A YANG DATA MODEL FOR ACTN VN OPERATION

draft-lee-teas-actn-vn-yang-08

Young Lee, Huawei
Dhruv Dhody, Huawei
Daniele Ceccarelli, Ericsson
Takuya Miyasaka, KDDI
Peter Park, KT
Bin Yeong Yoon, ETRI

Introduction

- A Yang Data Model for ACTN VNS operations
  - Virtual Network Service

- And this is for CMI
  - i.e. CNC – MDSC Interface of ACTN
  - Aligned to Customer Service Model
  - Business Boundary Between Customer & Network Provider

- VN Yang model for
  - Access Points
    - AP, VNAP
  - VN, VN-member
  - Constraint, Policy etc.
  - *Things that customer cares for…*

- Updates in the latest version
  - Aligned to ACTN framework WG draft
    - VN Types and VNS Types
  - Aligned to ACTN Information Model WG draft
  - Yang Model as per NMDA

A Virtual Network Service (VNS) is the service agreement between a customer and a provider to provide a VN.

- Type 1 VN is a set of edge-to-edge links (a Type 1 VN). Each link is referred as a VN member and is formed as an end-to-end tunnel across the underlying networks.

- Type 1 VNS refers to a VNS in which the customer is allowed to create and operate a Type 1 VN.

- Type 2 VN is a topology of virtual nodes and virtual links.

- With a Type 2a VNS, the VN is statically created at service configuration time and the customer is not allowed to change the VN.

- A Type 2b VNS is the same as a Type 2a VNS except that the customer is allowed to make dynamic changes to the initial VN topology created at service configuration time.

Type 1 VNS

- Customer defines a VN.

- Each VN-member identifies Source and Destination AP
  - Identified by VNAP

- All constraints, policies etc. are set for VN as a whole (i.e. on all VN members)

- MDSC is responsible for translating and mapping the VN into specific network centric-models (e.g., TE-tunnels) to coordinate the multi-domain network operations with PNCs.

Create VN {
End Points
VN Members
QoS Metrics for VN
Policy... }
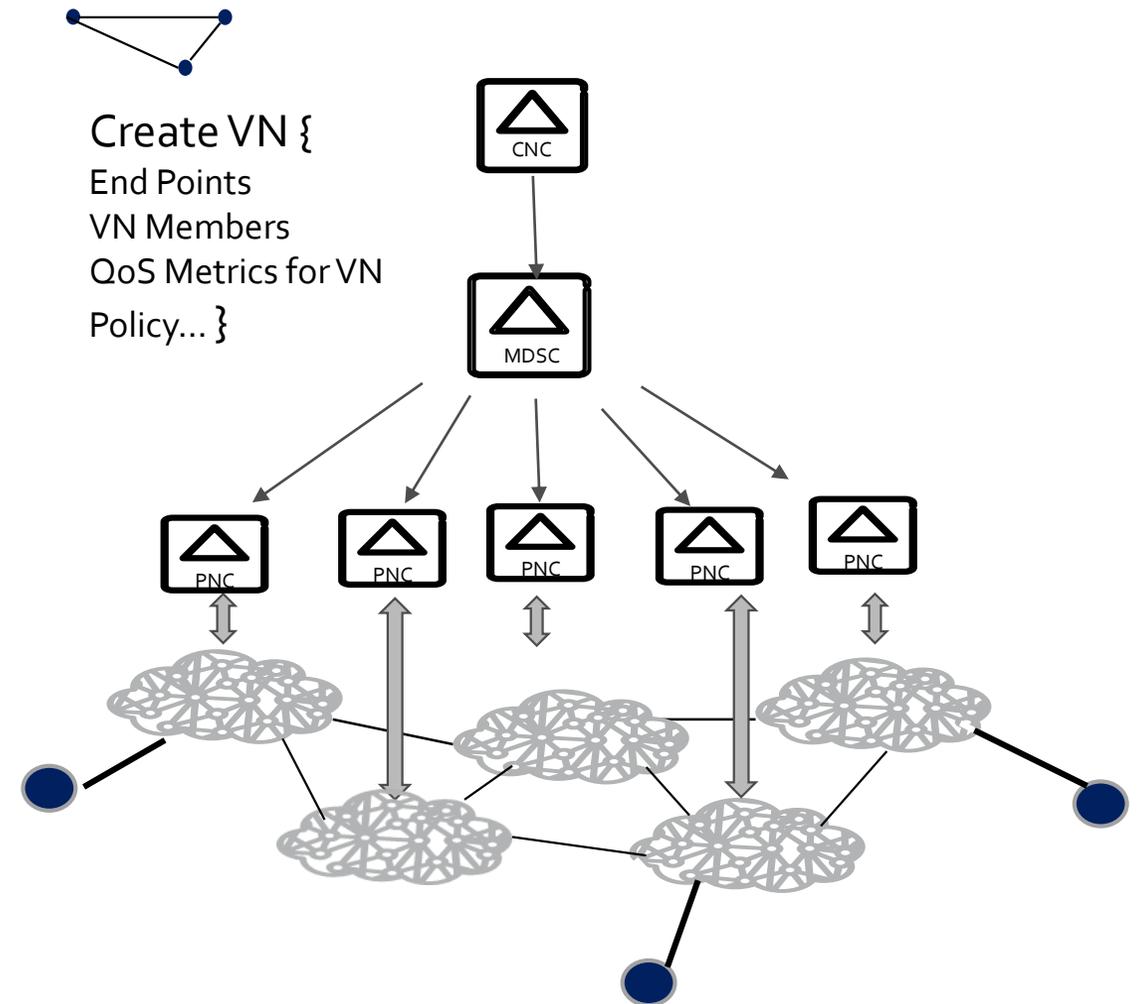
Type 2 VNS

