

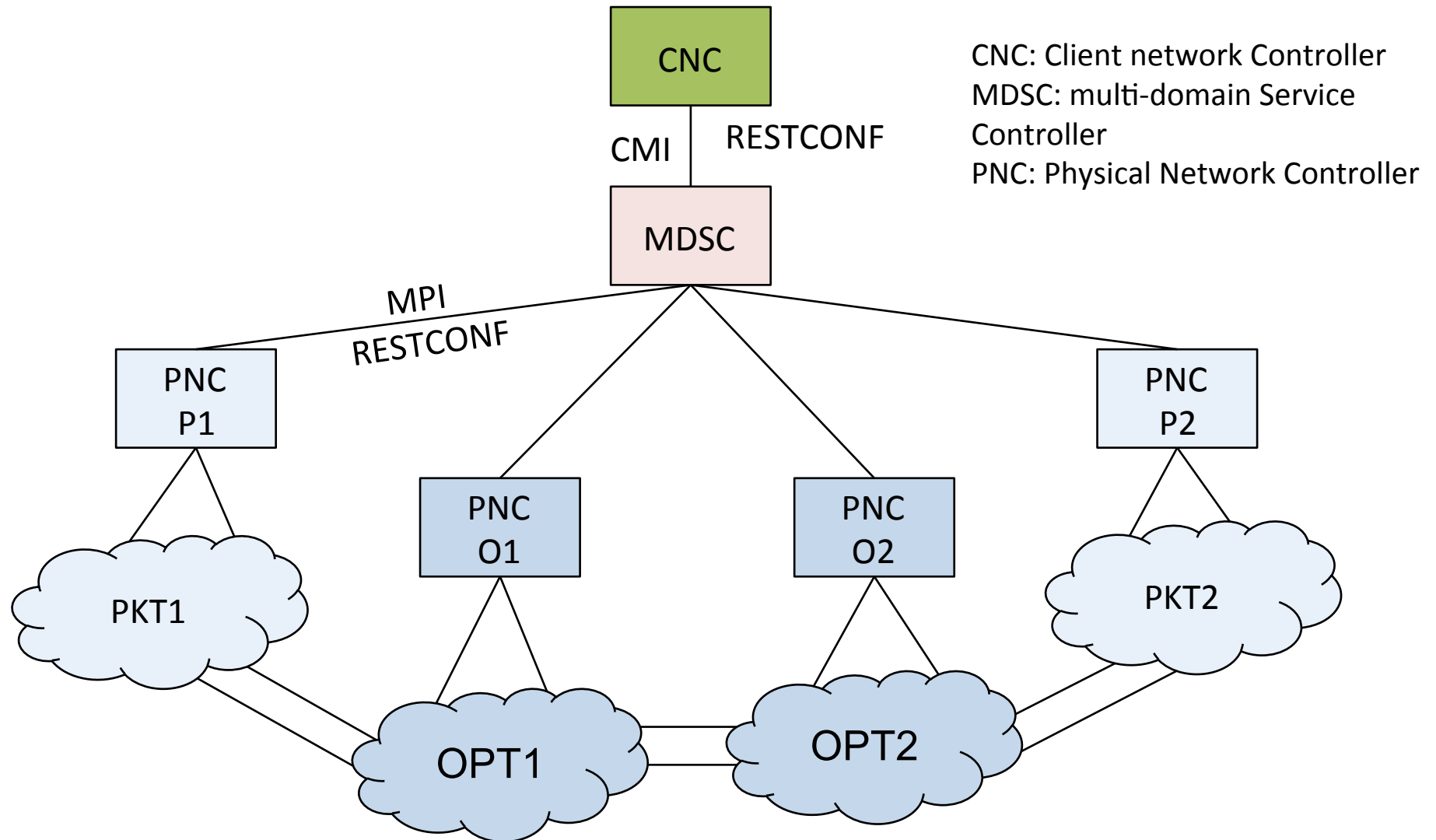
IETF Transport Model Instance Diagrams

June 11th, 2016

Use Case - 3

EPL Point-to-Point Service,
Multi-domain Topology

