

PCE Working Group  
Internet-Draft  
Intended status: Informational  
Expires: May 1, 2018

D. Dhody  
Y. Lee  
Huawei Technologies  
D. Ceccarelli  
Ericsson  
October 28, 2017

**Applicability of Path Computation Element (PCE) for Abstraction and  
Control of TE Networks (ACTN)  
draft-ietf-pce-applicability-actn-02**

**Abstract**

Abstraction and Control of TE Networks (ACTN) refers to the set of virtual network operations needed to orchestrate, control and manage large-scale multi-domain TE networks so as to facilitate network programmability, automation, efficient resource sharing, and end-to-end virtual service aware connectivity and network function virtualization services.

The Path Computation Element Communication Protocol (PCEP) provides mechanisms for Path Computation Elements (PCEs) to perform path computations in response to Path Computation Clients (PCCs) requests.

This document examines the applicability of PCE to the ACTN framework.

**Status of This Memo**

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on May 1, 2018.

## Copyright Notice

Copyright (c) 2017 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

<a href="#">1.</a>	Introduction . . . . .	<a href="#">2</a>
<a href="#">1.1.</a>	Path Computation Element (PCE) . . . . .	<a href="#">2</a>
<a href="#">1.1.1.</a>	Role of PCE in SDN . . . . .	<a href="#">3</a>
<a href="#">1.1.2.</a>	PCE in multi-domain and multi-layer deployments . . . . .	<a href="#">4</a>
<a href="#">1.2.</a>	Abstraction and Control of TE Networks (ACTN) . . . . .	<a href="#">4</a>
<a href="#">1.3.</a>	PCE and ACTN . . . . .	<a href="#">6</a>
<a href="#">2.</a>	Architectural Considerations . . . . .	<a href="#">6</a>
<a href="#">2.1.</a>	Multi domain coordination via Hierarchy . . . . .	<a href="#">6</a>
<a href="#">2.2.</a>	Virtualization/Abstraction function . . . . .	<a href="#">7</a>
<a href="#">2.3.</a>	Customer mapping function . . . . .	<a href="#">8</a>
<a href="#">2.4.</a>	Virtual Network Operations . . . . .	<a href="#">8</a>
<a href="#">3.</a>	Interface Considerations . . . . .	<a href="#">9</a>
<a href="#">4.</a>	Realizing ACTN with PCE (and PCEP) . . . . .	<a href="#">10</a>
<a href="#">5.</a>	Relationship to PCE based central control . . . . .	<a href="#">13</a>
<a href="#">6.</a>	IANA Considerations . . . . .	<a href="#">13</a>
<a href="#">7.</a>	Security Considerations . . . . .	<a href="#">13</a>
<a href="#">8.</a>	Acknowledgments . . . . .	<a href="#">13</a>
<a href="#">9.</a>	References . . . . .	<a href="#">13</a>
<a href="#">9.1.</a>	Normative References . . . . .	<a href="#">14</a>
<a href="#">9.2.</a>	Informative References . . . . .	<a href="#">14</a>
	Authors' Addresses . . . . .	<a href="#">17</a>

## [1.](#) Introduction

### [1.1.](#) Path Computation Element (PCE)

The Path Computation Element communication Protocol (PCEP) [[RFC5440](#)] provides mechanisms for Path Computation Elements (PCEs) [[RFC4655](#)] to perform path computations in response to Path Computation Clients (PCCs) requests.



The ability to compute shortest constrained TE LSPs in Multiprotocol Label Switching (MPLS) and Generalized MPLS (GMPLS) networks across multiple domains has been identified as a key motivation for PCE development.

A stateful PCE [[RFC8231](#)] is capable of considering, for the purposes of path computation, not only the network state in terms of links and nodes (referred to as the Traffic Engineering Database or TED) but also the status of active services (previously computed paths, and currently reserved resources, stored in the Label Switched Paths Database (LSPDB).

[RFC8051] describes general considerations for a stateful PCE deployment and examines its applicability and benefits, as well as its challenges and limitations through a number of use cases.

[RFC8231] describes a set of extensions to PCEP to provide stateful control. A stateful PCE has access to not only the information carried by the network's Interior Gateway Protocol (IGP), but also the set of active paths and their reserved resources for its computations. The additional state allows the PCE to compute constrained paths while considering individual LSPs and their interactions. [[I-D.ietf-pce-pce-initiated-lsp](#)] describes the setup, maintenance and teardown of PCE-initiated LSPs under the stateful PCE model.

[RFC8231] also describes the active stateful PCE. The active PCE functionality allows a PCE to reroute an existing LSP or make changes to the attributes of an existing LSP, or a PCC to delegate control of specific LSPs to a new PCE.

