DC aware TE topology model

draft-llc-teas-dc-aware-topo-model-05

Luis M. Contreras *(Telefónica)*

Xufeng Liu *(Alef Edge)*

IETF 124, TEAS WG, Montreal, November 2025

draft-llc-teas-dc-aware-topo-model

- The documents aims to define a model for providing the resource-related information of a compute domain, in a per DC PoP basis
  - The goal is to integrate the compute information together with the topological information of the network
- Cloud managers will be the source of the information
  - It is expected to obtain the compute node information via APIs, adapting it to the YANG data models here proposed
- Different cloud solutions impose different ways of modelling the compute resources and assets
  - E.g., OpenStack, Kubernetes
  - The approach should be easily extendable for future evolutions on the cloud side
- Original proposition was part of draft-ietf-teas-sf-aware-topo-model

Models considered

- Cloud technology-specific models
  - Hypervisor-based cloud solutions (e.g., OpenStack)
  - Container-based cloud solutions (e.g., Kubernetes)
- Integration with TE model

```
module: ietf-kubernetes-info
  +--rw dcpop
     +--rw dcpop-id?   string
     +--rw dc* [id]
        +--rw id                    string
        +--rw attachment-circuit
        |  +--rw ac* [ac-id]
        |     +--rw ac-id              string
        |     +--rw node-ref?          string
        |     +--rw interface-name?    string
        |     +--rw bandwidth?         uint64
        |     +--rw status?            enumeration {up, down, degraded}
        +--rw kubernetes
           +--rw system
           |  +--rw system-id   string
           |  +--rw name        string
           |  +--rw location    string
           +--rw cluster
              +--rw cluster-id  string
              +--rw name        string
              +--rw location    string
              +--rw nodes
              |  +--rw node* [name]
              |     +--rw name      string
              |     +--rw cpu
              |     |  +--rw capacity      uint64
              |     |  +--rw allocatable   uint64
              |     |  +--rw usage         uint64
              |     +--rw memory
              |     |  +--rw capacity      uint64
              |     |  +--rw allocatable   uint64
              |     |  +--rw usage         uint64
              |     +--rw workloads
              |        +--rw max           uint32
              |        +--rw running       uint32
              +--rw workloads
                 +--rw workload* [id]
                    +--rw id               string
                    +--rw namespace        string
                    +--rw name             string
                    +--rw cpu
                    |  +--rw request       uint64
                    |  +--rw limit         uint64
                    |  +--rw usage         uint64
                    +--rw memory
                    |  +--rw request       uint64
                    |  +--rw limit         uint64
                    |  +--rw usage         uint64
                    +--rw status
                       +--rw phase         enumeration {pending, running, failed}
                       +--rw conditions*   string
```

```
module: ietf-dc-aware-topology
  augment /nw:networks/nw:network/te:te-topology/te:node:
    +--rw dc-awareness?                          // presence container
       +--rw attachment-circuit* [ac-id]
       |  +--rw ac-id              string
       +--rw (dc-system-ref)?
          +--:(openstack)
          |  +--rw openstack-ref?    -> /os:dcpop/os:dc/os:openstack/os:system/os:system-id
          +--:(kubernetes)
             +--rw kubernetes-ref?   -> /k8s:dcpop/k8s:dc/k8s:kubernetes/k8s:system/k8s:system-id
```

Changes in version -05

- Comments from IETF 122 have been addressed
  - Lou: Comments about the absence of network resources in the YANG model, and how available network resources will be linked
  - Oscar: Suggestion on including links to network resources
  - Italo: Question about if the solution applies only to TE networks or to any network type
- Refinement of the exemplary models provided (Kubernetes and Openstack)
- Addition of a new model for integration with TE model
- Leverage on the notion of Attachment Circuit for clearly identifying the attachment between network and cloud environments

This document proposes a DC-aware extension for the topology model. This model is intended to co-exist and interoperate with existing traffic engineering (TE) topology models (e.g. ietf-te-topology [RFC8795] and technology-specific augmentations). It does not re-define the TE topology, but rather links cloud/DC resource entities to the network via attachment points / circuits, providing a bridge between the DC-aware model and the TE domain.

Although the model is designed to be compatible with TE aware topologies, it can also be applied to non-TE networks. The TE-related leafrefs are optional and only used when integration with TE models is required.

Table of Contents

1. Introduction 2
2. Datacenter information 3
3. Relationship between TE and DC Resource Models 3
4. Model structure 4
4.1. Cloud technology-specific models 4
4.1.1. Hypervisor-based cloud solutions 5
4.1.2. Container-based cloud solutions 7
4.2. Integration with TE model 8
5. Security and operational considerations 8
6. Informative References 8
Acknowledgements 9
Authors' Addresses 9

4. Model structure

This section provides the YANG modules that describe how cloud resources which can be associated with nodes in a TE topology. Three modules are defined:

* ietf-openstack-info: describes OpenStack-based systems.

* ietf-kubernetes-info: describes Kubernetes-based systems.

* ietf-dc-aware-topology: provides the augmentation of the TE topology model to expose attachment circuits from network nodes to the data center systems defined in the previous two modules.

3. Relationship between TE and DC Resource Models

The modules proposed in this document define a data structure that allows a TE or network topology model to associate network nodes and links with data-center (DC) computing resources. The intent is to expose the awareness of available compute capabilities within the network view, without embedding full compute-domain models.

The design follows these principles:

* Separation of concerns: the ietf-dc-aware-topology module provides references towards cloud technology-specific models such as ietf-openstack-info and ietf-kubernetes-info. This helps to augment the model as long as the cloud-related technology evolves along the time.

