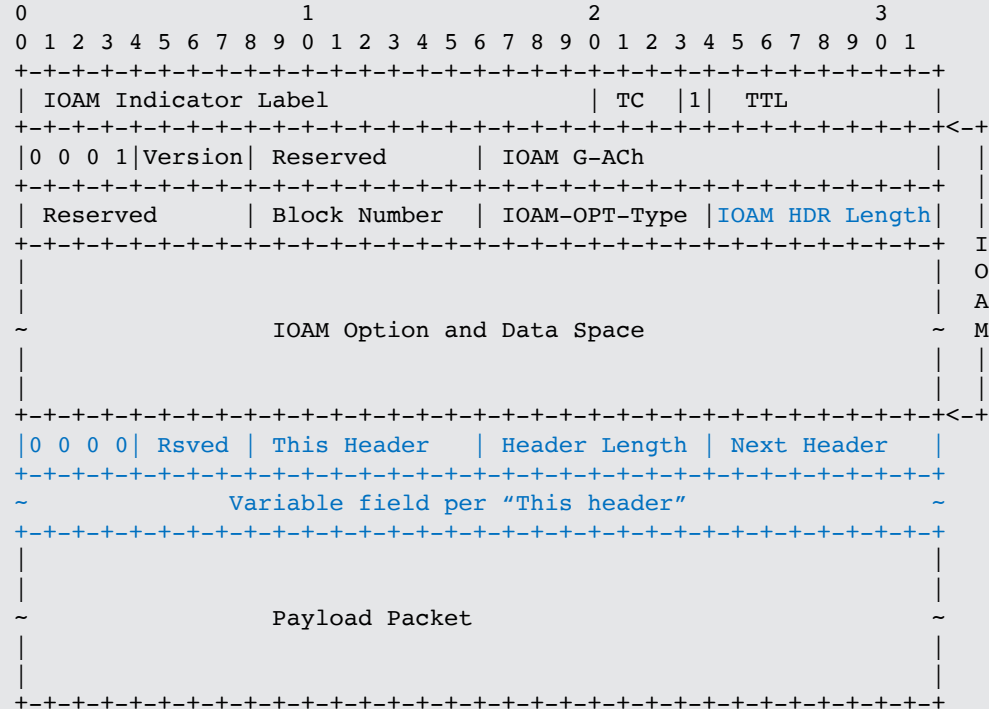
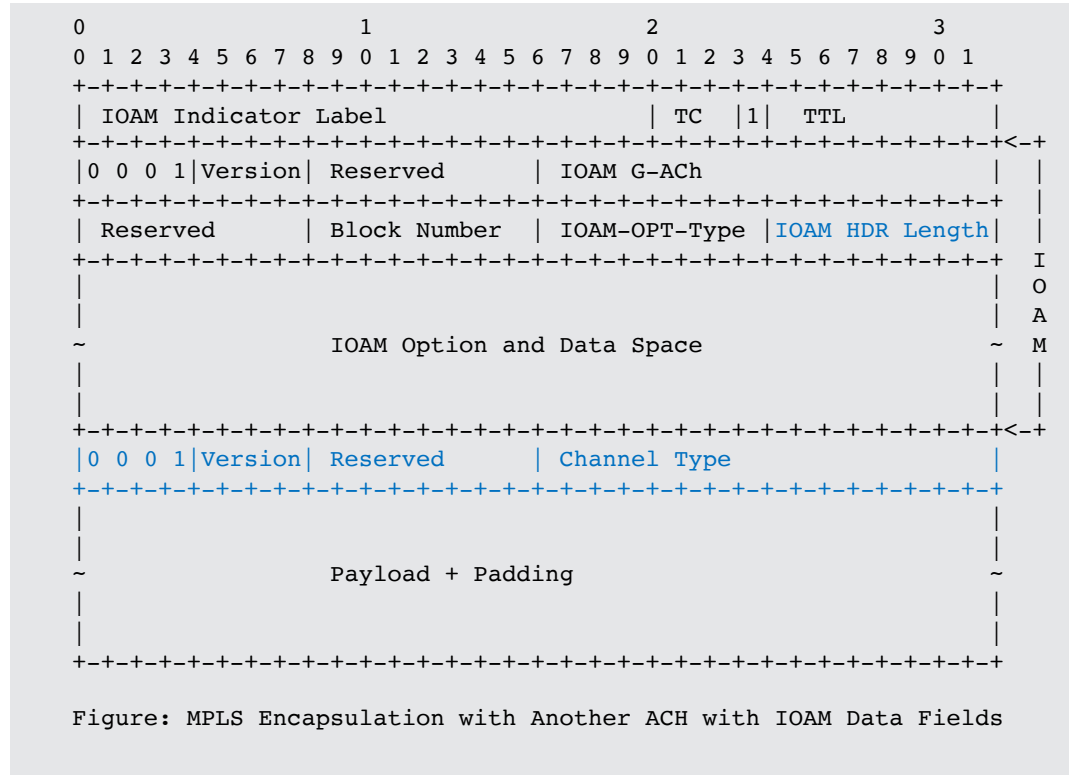


Figure: MPLS Encapsulation with Generic Delivery Functions with IOAM Data Fields

- <https://datatracker.ietf.org/doc/draft-zhang-intarea-generic-delivery-functions/>
- GDF Ingress/Egress Nodes only.
- GDF has no Hop-by-hop processing

## Example 5 - MPLS Encap with Another ACH with IOAM Data Fields



# Thank you