Application-oriented Stateful PCE Architecture and Use-cases for Transport Networks

<draft-lee-pce-app-oriented-arch-00.txt>

Young Lee, Xian Zhang, Haomian Zhang, Dhruv Dhody (Huawei), Guoying Zhang (CATR), Oscar Gonzalez de Dios (Telefonica)

PCE WG
IETF 88 Vancouver
Background & Motivation

• Can PCE support open programmable interfaces that it might support SDN network virtualization for transport networks?
• Currently, it is out of scope.
• Related work:
SDN concept has been applied for transport networks.

- Separation of control plane functions from data planes by GMPLS/ASON control plane technology
  - Link Discovery (LMP)
  - Dissemination of Link/Resource Information (OSPF-TE)
  - Connection/Provisioning (RSVP-TE)
- Global view of a network
  - TEDB, LSDB give the global domain view of a network
- Logically centralized control
  - PCE for path computation; Stateful PCE for initiation of path provisioning (in cooperation with GMPLS signaling)

Can PCE architecture support network virtualization?
Client Control

• Supports various applications via various NB APIs (e.g., OpenStack, etc.)
• Various types of client to network
  – Data Center Operators
  – Virtual Network Providers
  – Contents Providers
  – Carriers of carrier
• Primary source for application service/connectivity requirements and location information (client end points).

But current GMPLS/PCE architecture does not support programmable interfaces for network virtualization
Virtual Network Control Layer

- Virtual Network Control separated from Physical network control
  - Open interfaces creation
  - Third party developer can develop VNC layer
- Virtual Network Control Layer provides virtual network control functions:
  - Virtual Service Creation
  - Virtual Path Computation
  - Virtual Topology Database Creation
  - Virtual Network Discovery
  - Topology Abstraction for Virtual Service
  - Virtual connection setup
Application-oriented Stateful PCE Architecture

------------------------------------------
<table>
<thead>
<tr>
<th>Application Stratum</th>
</tr>
</thead>
<tbody>
<tr>
<td>/\       /\       /\       North Bound API</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>|/</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>|/     |/     |/</td>
</tr>
<tr>
<td>------------------------------------------</td>
</tr>
<tr>
<td>/</td>
</tr>
<tr>
<td>Transport</td>
</tr>
<tr>
<td>Network</td>
</tr>
<tr>
<td>Controller</td>
</tr>
<tr>
<td>(TNC)</td>
</tr>
<tr>
<td>/\</td>
</tr>
<tr>
<td>|/</td>
</tr>
<tr>
<td>/\</td>
</tr>
<tr>
<td>|/</td>
</tr>
<tr>
<td>Transport Stateful PCE</td>
</tr>
<tr>
<td>------------------------</td>
</tr>
<tr>
<td>/\</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>|/</td>
</tr>
</tbody>
</table>

Physical Network Infrastructure
Use-case A: application-specific topology abstraction and virtual control

Client A Controller

Client B Controller

Client C Controller

VNC

PCE

Creates abstraction topology per application/client need

network topology
Use-case: Dynamic DCI in multi-domain network (Topology Request)

1. Topology Request: Endpoints list
2. Topology Request: Endpoints list, peering point
3. Abstracted Topology
4. E2E Abstracted Topology

DC Controller

VNC

PCE 1

PCE 2

PCE 3

DC1

DC2

Network 1

Network 2

Network 3

DC3

DC4

DC5

DC6

IETF 88, Vancouver
Use-case: Dynamic DCI in multi-domain network (Connection Request)

1. Connect Request: 1-6
2. V_Path Compute
3. Connect

DC Controller
VNC
PNC 1
PNC 2
PNC 3
DC 1
DC 2
DC 3
DC 4
DC 5
DC 6
Network 1
Network 2
Network 3

IETF 88, Vancouver
Implementation Alternatives

Focus of PCE protocol extensions

Option A:
PCE interacts with VNC

Option B:
PCE interacts with Client/APP directly
Next Steps

• Extend the charter if WG thinks this is a viable PCE direction.

• Explore a new WG formation if WG thinks this is out of scope.