- Allow the customer to also configure (Type 2a) or learn a VN topology via the TE topology yang model [TE-TOPO].

- A reference to the abstract topology is maintain in the VN Yang model.

- Any change in topology (Type 2b VN) would be made via [TE-TOPO].

- Using the path in VN-member in the ACTN VN yang model, CNC can also set a path based on the VN Topology. The MDSC translates the abstracted path to tunnels via [TE-Tunnel] or similar path setup operation.

- Rest of the procedures, remains the same

1
Create
VN

CNC

MDSC

PNC PNC PNC PNC PNC
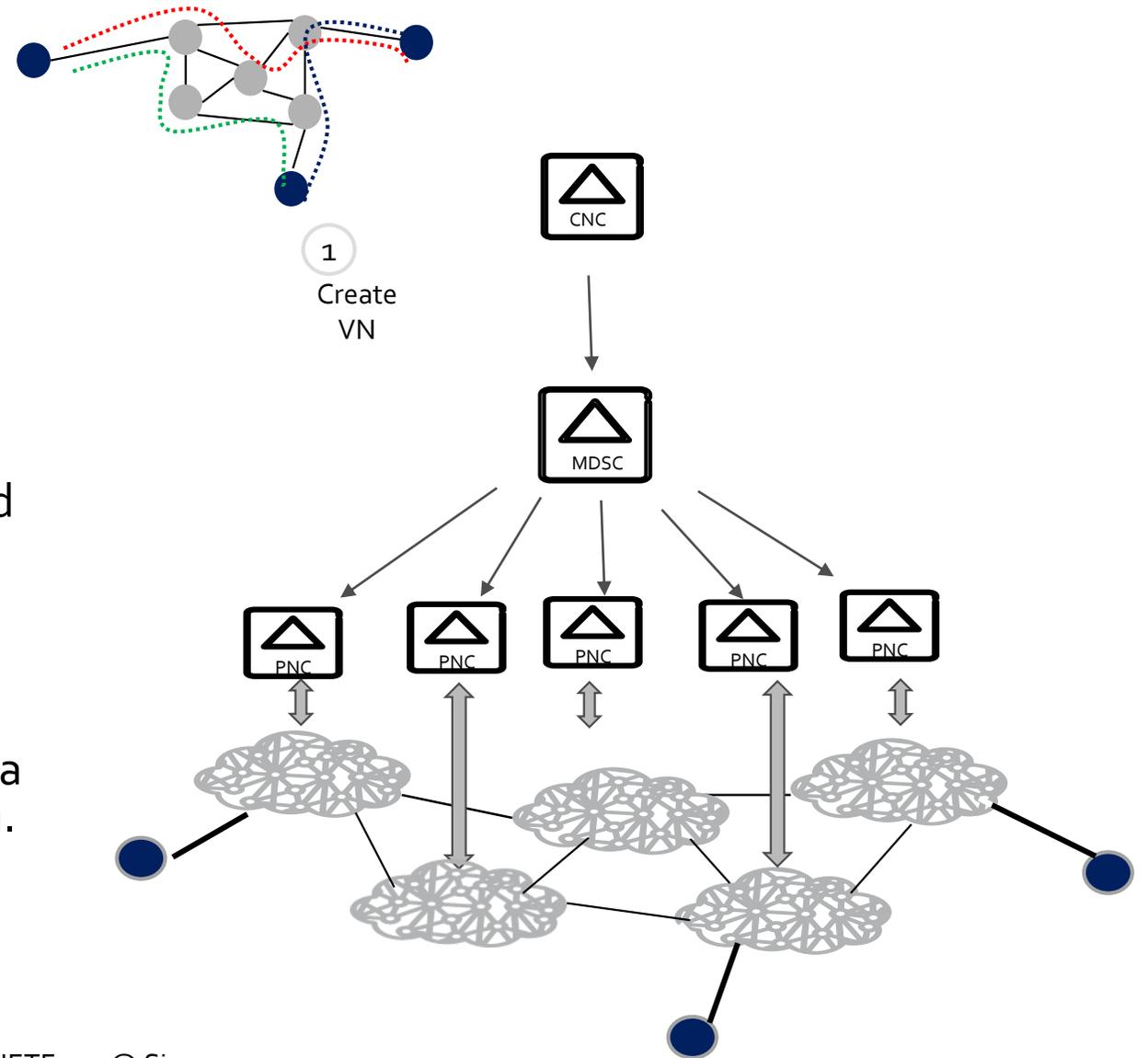
Why?