#### **[1.1.1. Role of PCE in SDN](#)**

Software-Defined Networking (SDN) refers to a separation between the control elements and the forwarding components so that software running in a centralized system called a controller, can act to program the devices in the network to behave in specific ways. A required element in an SDN architecture is a component that plans how the network resources will be used and how the devices will be programmed. It is possible to view this component as performing specific computations to place flows within the network given knowledge of the availability of network resources, how other forwarding devices are programmed, and the way that other flows are routed. It is concluded in [[RFC7399](#)], that this is the same function that a PCE might offer in a network operated using a dynamic control plane. This is the function and purpose of a PCE, and the way that a PCE integrates into a wider network control system including SDN is presented in Application-Based Network Operation (ABNO) [[RFC7491](#)].



### **1.1.2. PCE in multi-domain and multi-layer deployments**

Computing paths across large multi-domain environments require special computational components and cooperation between entities in different domains capable of complex path computation. The PCE provides an architecture and a set of functional components to address this problem space. A PCE may be used to compute end-to-end paths across multi-domain environments using a per-domain path computation technique [[RFC5152](#)]. The Backward recursive PCE based path computation (BRPC) mechanism [[RFC5441](#)] defines a PCE-based path computation procedure to compute inter-domain constrained MPLS and GMPLS TE networks. However, both per-domain and BRPC techniques assume that the sequence of domains to be crossed from source to destination is known, either fixed by the network operator or obtained by other means.

[RFC6805] describes a Hierarchical PCE (H-PCE) architecture which can be used for computing end-to-end paths for inter-domain MPLS Traffic Engineering (TE) and GMPLS Label Switched Paths (LSPs) when the domain sequence is not known. Within the Hierarchical PCE (H-PCE) architecture, the Parent PCE (P-PCE) is used to compute a multi-domain path based on the domain connectivity information. A Child PCE (C-PCE) may be responsible for a single domain or multiple domains, it is used to compute the intra-domain path based on its domain topology information.

[I-D.ietf-pce-stateful-hpce] state the considerations for stateful PCE(s) in hierarchical PCE architecture. In particular, the behavior changes and additions to the existing stateful PCE mechanisms (including PCE- initiated LSP setup and active PCE usage) in the context of networks using the H-PCE architecture.

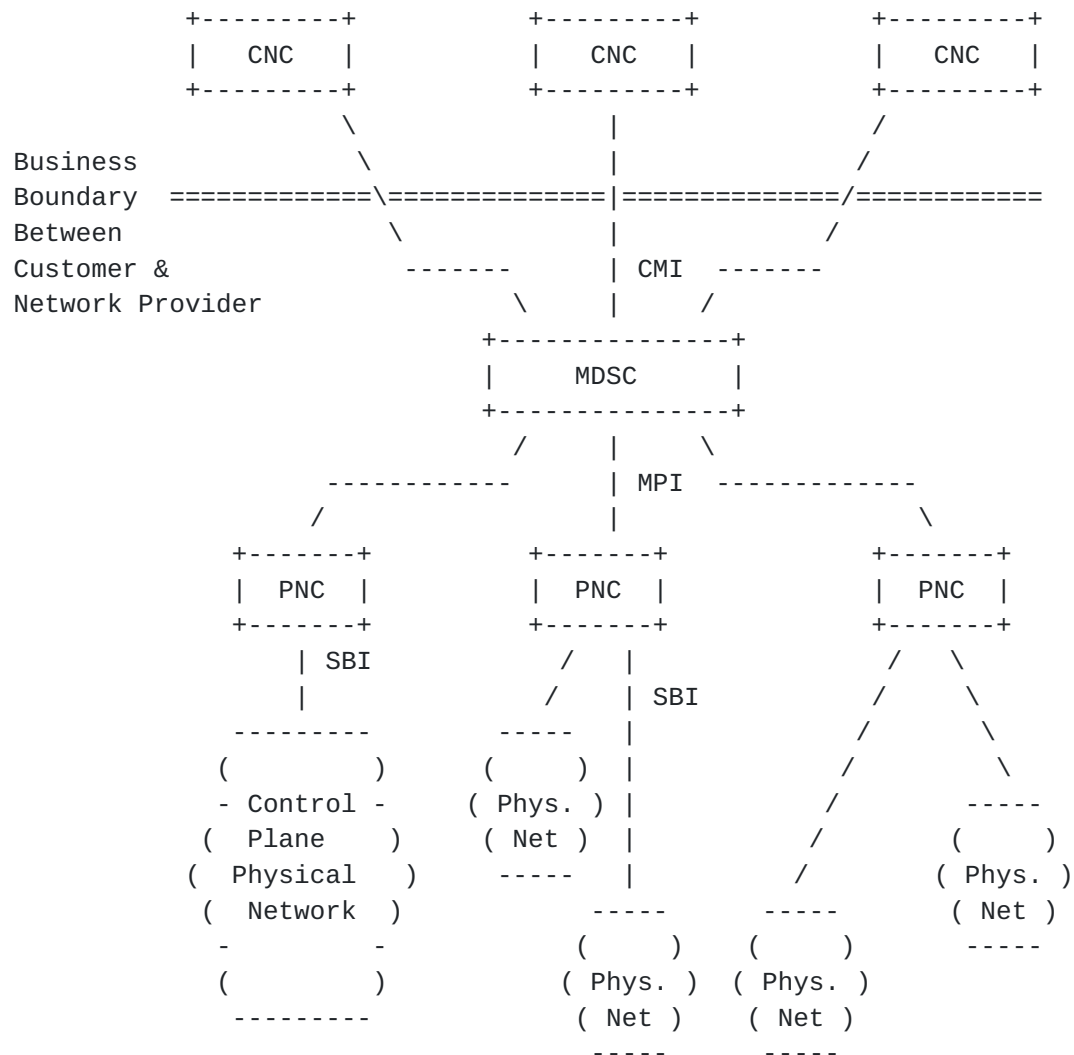
[RFC5623] describes a framework for applying the PCE-based architecture to inter-layer to (G)MPLS TE. It provides suggestions for the deployment of PCE in support of multi-layer networks. It also describes the relationship between PCE and a functional component in charge of the control and management of the VNT, called the Virtual Network Topology Manager (VNTM).