Reference Controller Hierarchy

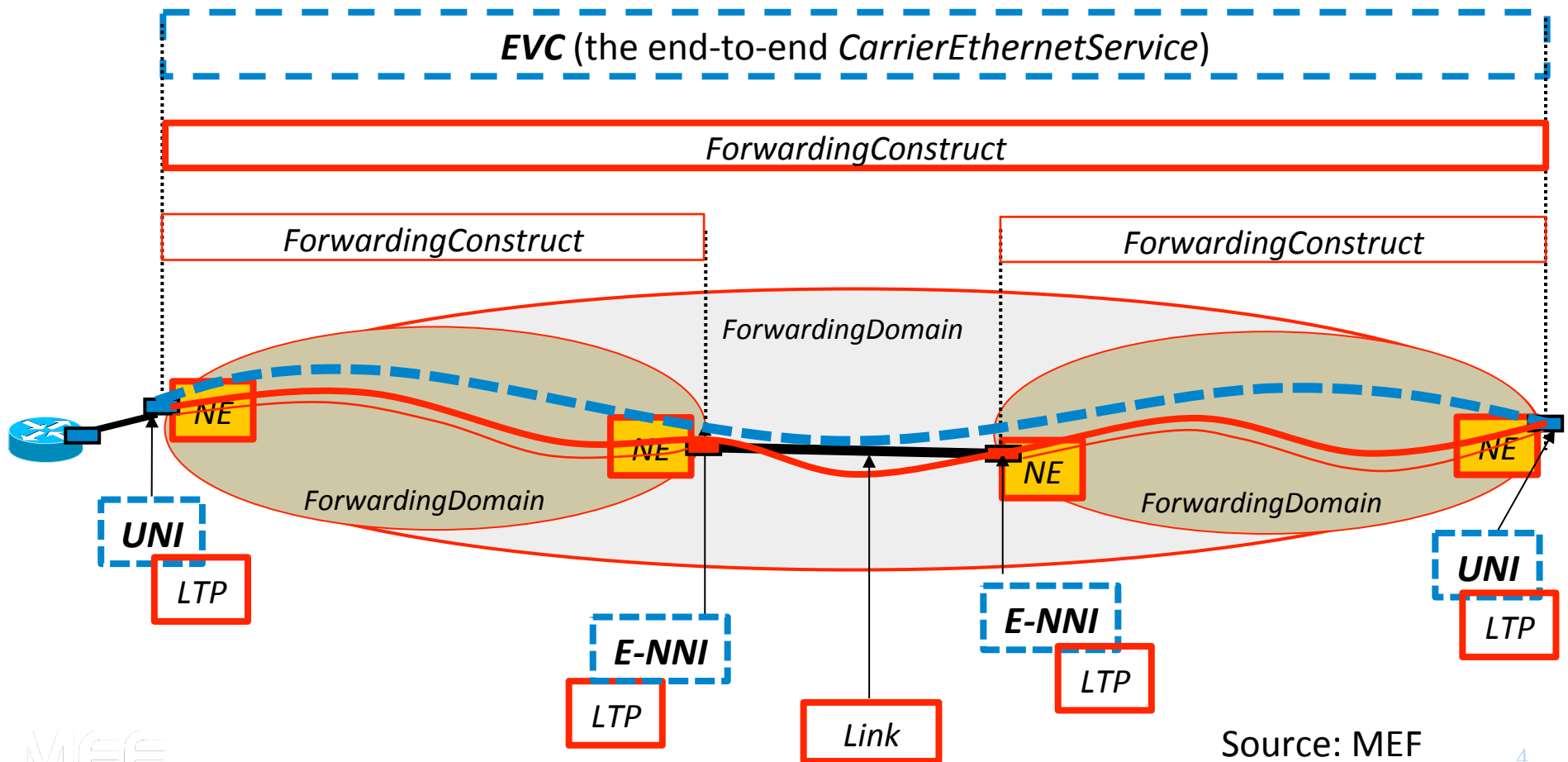


Mapping from Service/EVC to ForwardingConstruct

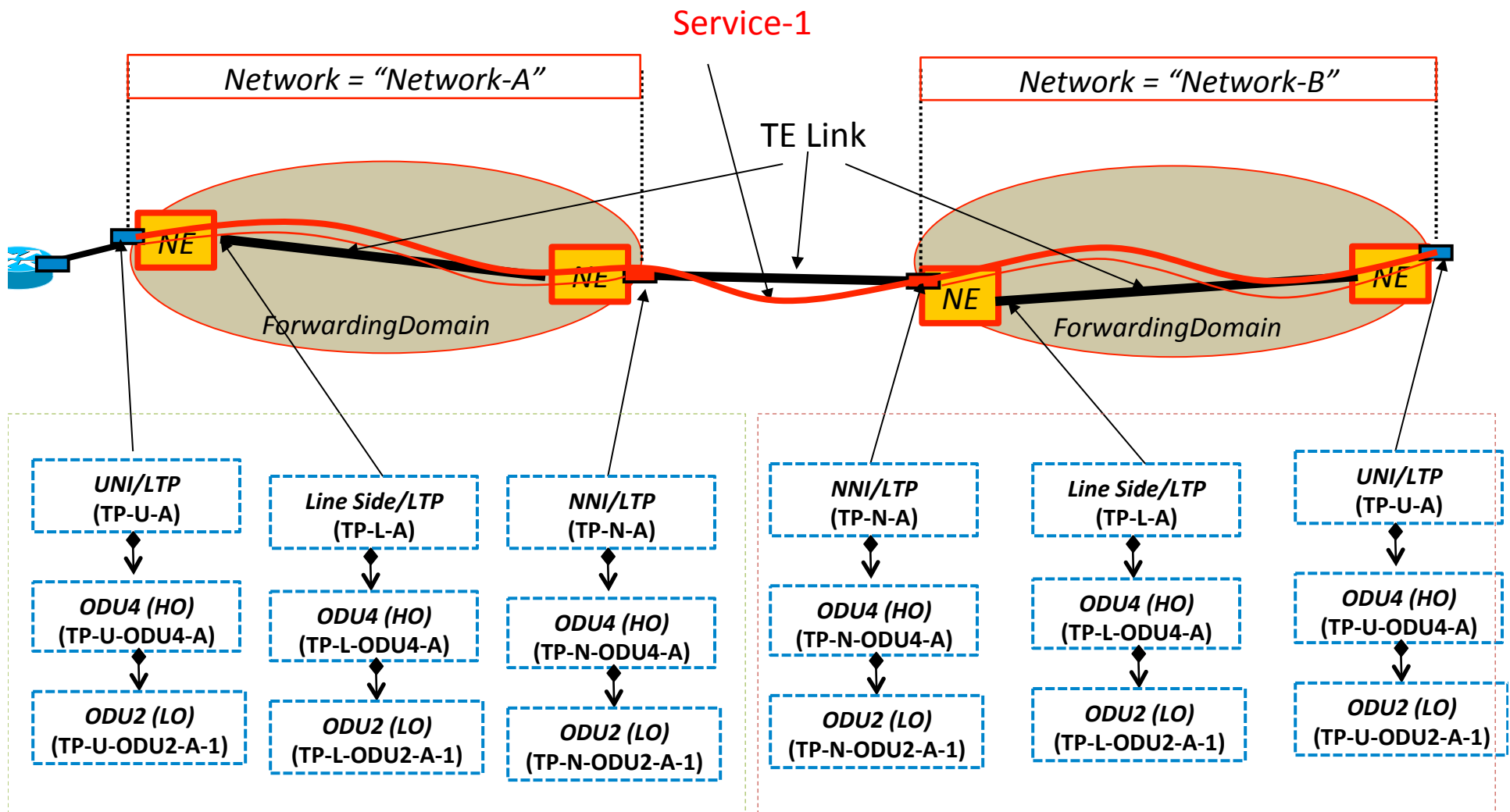
Single Provider, single managed FD, partitioning

managed object classes at Service level
(potentially appearing at Legato, Interlude)

managed object classes at Resource level
(potentially appearing at Presto)