Customer Service Model

- This allows the customer to instantiate and view the VN as one entity, making it easier for some customers to work on VN without worrying about the details of the provider based YANG models.
- Better to maintain a set of E2E Tunnels as one VN unit for applying policy, reroute, protection, restoration, etc., rather than treating each TE-tunnel as individual unit.
- Service models do not make any assumption of how a service is actually engineered and delivered for a customer.
- In some scenarios customer does not explicitly create the VN via this model, but a VN is auto-instantiated by the MDSC based on other service model (e.g. L3SM), the operational state of this model is still very useful for the customer service operations.

Benefit of VN model over TE-tunnel model

Gets more complex and operationally difficult as the networks grows!
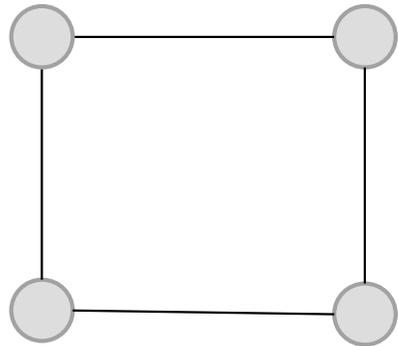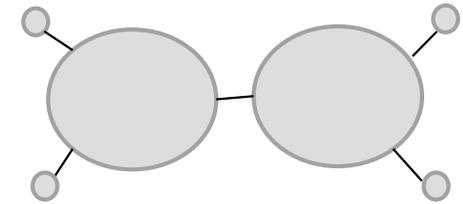
VN Model
- Need to set up one VN model with four VN member
- They are all correlated to each other.
- All properties set together!
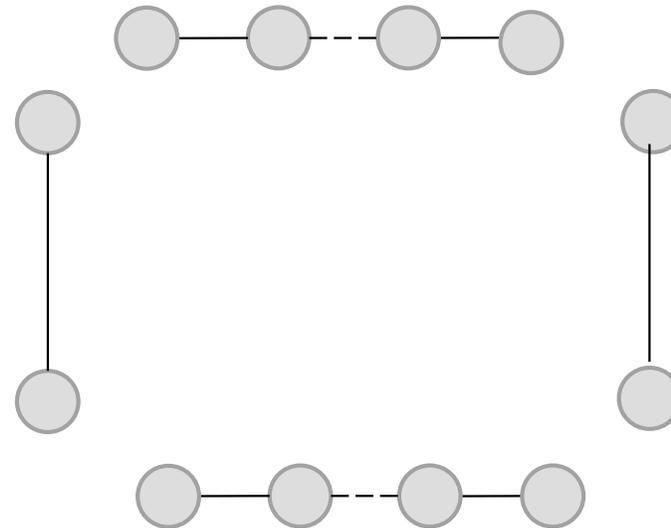
TE Tunnel Model
- Need to set up 6 separate tunnels with two stitching points
- They are all independent and difficult to correlate.

Why?

Innovative Services to the Customer

- VN Compute
  - A pre-instantiation mode to view the full VN as a single entity before instantiation.
  - Achieving this via path computation or "compute only" tunnel setup does not provide the same functionality on a full VN.
- Multi-source / Multi-destination
  - Sources or destinations or both may not be pre-determined by the customer.
  - For instance, for a given source, there may be a list of multiple-destinations to which the optimal destination may be chosen depending on the network resource situations.
  - The MDSC would pick a destination, MDSC is also notify CNC if it wish to change the destination.

Others

- Mapping VN to services (L3SM, L2SM, L1CSM)
  - Easily augmented to support the mapping of services to a single VN (L3SM and L2SM)
- Telemetry, performance monitoring, network autonomics (auto-scaling)

Multi-destination case

D1

S

D2

D3
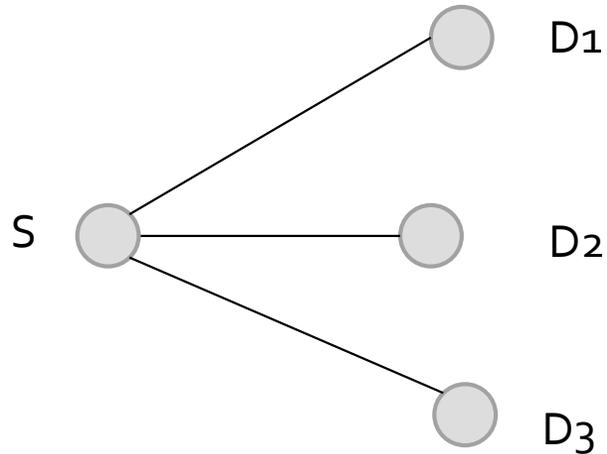
VN YANG has a way to express
Its policy for the network to select
The best one among S-D1, S-D2, or S-D3
with the model.