## **1.2. Abstraction and Control of TE Networks (ACTN)**

[I-D.ietf-teas-actn-requirements] describes the high-level ACTN requirements. [[I-D.ietf-teas-actn-framework](#)] describes the architecture model for ACTN including the entities (Customer Network Controller(CNC), Mult-domain Service Coordinator(MDSC), and Provisioning Network Controller (PNC) and their interfaces.



The ACTN reference architecture identified a three-tier control hierarchy as depicted in Figure 1:



CMI - (CNC-MDSC Interface)

MPI - (MDSC-PNC Interface)

Figure 1: ACTN Hierarchy

The two interfaces with respect to the MDSC, one north of the MDSC Interface) and MPI (MDSC-PNC Interface), respectively. A hierarchy of MDSC is possible with a recursive MPI interface.

[I-D.ietf-teas-actn-info-model] provides an information model for ACTN interfaces.





### **1.3. PCE and ACTN**

This document examines the PCE and ACTN architecture and describes how the PCE architecture is applicable to ACTN. It also lists the PCEP extensions that are needed to use PCEP as an ACTN interface. This document also identifies any gaps in PCEP, that exist at the time of publication of this document.

## **2. Architectural Considerations**

ACTN [[I-D.ietf-teas-actn-framework](#)] architecture is based on hierarchy and recursiveness of controllers. It defines three types of controllers (depending on the functionalities they implement). The main functionalities are -

- o Multi domain coordination function
- o Virtualization/Abstraction function
- o Customer mapping/translation function
- o Virtual service coordination function

Section 3 of [[I-D.ietf-teas-actn-framework](#)] describes these functions.

It should be noted that, in this document we list all possible ways in which PCEP could be used for each of the above functions, but all functions are not required to be implemented via PCEP. Operator may choose to use the PCEP for multi domain coordination via stateful H-PCE but use RestConf or BGP-LS to get the topology and support virtualization/abstraction function.

### **2.1. Multi domain coordination via Hierarchy**

With the definition of domain being "everything that is under the control of the single logical controller", as per [[I-D.ietf-teas-actn-framework](#)], it is needed to have a control entity that oversees the specific aspects of the different domains and to build a single abstracted end-to-end network topology in order to coordinate end-to-end path computation and path/service provisioning.

The MDSC in ACTN framework realizes this function by coordinating the per-domain PNCs in a hierarchy of controllers. It also needs to detach from the underlying network technology and express customer concerns by business needs.



[RFC6805] and [[I-D.ietf-pce-stateful-hpce](#)] describes a hierarchy of PCE with Parent PCE coordinating multi-domain path computation function between Child PCE(s). It is easy to see how these principles align, and thus how stateful H-PCE architecture can be used to realize ACTN.

The Per domain stitched LSP in the Hierarchical stateful PCE architecture, described in Section 3.3.1 of [[I-D.ietf-pce-stateful-hpce](#)] is well suited for multi-domain coordination function. This includes domain sequence selection; E2E path computation; Controller (PCE) initiated path setup and reporting. This is also applicable to multi-layer coordination in case of IP+optical networks.

[[I-D.litkowski-pce-state-sync](#)]" describes the procedures to allow a stateful communication between PCEs for various use-cases. The procedures and extensions are also applicable to Child and Parent PCE communication and thus useful for ACTN as well.

## **2.2. Virtualization/Abstraction function**

To realize ACTN, an abstracted view of the underlying network resources needs to be built. This includes global network-wide abstracted topology based on the underlying network resources of each domain. This also include abstract topology created as per the customer service connectivity requests and represented as a network slice allocated to each customer.