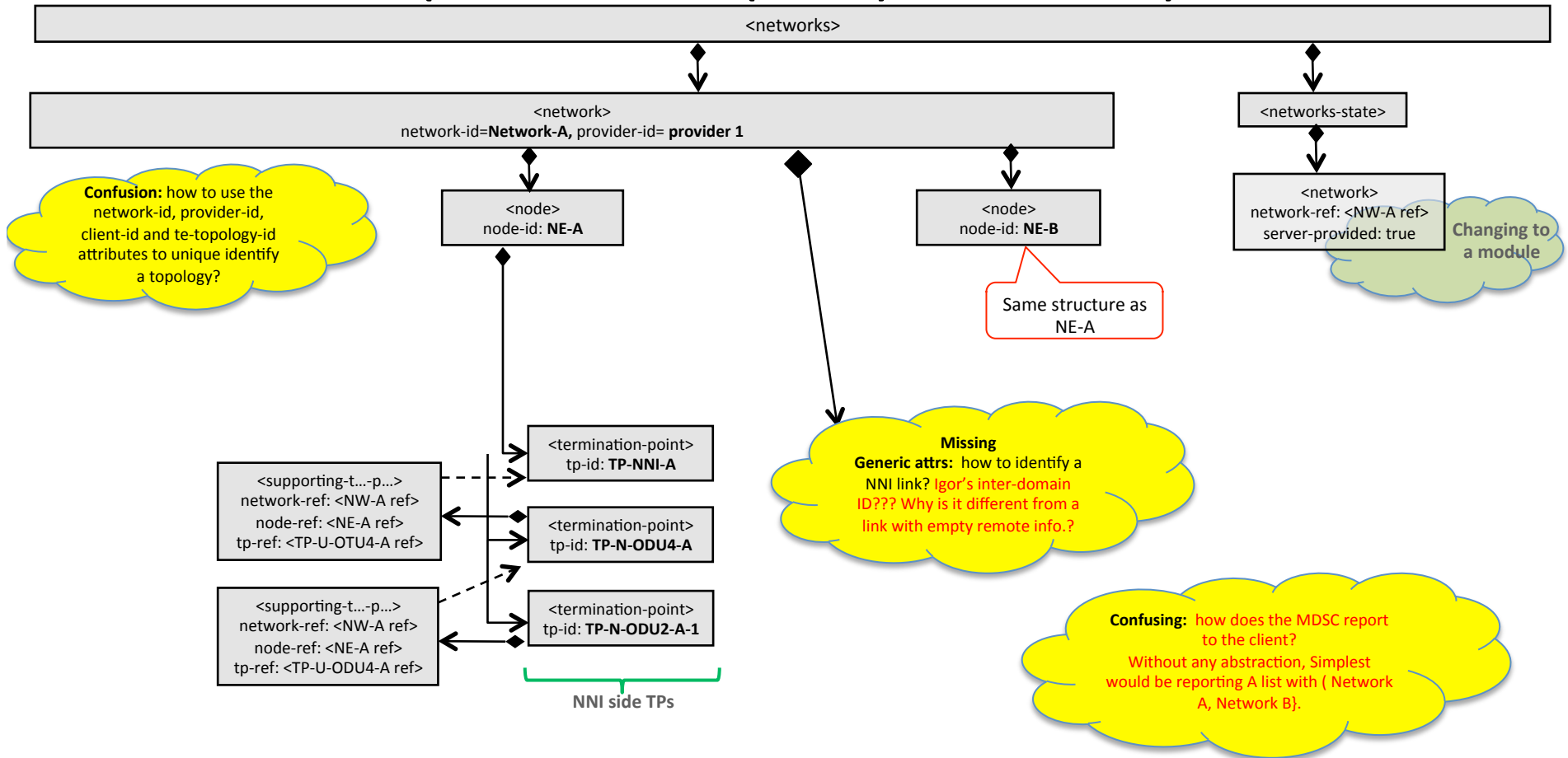


What a MDSC(multi-domain service controller) see and can report to its northbound client:



Note: each domain controller can only report partial of the topology shown above. More Details provided in the next slide.

