Multicast Information Model

draft-zhang-mboned-multicast-info-model-02

Mboned WG
IETF 100#Singapore

Sandy. Zhang
Linda. Wang
Ying. Cheng

Why introduce Multicast Info Model?

✓Existed multicast YANG models:

- These models describe different technologies for multicast;
- These models are distributed as separate file and focus on the protocol itself;
- They are device models;
- They cannot describe a high-level multicast information.

➢ Stand at a high level to take advantage of these models to control the multicast network to implement multicast service

What is Multicast Info Model?

- Provide a human readability of the whole multicast network;
- Frame different components and correlate them;
- Based on the human readable UML Class Diagram, instantiate these classes through YANG model;
- Take full advantage of and depend on existed multicast YANG models;
- Open for future multicast technologies;

Multicast Information Model 02 update

- Add Overlay Tech in the UML diagram.
- Add BIER-TE in Multicast Transport layer.
- Revise the model YANG program.

- This model has been verified in ODL BIER project. The project had been released in Carbon version.
- This model is feasible and practicable.

- http://www.opendaylight.org/

- OpenDaylight is a highly available, modular, extensible, scalable and multi-protocol c ontroller infrastructure built for SDN deployments on modern heterogeneous multi-v endor networks. OpenDaylight provides a model-driven service abstraction platform that allows users to write apps that easily work across a wide variety of hardware an d south-bound protocols.

- ODL employs a model-driven approach to describe the network, the functions to be performed on it and the resulting state or status achieved.

- By sharing YANG data structures in a common data store and messaging infrastructur e, OpenDaylight allows for fine-grained services to be created then combined togeth er to solve more complex problems. In the ODL Model Driven Service Abstraction Lay er (MD-SAL), any app or function can be bundled into a service that is then then load ed into the controller. Services can be configured and chained together in any numbe r of ways to match fluctuating needs within the network.