In order to compute and provide optimal paths, PCEs require an accurate and timely Traffic Engineering Database (TED). Traditionally this TED has been obtained from a link state (LS) routing protocol supporting traffic engineering extensions. PCE may construct its TED by participating in the IGP ([[RFC3630](#)] and [[RFC5305](#)] for MPLS-TE; [[RFC4203](#)] and [[RFC5307](#)] for GMPLS). An alternative is offered by BGP-LS [[RFC7752](#)].

In case of H-PCE [[RFC6805](#)], the parent PCE needs to build the domain topology map of the child domains and their interconnectivity. [[RFC6805](#)] and [[I-D.ietf-pce-inter-area-as-applicability](#)] suggest that BGP-LS could be used as a "northbound" TE advertisement from the child PCE to the parent PCE.

[[I-D.dhodylee-pce-pcep-ls](#)] proposes another approaches for learning and maintaining the Link-State and TE information as an alternative to IGPs and BGP flooding, using PCEP itself. The child PCE can use this mechanism to transport Link-State and TE information from child PCE to a Parent PCE using PCEP.



In ACTN, there is a need to control the level of abstraction based on the deployment scenario and business relationship between the controllers. The mechanism used to disseminate information from PNC (child PCE) to MDSC (parent PCE) should support abstraction.

[[I-D.lee-teas-actn-abstraction](#)] describes a few alternative approaches of abstraction. The resulting abstracted topology can be encoded using the PCEP-LS mechanisms [[I-D.dhodylee-pce-pcep-ls](#)]. PCEP-LS is an attractive option when the operator would wish to have a single control plane protocol (PCEP) to achieve ACTN functions.

### **2.3. Customer mapping function**

In ACTN, there is a need to map customer virtual network (VN) requirements into network provisioning request to the PNC. That is, the customer requests/commands are mapped into network provisioning requests that can be sent to the PNC. Specifically, it provides mapping and translation of a customer's service request into a set of parameters that are specific to a network type and technology such that network configuration process is made possible.

[[I-D.ietf-pce-pce-initiated-lsp](#)] describes the setup, maintenance and teardown of PCE-initiated LSPs under the stateful PCE model, without the need for local configuration on the PCC, thus allowing for a dynamic network that is centrally controlled and deployed. To instantiate or delete an LSP, the PCE sends the Path Computation LSP Initiate Request (PCInitiate) message to the PCC. As described in [[I-D.ietf-pce-stateful-hpce](#)], for inter-domain LSP in Hierarchical PCE architecture, the initiation operations can be carried out at the parent PCE. In which case after parent PCE finishes the E2E path computation, it can send the PCInitiate message to the child PCE, the child PCE further propagates the initiate request to the LSR. The customer request is received by the MDSC (parent PCE) and based on the business logic, global abstracted topology, network conditions and local policy, the MDSC (parent PCE) translates this into per domain LSP initiation request that a PNC (child PCE) can understand and act on. This can be done via the PCInitiate message.

PCEP extensions for associating opaque policy between PCEP peer [[I-D.ietf-pce-association-policy](#)] can be used.

### **2.4. Virtual Network Operations**

Virtual service coordination function in ACTN incorporates customer service-related information into the virtual network service operations in order to seamlessly operate virtual networks while meeting customer's service requirements.



[I-D.leedhody-pce-vn-association] describes the need for associating a set of LSPs with a VN "construct" to facilitate VN operations in PCE architecture. This association allows the PCEs to identify which LSPs belong to a certain VN.

This association based on VN is useful for various optimizations at the VN level which can be applied to all the LSPs that are part of the VN slice. During path computation, the impact of a path for an LSP is compared against the paths of other LSPs in the VN. This is to make sure that the overall optimization and SLA of the VN rather than of a single LSP. Similarly, during re-optimization, advanced path computation algorithm and optimization technique can be considered for all the LSPs belonging to a VN/customer and optimize them all together.

### 3. Interface Considerations

As per [I-D.ietf-teas-actn-framework], to allow virtualization and multi domain coordination, the network has to provide open, programmable interfaces, in which customer applications can create, replace and modify virtual network resources and services in an interactive, flexible and dynamic fashion while having no impact on other customers. The 3 ACTN interfaces are -

- o The CNC-MDSC Interface (CMI) is an interface between a Customer Network Controller and a Multi Domain Service Coordinator. It requests the creation of the network resources, topology or services for the applications. The MDSC may also report potential network topology availability if queried for current capability from the Customer Network Controller.
- o The MDSC-PNC Interface (MPI) is an interface between a Multi Domain Service Coordinator and a Physical Network Controller. It communicates the creation request, if required, of new connectivity of bandwidth changes in the physical network, via the PNC. In multi-domain environments, the MDSC needs to establish multiple MPIs, one for each PNC, as there are multiple PNCs responsible for its domain control.
- o Incase of hierarchy of MDSC, the MPI is applied recursively. From an abstraction point of view, the top level MDSC which interfaces the CNC operates on a higher level of abstraction (i.e., less granular level) than the lower level MSDCs.