Network Topology (teas) Model Instantiation (network-A(or B) to MDSC)

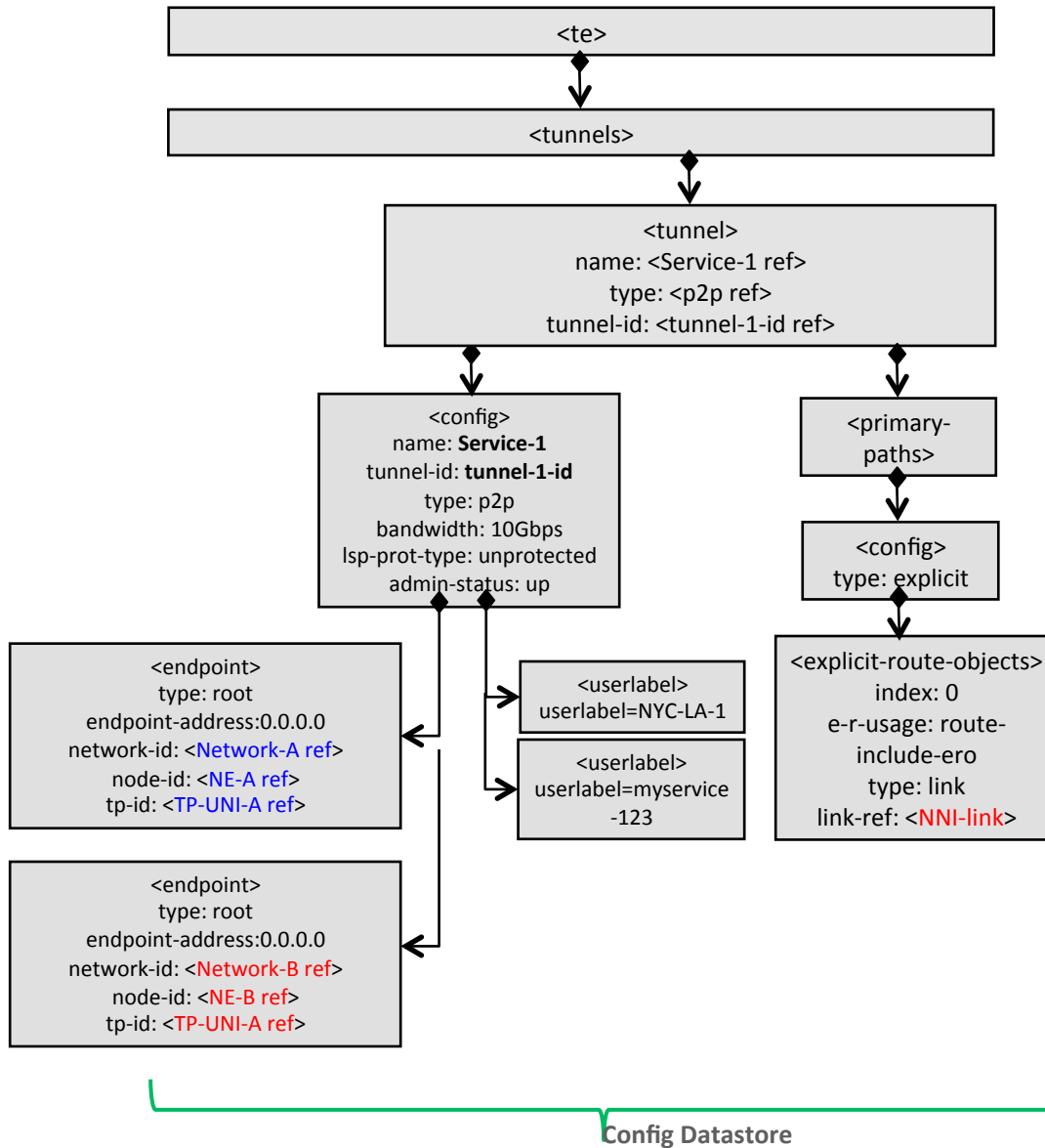


Note:

- 1: intra-domain link not shown, captured in Use case 1;
- 2: UNI and Line side TP not shown, captured in Use case2;
- 3: missing information is on top of previous two use cases;

Note: Assume all the entities within the two Networks are identified using the same ID Except the network-id is different.