With TE-tunnel Model, there is no good way to
do this operation except:

- RPC three path compute Requests to MDSC:
    i) S-D1, ii) S-D2, iii) S-D3
- Then CNC received each of the three results
- Then the CNC selects the best one out of the
  three. (Say S-D2 is the best based on the Path
  Compute result from the MDSC)
- Then the CNC requests for S-D2 Tunnel creation
  to the MDSC.

This is way too complex and take much longer.

And when S-D2 has issues, it is not easy to move things to D1
or D3.

Summary

Ability to support various VN & VNS Types as per the customer view

- Edge to edge links (type 1)
- VN topology (type 2)

Maintenance of AP and VNAP along with VN and map easily to the services.

New Services

- VN Compute (pre-instantiate)
- Multi-Source / Multi-Destination

ACTN VN Yang Model

Reference to TE Topology

Access Point

List of VNAP, inside each AP

List of VN Member inside each VN

Configure abstract path

Handle Multi-src / Multi-dest

```
module: ietf-actn-vn
    +--rw actn
        +--rw ap
            +--rw access-point-list* [access-point-id]
                +--rw access-point-id        uint32
                +--rw access-point-name?     string
                +--rw src-tp-id?             binary
                +--rw dst-tp-id?             binary
                +--rw max-bandwidth?         te-types:te-bandwidth
                +--rw avl-bandwidth?         te-types:te-bandwidth
                +--rw vn-ap* [vn-ap-id]
                    +--rw vn-ap-id    uint32
                    +--rw vn?         -> /actn/vn/vn-list/vn-id
```

```
+--rw vn
    +--rw vn-list* [vn-id]
        +--rw vn-id                  uint32
        +--rw vn-name?               string
        +--rw vn-topology-id?        te-types:te-topology-id
        |       {type-2}?
        +--rw vn-member-list* [vn-member-id]
            +--rw vn-member-id    uint32
            +--rw src
            |   +--rw src?             leafref
            |   +--rw src-vn-ap-id?    leafref
            |   +--rw multi-src?       boolean
            |         {multi-src-dest}?
            +--rw dest
            |   +--rw dest?            leafref
            |   +--rw dest-vn-ap-id?   leafref
            |   +--rw multi-dest?      boolean
            |         {multi-src-dest}?
            +--rw path {type-2}?
            |   +--rw path-element* [path-element-id]
            |       +--rw path-element-id    uint32
            |       +--rw index?             uint32
            |       +--rw address?           te-types:te-tp-id
            |       +--rw hop-type?          te-types:te-hop-type
            +--ro metric* [metric-type]
            |   +--ro metric-type    identityref
            |   +--ro value?         uint32
```

```
        |   +--ro oper-status?     identityref
        |   +--ro tunnel-ref?      te:tunnel-ref
        +--ro multi-src-dest {multi-src-dest}?
        |   +--ro selected-vn-member?   leafref
        +--rw objective-function?   pcep:objective-function
        +--rw metric* [metric-type]
        |   +--rw metric-type    identityref
        |   +--rw limit
        |   |   +--rw enabled?   boolean
        |   |   +--rw value?     uint32
        |   +--rw optimize
        |       +--rw enabled?   boolean
        |       +--rw value?     uint32
        +--rw bandwidth?            te-types:te-bandwidth
        +--rw protection?          identityref
        +--rw local-reroute?       boolean
        +--rw push-allowed?        boolean
        +--rw incremental-update?  boolean
        +--rw admin-status?        identityref
        +--ro oper-status?         identityref
```

ACTN VN Model as Customer Service Model

- ACTN VN model is the customer view of the VN.
  - It does not have finer details from the provider point of view.

- Positioning the TE-Tunnel and TE-Topo models as both Customer Service Model (as well as Network/Device Model)
  - Is theoretically possible!
  - But, is this operationally correct?
    - Different yang models at service, network and device layers are generally preferred!
      - Layers are good ☺

ACTN VN Model

- Does it makes sense to also have an ACTN VN Yang model as part of CMI as per the ACTN framework document?
  - Which models VN as a whole

- Is ACTN VN Model a good base for this?

Next Step

- Aligned to the philosophy of customer service model

- Work well with other ACTN and related TE yang models

- WG Adoption?