PCEP is especially suitable on the MPI as it meets the requirement and the functions as set out in the ACTN framework [I-D.ietf-teas-actn-framework]. Its recursive nature is well suited

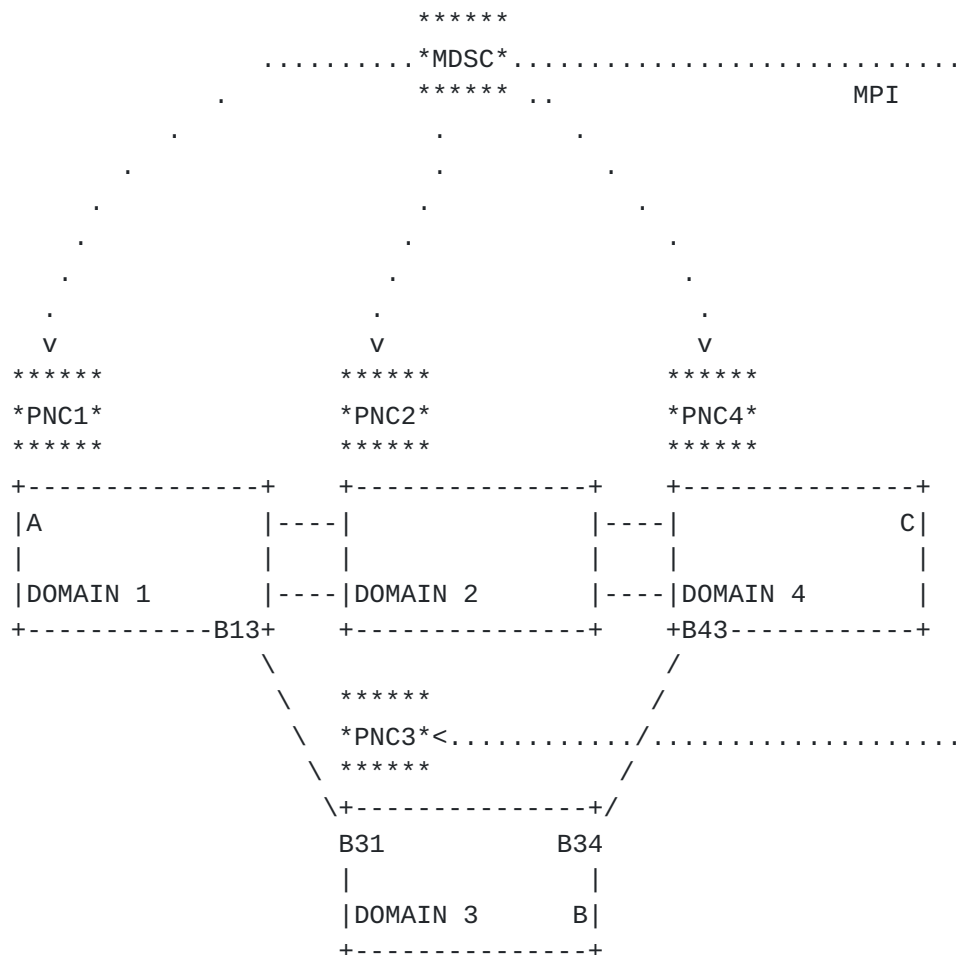




via the multi-level hierarchy of PCE. The [Section 4](#) describe how PCE and PCEP could help realize ACTN.

#### 4. Realizing ACTN with PCE (and PCEP)

As per the example in the Figure 2, there are 4 domains, each with its own PNC and a MDSC at top. The PNC and MDSC need PCE as a important function. The PNC (or child PCE) already uses PCEP to communicate to the network device. It can utilize the PCEP as the MPI to communicate between controllers too.



```
MDSC -> Parent PCE
PNC  -> Child  PCE
MPI  -> PCEP
```

Figure 2: ACTN with PCE

- o Building Domain Topology at MDSC: PNC (or child PCE) needs to have the TED to compute path in its domain. As described in



[Section 2.2](#), it can learn the topology via IGP or BGP-LS. PCEP-LS is also a proposed mechanism to carry link state and traffic engineering information within PCEP. A mechanism to carry abstracted topology while hiding technology specific information between PNC and MDSC is described in [[I-D.dhodylee-pce-pcep-ls](#)]. At the end of this step the MDSC (or parent PCE) has the abstracted topology from each of its PNC (or child PCE). This could be as simple as a domain topology map as described in [[RFC6805](#)] or it can have full topology information of all domains. The latter is not scalable and thus an abstracted topology of each domain interconnected by inter-domain links is the most common case.