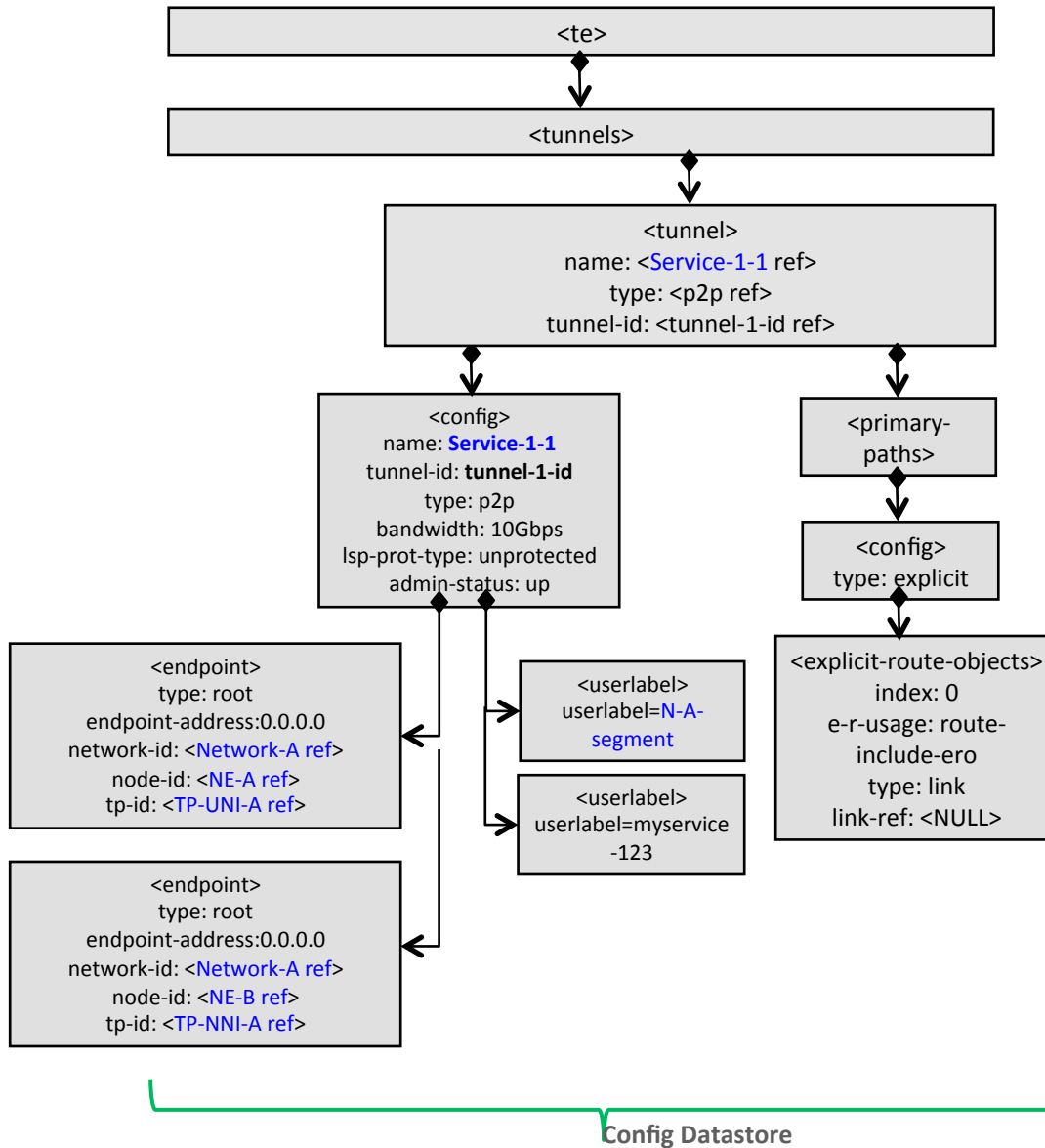
TEAS Tunnel Model Instantiation (client => MDSC)



Confusion: Tunnel has to start and end on a tunnel-termination point. What if a client does not know the TTP?

Missing: network-id to scope the NNI-link-id?