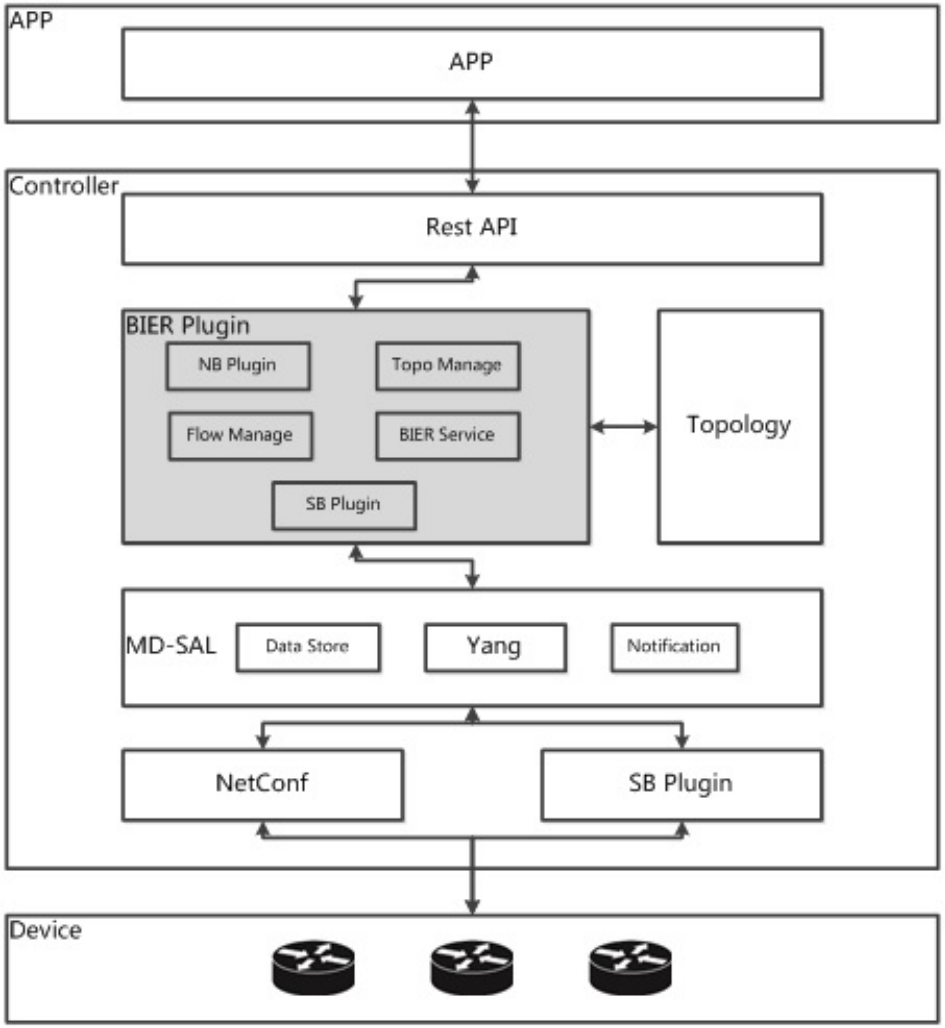
BIER project in ODL

https://wiki.opendaylight.org/view/BIER:Main

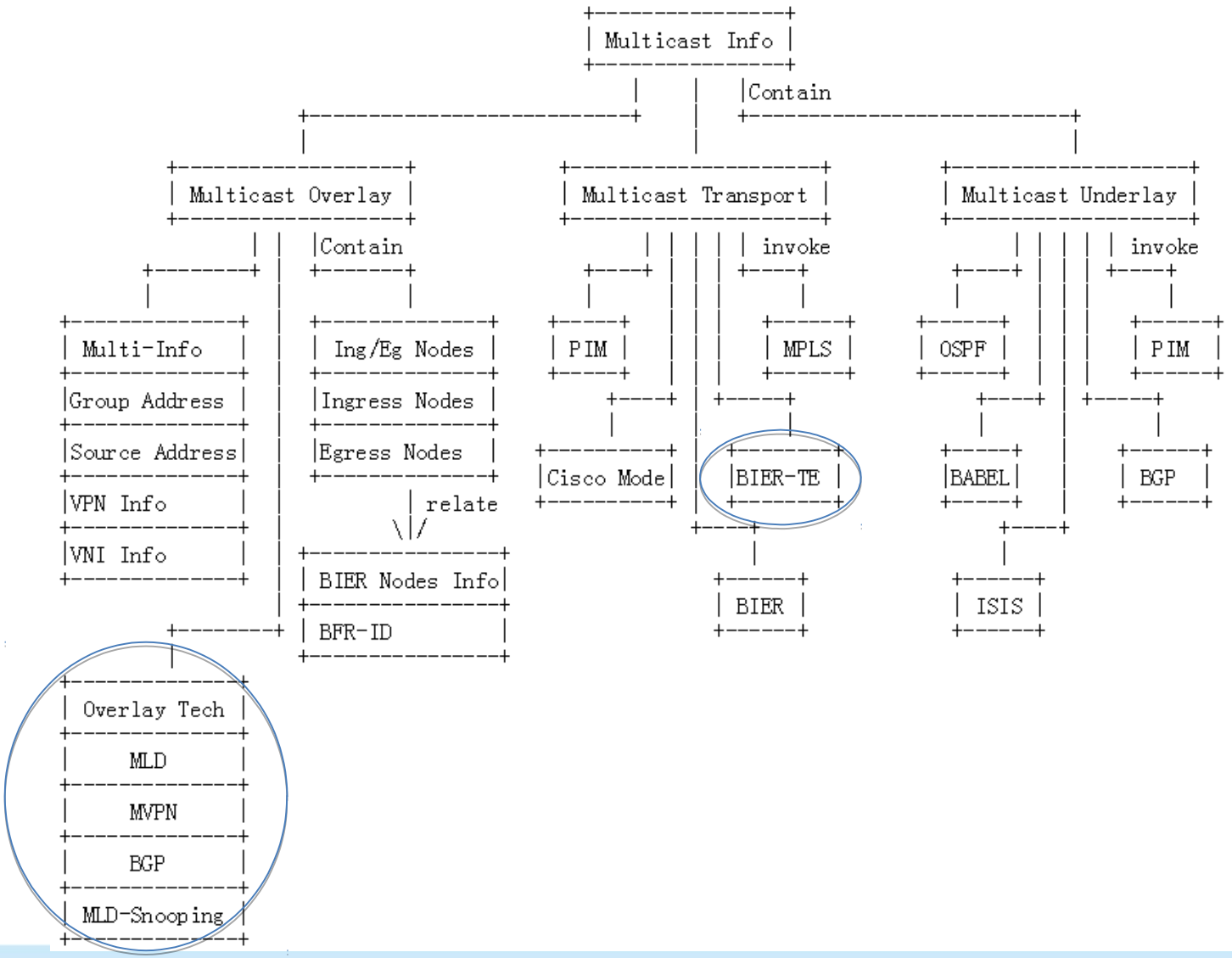
The BIER project is driven by two YANG models:

Multicast Information Model
draft-zhang-mboned-multicast-info-model-02

YANG Data Model for BIER Protocol
draft-ietf-bier-bier-yang-02



Multicast UML Class Diagram 02 update



Multicast Data Model Overview

module: ietf-multicast-information

+--rw multicast-information

+--rw multicast-info

……

+--rw multicast-overlay

……

+--rw multicast-transport

……

+--rw multicast-underlay

……

Divide the multicast data model into three layers as well.

Multicast Data Model – Information

+--rw multicast-info* [vpn-id source-address source-wildcard group-address group-wildcard vni-type vni-value]

- +--rw vpn-id uint32
- +--rw source-address inet:ip-address
- +--rw source-wildcard uint8
- +--rw group-address inet:ip-address
- +--rw group-wildcard uint8
- +--rw vni-type virtual-type
- +--rw vni-value uint32

Multicast info:

- ✓ Basic multicast flow information;
- ✓ Key of the multicast service.

Multicast Data Model – Overlay 02 update

```
+--rw multicast-overlay
      |  +--rw nodes-information
      |  |  +--rw ingress-node?   inet:ip-address
      |  |  +--rw egress-nodes* [egress-node]
      |  |     +--rw egress-node    inet:ip-address
      |  +--rw bier-information
      |  |  +--rw sub-domain?     sub-domain-id
      |  |  +--rw ingress-node?   bfr-id
      |  |  +--rw egress-nodes* [egress-node]
      |  |     +--rw egress-node    bfr-id
      |  +--rw overlay-technology
      |     +--rw (overlay-tech-type)?
      |        +--:(mld)
      |        +--:(mvpn)
      |        +--:(bgp)
      |        +--:(mld-snooping)
```

Overlay layer includes:

- ✓ Ingress/egress nodes information;
- ✓ Overlay technology.

Multicast Data Model – Transport 02 update

```
+--rw multicast-transport
      |  +--rw bier
      |  |  +--rw sub-domain?       sub-domain-id
      |  |  +--rw (encap-type)?
      |  |  |  +--:(mpls)
      |  |  |  +--:(non-mpls)
      |  |  |  +--:(ipv6)
      |  |  +--rw bitstringlength?   uint16
      |  |  +--rw set-identifier?    si
      |  |  +--rw ecmp?              boolean
      |  |  +--rw frr?               boolean
      |  +--rw bier-te
      |  |  +--rw sub-domain?       sub-domain-id
      |  |  +--rw (encap-type)?
      |  |  |  +--:(mpls)
      |  |  |  +--:(non-mpls)
      |  |  +--rw bitstringlength?   uint16
      |  |  +--rw set-identifier?    si
      |  |  +--rw ecmp?              boolean
      |  |  +--rw frr?               boolean
      |  +--rw cisco-mode
      |  |  +--rw p-group?           inet:ip-address
      |  |  +--rw graceful-restart?  boolean
      |  |  +--rw bfd?               boolean
```

Transport layer includes:

- ✓ Transport technology type
- ✓ Corresponding individual YANG models

```
|  +--rw mpls
      |  |  +--rw (mpls-tunnel-type)?
      |  |    +--:(mldp)
      |  |    |  +--rw mldp-tunnel-id?      uint32
      |  |    |  +--rw mldp-frr?            boolean
      |  |    |  +--rw mldp-backup-tunnel?  boolean
      |  |    +--:(p2mp-te)
      |  |       +--rw te-tunnel-id?        uint32
      |  |       +--rw te-frr?              boolean
      |  |       +--rw te-backup-tunnel?    boolean
      |  +--rw pim
      |     +--rw graceful-restart?  boolean
      |     +--rw bfd?               boolean
```

Multicast Data Model – Underlay 02 update

```
+--rw multicast-underlay
        +--rw underlay-requirement?   boolean
        +--rw bgp
        +--rw ospf
        |  +--rw topology-id?   uint16
        +--rw isis
        |  +--rw topology-id?   uint16
        +--rw babel
        +--rw pim
```

Underlay layer includes:

- ✓ Underlay technology type
- ✓ Corresponding individual YANG models

Next Steps

- Any comments ?
- WG adoption?