- \* Topology Change: When the PNC learns of any topology change, the PNC needs to decide if the change needs to be notified to the MDSC. This is dependent on the level of abstraction between the MDSC and the PNC.
- o VN Instantiate: MDSC is requested to instantiate a VN, the minimal information that is required would be a VN identifier and a set of end points. Various path computation, setup constraints and objective functions may also be provided. In PCE terms, a VN Instantiate can be considered as a set of paths belonging to the same VN. As described in [Section 2.4](#) and [[I-D.leedhody-pce-vn-association](#)] the VN association can help in identifying the set of paths that belong to a VN. The rest of the information like the endpoints, constraints and objective function is already defined in PCEP in terms of a single path.
- \* Path Computation: As per the example in the Figure 2, the VN instantiate requires two end to end paths between (A in Domain 1 to B in Domain 3) and (A in Domain 1 to C in Domain 4). The MDSC (or parent PCE) triggers the end to end path computation for these two paths. MDSC can do path computation based on the abstracted domain topology that it already has or it may use the H-PCE procedures ([Section 2.1](#)) using the PCReq and PCRep messages to get the end to end path with the help of PNC. Either way, the resulted E2E paths may be broken into per-domain paths.
- \* A-B: (A-B13, B13-B31, B31-B)
- \* A-C: (A-B13, B13-B31, B34-B43, B43-C)
- \* Per Domain Path Instantiation: Based on the above path computation, MDSC can issue the path instantiation request to each PNC via PCInitiate message (see [[I-D.ietf-pce-stateful-hpce](#)] and



- [[I-D.leadhody-pce-vn-association](#))). A suitable stitching mechanism would be use to stitch these per domain LSPs. One such mechanism is described in [[I-D.lee-pce-lsp-stitching-hpce](#)], where PCEP is extended to support stitching in stateful H-PCE context.
- \* Per Domain Path Report: Each PNC should report the status of the per-domain LSP to the MDSC via PCRpt message, as per the Hierarchy of stateful PCE ([[I-D.ietf-pce-stateful-hpce](#)]). The status of the end to end LSP (A-B and A-C) is made up when all the per domain LSP are reported up by the PNCs.
  - \* Delegation: It is suggested that the per domain LSPs are delegated to respective PNC, so that they can control the path and attributes based on each domain network conditions.
  - \* State Synchronization: The state needs to be synchronized between the parent PCE and child PCE. The mechanism described in [[I-D.litkowski-pce-state-sync](#)] can be used.
- o VN Modify: MDSC is requested to modify a VN, for example the bandwidth for VN is increased. This may trigger path computation at MDSC as described in the previous step and can trigger an update to existing per-intra-domain path (via PCUpd message) or creation (or deletion) of a per-domain path (via PCInitiate message). This should be done in make-before-break fashion.
  - o VN Delete: MDSC is requested to delete a VN, in this case, based on the E2E paths and the resulting per-domain paths need to be removed (via PCInitiate message).
  - o VN Update (based on network changes): Any change in the per-domain LSP are reported to the MDSC (via PCRpt message) as per [[I-D.ietf-pce-stateful-hpce](#)]. This may result in changes in the E2E path or VN status. This may also trigger a re-optimization leading to a new per-domain path, update to existing path, or deletion of the path.
  - o VN Protection: The VN protection/restoration requirements, need to applied to each E2E path as well as each per domain path. The MDSC needs to play a crucial role in coordinating the right protection/restoration policy across each PNC. The existing protection/restoration mechanism of PCEP can be applied on each path.
  - o In case PNC generates an abstract topology to the MDSC, the PCInitiate/PCUpd messages from the MDSC to a PNC will contain a path with abstract nodes and links. PNC would need to take that



as an input for path computation to get a path with physical nodes and links. Similarly PNC would convert the path received from the device (with physical nodes and links) into abstract path (based on the abstract topology generated before with abstract nodes and links) and reported to the MDSC.

## **5. Relationship to PCE based central control**

[I-D.ietf-teas-pce-central-control] introduces the architecture for PCE as a central controller (PCECC), it further examines the motivations and applicability for PCEP as a southbound interface, and introduces the implications for the protocol. The section 2.1.3 of [I-D.ietf-teas-pce-central-control] describe an hierarchy of PCE-based controller as per the Hierarchy of PCE framework defined in [RFC6805]. Both ACTN and PCECC is based on the same basic framework and thus compatible with each other.

## **6. IANA Considerations**

This is an informational document and thus does not have any IANA allocations to be made.

## **7. Security Considerations**

The ACTN framework described in [I-D.ietf-teas-actn-framework] defines key components and interfaces for managed traffic engineered networks. It also list various security considerations such as request and control of resources, confidentiality of the information, and availability of function which should be taken into consideration.