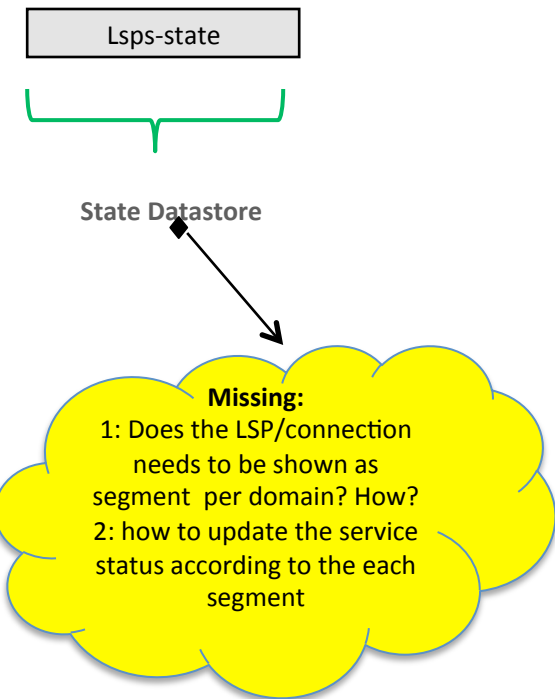
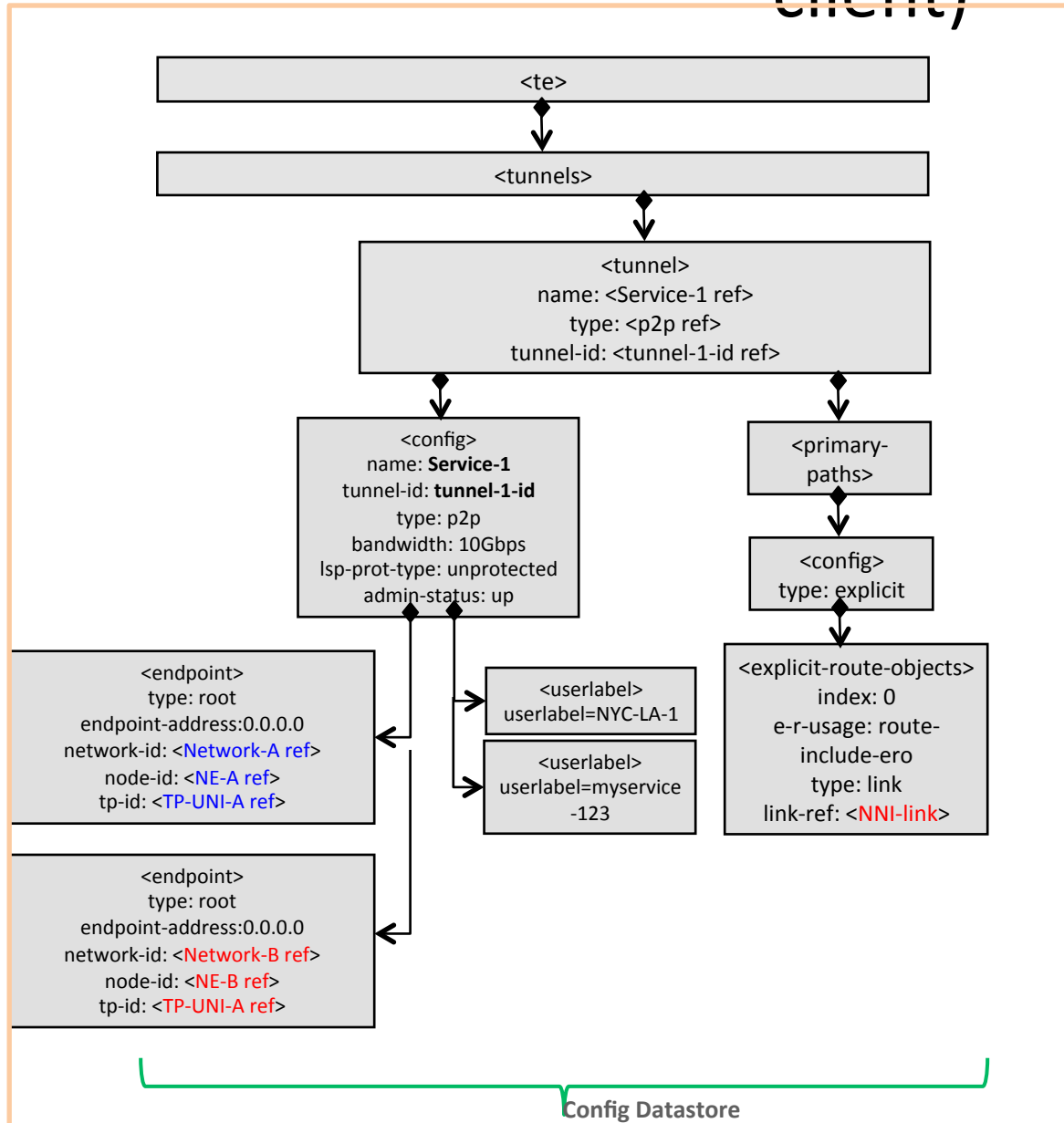
TEAS Tunnel Model Instantiation (MDSC=> PNC A (or B))