* Attachment circuit abstraction [RFC9834]: Each DC reachable from the network is represented through one or more attachment circuits (ACs). The AC serves as the conceptual interface between the TE node and the computing domain. An attachment circuit may map to a physical port, a virtual link, or a service-level endpoint. This document adopts the AC semantics defined in [RFC9834].

* Integration with TE topology models: The model augments TE nodes and termination points defined in [RFC8795] (ietf-te-topology) with optional containers that reference DC resources connected through an attachment circuit. This allows a TE-aware orchestrator to take into account computing capacity, storage, or service location when performing path computation or placement decisions.

* Generic applicability: Although this module integrates with TE models, it is applicable to non-TE topologies as well. The augmentation is optional and can be ignored when TE-awareness is not required.

Next steps

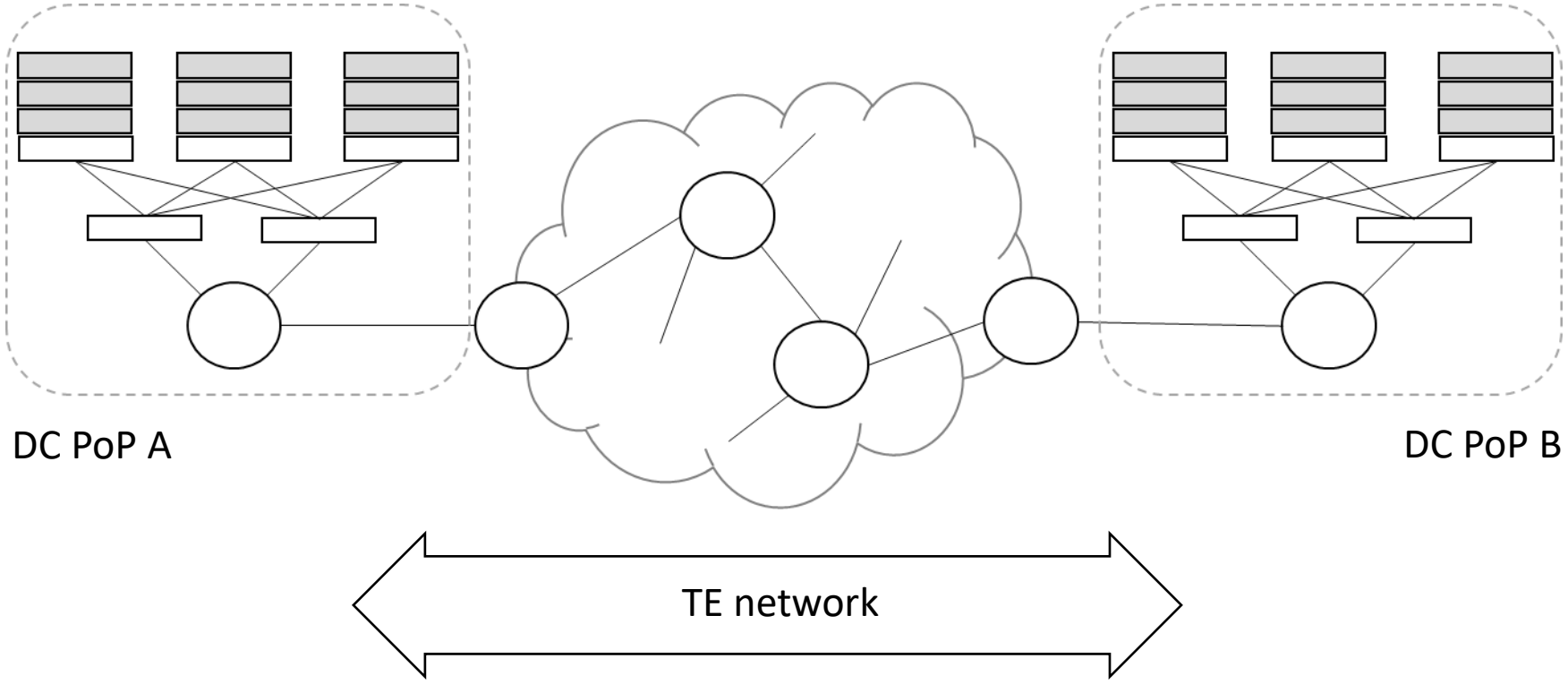
- Check if in this way the draft is perceived as subject for TEAS
- Collect new comments / suggestions
- Refine the models

- Prepare new version and keep progressing the document
  - Final aim is to get a better integration between network and cloud environments for the provision of guaranteed services

Backup

Problem statement

- Wide deployment of computing facilities across service provider's Networks, in the form of DC PoPs (as edge and/or central cloud)
- Interesting **to have joint topological view of both networking and computing resources** available to assist on **TE decisions that could require combined awareness of network and compute domains**
- Similar approach as the one followed in *draft-ietf-teas-sf-aware-topo-model* but concentrated on available DC resources instead of functions



- DC PoPs described in terms of resource capabilities such as CPU, memory, storage, etc
- Alternatively, they could be described in terms of resource bundles (quotas, flavors)

Flavor	vCPU	RAM	Storage	Bandwidth
.tiny	1	512 MB	1 GB	1 Gbps
.small	1	2 GB	20 GB	1 Gbps
.medium	2	4 GB	40 GB	1 Gbps
.large	4	8 GB	80 GB	1 Gbps
.2xlarge	8	16 GB	160 GB	1 Gbps
.4xlarge	16	32 GB	320 GB	1 Gbps
.8xlarge	32	64 GB	640 GB	1 Gbps

Need for the model

- Means for an optimal orchestration in the telecom-cloud network operation
- Facilitator of interoperability through proper resource abstraction
- Enabler seamless integration of cloud and network services
- Allows the exposure of combined network-cloud information
- Permits a more efficient dynamic resource allocation cross-domain (i.e., for cloud and network)