When PCEP is used on the MPI/MMI, this interface needs to be secured, use of [RFC8253] is RECOMENDED. Each PCEP extension listed in this document, presents its individual security considerations, which continue to apply.

## **8. Acknowledgments**

The authors would like to thank Jonathan Hardwick for the inspiration behind this document. Further thanks to Avantika for her comments with suggested text.

## **9. References**





### **9.1. Normative References**

- [RFC5440] Vasseur, JP., Ed. and JL. Le Roux, Ed., "Path Computation Element (PCE) Communication Protocol (PCEP)", [RFC 5440](#), DOI 10.17487/RFC5440, March 2009, <<https://www.rfc-editor.org/info/rfc5440>>.

### **9.2. Informative References**

- [RFC3630] Katz, D., Kompella, K., and D. Yeung, "Traffic Engineering (TE) Extensions to OSPF Version 2", [RFC 3630](#), DOI 10.17487/RFC3630, September 2003, <<https://www.rfc-editor.org/info/rfc3630>>.
- [RFC4203] Kompella, K., Ed. and Y. Rekhter, Ed., "OSPF Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)", [RFC 4203](#), DOI 10.17487/RFC4203, October 2005, <<https://www.rfc-editor.org/info/rfc4203>>.
- [RFC4655] Farrel, A., Vasseur, J., and J. Ash, "A Path Computation Element (PCE)-Based Architecture", [RFC 4655](#), DOI 10.17487/RFC4655, August 2006, <<https://www.rfc-editor.org/info/rfc4655>>.
- [RFC5152] Vasseur, JP., Ed., Ayyangar, A., Ed., and R. Zhang, "A Per-Domain Path Computation Method for Establishing Inter-Domain Traffic Engineering (TE) Label Switched Paths (LSPs)", [RFC 5152](#), DOI 10.17487/RFC5152, February 2008, <<https://www.rfc-editor.org/info/rfc5152>>.
- [RFC5305] Li, T. and H. Smit, "IS-IS Extensions for Traffic Engineering", [RFC 5305](#), DOI 10.17487/RFC5305, October 2008, <<https://www.rfc-editor.org/info/rfc5305>>.
- [RFC5307] Kompella, K., Ed. and Y. Rekhter, Ed., "IS-IS Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)", [RFC 5307](#), DOI 10.17487/RFC5307, October 2008, <<https://www.rfc-editor.org/info/rfc5307>>.
- [RFC5441] Vasseur, JP., Ed., Zhang, R., Bitar, N., and JL. Le Roux, "A Backward-Recursive PCE-Based Computation (BRPC) Procedure to Compute Shortest Constrained Inter-Domain Traffic Engineering Label Switched Paths", [RFC 5441](#), DOI 10.17487/RFC5441, April 2009, <<https://www.rfc-editor.org/info/rfc5441>>.



- [RFC5623] Oki, E., Takeda, T., Le Roux, J.L., and A. Farrel, "Framework for PCE-Based Inter-Layer MPLS and GMPLS Traffic Engineering", [RFC 5623](#), DOI 10.17487/RFC5623, September 2009, <<https://www.rfc-editor.org/info/rfc5623>>.
- [RFC6805] King, D., Ed. and A. Farrel, Ed., "The Application of the Path Computation Element Architecture to the Determination of a Sequence of Domains in MPLS and GMPLS", [RFC 6805](#), DOI 10.17487/RFC6805, November 2012, <<https://www.rfc-editor.org/info/rfc6805>>.
- [RFC7399] Farrel, A. and D. King, "Unanswered Questions in the Path Computation Element Architecture", [RFC 7399](#), DOI 10.17487/RFC7399, October 2014, <<https://www.rfc-editor.org/info/rfc7399>>.
- [RFC7491] King, D. and A. Farrel, "A PCE-Based Architecture for Application-Based Network Operations", [RFC 7491](#), DOI 10.17487/RFC7491, March 2015, <<https://www.rfc-editor.org/info/rfc7491>>.
- [RFC7752] Gredler, H., Ed., Medved, J., Previdi, S., Farrel, A., and S. Ray, "North-Bound Distribution of Link-State and Traffic Engineering (TE) Information Using BGP", [RFC 7752](#), DOI 10.17487/RFC7752, March 2016, <<https://www.rfc-editor.org/info/rfc7752>>.
- [RFC8051] Zhang, X., Ed. and I. Minei, Ed., "Applicability of a Stateful Path Computation Element (PCE)", [RFC 8051](#), DOI 10.17487/RFC8051, January 2017, <<https://www.rfc-editor.org/info/rfc8051>>.
- [RFC8231] Crabbe, E., Minei, I., Medved, J., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE", [RFC 8231](#), DOI 10.17487/RFC8231, September 2017, <<https://www.rfc-editor.org/info/rfc8231>>.
- [RFC8253] Lopez, D., Gonzalez de Dios, O., Wu, Q., and D. Dhody, "PCEPS: Usage of TLS to Provide a Secure Transport for the Path Computation Element Communication Protocol (PCEP)", [RFC 8253](#), DOI 10.17487/RFC8253, October 2017, <<https://www.rfc-editor.org/info/rfc8253>>.