Assume: inter-domain link is ODU4/OTU4 link.

This will trigger PNC A/B to set up ODU4 tunnel If there is none available.

TEAS Tunnel Model Instantiation (MDSC=> client)

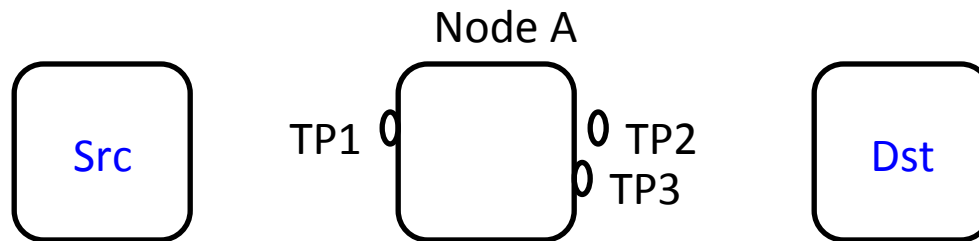


Shown in Page 6

Other points that need clarification TE.yang model

- Not particular about this use case, but important.
 1. A service may have two tunnel/lsp associated with different access link as a starting point. Any support of this?
 2. Why primary path is not under config tunnel? Why having a list of primary paths and each with a set of secondary paths? Why not make them in parallel with a role attached? [for transport: only two paths usually]
 3. A parameter to show it is loose or strict for path config? When it is loose, need a parameter to show whether it is ingress or egress? [an example shown in next page]
 4. Objective function needed: min hop; min delay, min distance;
 5. Adding revertive-type? And reversion-lock?
 6. Preference is not key, why put it outside of config and with key?
 7. Tiebreaker-type, tunnel-path-affinities: meaning?
 8. No-cspf: meaning? does it mean no show of this parameter meant it is CSPF?
 9. Lockdown: meaning? Does it mean no show of this parameter meant the entity will do auto-reoptimization?
 10. Why do we need origin type? [Anurag already covered this]
 11. Transport will not remove the original path, so need a leaf to show which path to revert back to?
 12. Lsp-operational-status: meaning? Suggest a name change to avoid misunderstanding.
 13. Need to add 32-bit node ID and 32bit TP-ID in RRO;
 14. What does the flag mean after the label leaf?
 15. Tunnel type [Anurag already covered this]
 16. Life-cycle state covered?
 17. Only mentioned using NETCONF, also should include support using RESTCONF.

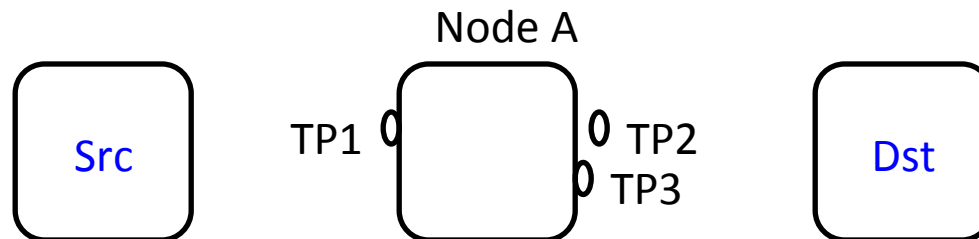
Case A



Loose ERO as: NodeA/TP1 (**ingress**)

It will be like: Src-**NodeA/TP1-NodeA/TP2**-Dst

Case B



Loose ERO as: NodeA/TP1 (**egress**)

It will be like: Src-**NodeA/TP2-NodeA/TP1**-Dst