[I-D.ietf-pce-pce-initiated-lsp]

Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "PCEP Extensions for PCE-initiated LSP Setup in a Stateful PCE Model", [draft-ietf-pce-pce-initiated-lsp-11](#) (work in progress), October 2017.

[I-D.ietf-pce-stateful-hpce]

Dhody, D., Lee, Y., Ceccarelli, D., Shin, J., King, D., and O. Dios, "Hierarchical Stateful Path Computation Element (PCE).", [draft-ietf-pce-stateful-hpce-01](#) (work in progress), June 2017.

[I-D.ietf-teas-pce-central-control]

Farrel, A., Zhao, Q., Li, Z., and C. Zhou, "An Architecture for Use of PCE and PCEP in a Network with Central Control", [draft-ietf-teas-pce-central-control-05](#) (work in progress), September 2017.

[I-D.ietf-teas-actn-requirements]

Lee, Y., Dhody, D., Belotti, S., Pithewan, K., Ceccarelli, D., Miyasaka, T., Shin, J., and K. Lee, "Requirements for Abstraction and Control of TE Networks", [draft-ietf-teas-actn-requirements-07](#) (work in progress), October 2017.

[I-D.ietf-teas-actn-framework]

Ceccarelli, D. and Y. Lee, "Framework for Abstraction and Control of Traffic Engineered Networks", [draft-ietf-teas-actn-framework-11](#) (work in progress), October 2017.

[I-D.ietf-teas-actn-info-model]

Lee, Y., Belotti, S., Dhody, D., Ceccarelli, D., and B. Yoon, "Information Model for Abstraction and Control of TE Networks (ACTN)", [draft-ietf-teas-actn-info-model-04](#) (work in progress), October 2017.

[I-D.ietf-pce-inter-area-as-applicability]

King, D., Meuric, J., Dugeon, O., Zhao, Q., Dhody, D., and O. Dios, "Applicability of the Path Computation Element to Inter-Area and Inter-AS MPLS and GMPLS Traffic Engineering", [draft-ietf-pce-inter-area-as-applicability-06](#) (work in progress), July 2016.

[I-D.dhodylee-pce-pcep-ls]

Dhody, D., Lee, Y., and D. Ceccarelli, "PCEP Extension for Distribution of Link-State and TE Information.", [draft-dhodylee-pce-pcep-ls-08](#) (work in progress), June 2017.



[I-D.leedhody-pce-vn-association]

Lee, Y., Dhody, D., Zhang, X., and D. Ceccarelli, "PCEP Extensions for Establishing Relationships Between Sets of LSPs and Virtual Networks", [draft-leedhody-pce-vn-association-03](#) (work in progress), September 2017.

[I-D.litkowski-pce-state-sync]

Litkowski, S., Sivabalan, S., and D. Dhody, "Inter Stateful Path Computation Element communication procedures", [draft-litkowski-pce-state-sync-02](#) (work in progress), August 2017.

[I-D.ietf-pce-association-policy]

Dhody, D., Sivabalan, S., Litkowski, S., Tantsura, J., and J. Hardwick, "Path Computation Element communication Protocol extension for associating Policies and LSPs", [draft-ietf-pce-association-policy-01](#) (work in progress), June 2017.

[I-D.lee-teas-actn-abstraction]

Lee, Y., Dhody, D., Ceccarelli, D., and O. Dios, "Abstraction and Control of TE Networks (ACTN) Abstraction Methods", [draft-lee-teas-actn-abstraction-02](#) (work in progress), June 2017.

[I-D.lee-pce-lsp-stitching-hpce]

Lee, Y., Dhody, D., and D. Ceccarelli, "PCEP Extensions for Stitching LSPs in Hierarchical Stateful PCE Model", [draft-lee-pce-lsp-stitching-hpce-00](#) (work in progress), June 2017.

Authors' Addresses

Dhruv Dhody  
Huawei Technologies  
Divyashree Techno Park, Whitefield  
Bangalore, Karnataka 560066  
India

EMail: [dhruv.ietf@gmail.com](mailto:dhruv.ietf@gmail.com)





Young Lee  
Huawei Technologies  
5340 Legacy Drive, Building 3  
Plano, TX 75023  
USA

E-Mail: [leeyoung@huawei.com](mailto:leeyoung@huawei.com)

Daniele Ceccarelli  
Ericsson  
Torshamnsgatan, 48  
Stockholm  
Sweden

E-Mail: [daniele.ceccarelli@ericsson.com](mailto:daniele.ceccarelli@ericsson.com)

