**An Architecture for Use of PCE and PCEP in a Network with Central Control**
**draft-ietf-teas-pce-central-control-00**

Abstract

The Path Computation Element (PCE) has become established as a core
component of Software Defined Networking (SDN) systems.  It can
compute optimal paths for traffic across a network for any definition
of "optimal" and can also monitor changes in resource availability
and traffic demands to update the paths.

Conventionally, the PCE has been used to derive paths for MPLS Label
Switched Paths (LSPs).  These paths are supplied using the Path
Computation Element Communication Protocol (PCEP) to the head end of
the LSP for signaling in the MPLS network.

SDN has a far broader applicability than just signaled MPLS traffic
engineered networks, and the PCE may be used to determine paths in a
wide range of use cases including static LSPs, segment routing,
service function chaining (SFC), and indeed any form of routed or
switched network.  It is, therefore reasonable to consider PCEP as a
general southbound control protocol for use in these environments to
allow the PCE to be fully enabled as a central controller.

This document briefly introduces the architecture for PCE as a
central controller, examines the motivations and applicability for
PCEP as a southbound interface, and introduces the implications for
the protocol.  This document does not describe the use cases in
detail and does not define protocol extensions: that work is left for
other documents.

Status of This Memo

Table of Contents

## 1.  Introduction

The Path Computation Element (PCE) [RFC4655] was developed to offload
path computation function from routers in an MPLS traffic engineered
network.  Since then, the role and function of the PCE has grown to
cover a number of other uses (such as GMPLS [RFC7025]) and to allow
delegated control [I-D.ietf-pce-stateful-pce] and PCE-initiated use
of network resources [I-D.ietf-pce-pce-initiated-lsp].

According to [RFC7399], 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.  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 [RFC7491].

In early PCE implementations, where the PCE was used to derive paths
for MPLS Label Switched Paths (LSPs), paths were requested by network
elements and the results of the path computations were supplied to
network elements using the Path Computation Element Communication
Protocol (PCEP) [RFC5440].  This protocol was later extended to allow
a PCE to send unsolicited requests to the network for LSP
establishment [I-D.ietf-pce-pce-initiated-lsp].

SDN has a far broader applicability than just signaled MPLS or GMPLS
traffic engineered networks.  The PCE component in an SDN system may
be used to determine paths in a wide range of use cases including
static LSPs, segment routing [I-D.ietf-spring-segment-routing],
service function chaining (SFC) [RFC7665], and indeed any form of
routed or switched network.  It is, therefore reasonable to consider
PCEP as a general southbound control protocol for use in these

environments to allow the PCE to be fully enabled as a central
controller.

This document introduces the architecture for PCE as a central
controller, examines the motivations and applicability for PCEP as a
southbound interface, and introduces the implications for the
protocol.  This document dos not describe the use cases in detail and
does not define protocol extensions: that work is left for other
documents.

## 2.  Architecture

The architecture for the use of PCE within centralized control of a
network is based on the understanding that a PCE can determine how
connections should be placed and how resources should be used within
the network, and that the PCE can then cause those connections to be
established.  Figure 1 shows how this control relationship works in a
network with an active control plane.  This is a familiar view for
those who have read and understood [RFC4655] and
[I-D.ietf-pce-pce-initiated-lsp].

In this mode of operation, the central controller is asked to create
connectivity by a network orchestrator, a service manager, an
Operations Support System (OSS), a Network Management Station (NMS),
or some other application.  The PCE-based controller computes paths
with awareness of the network topology, the available resources, and
the other services supported in the network.  This information is
held in the Traffic Engineering Database (TED) and other databases
available to the PCE.  Then the PCE sends a request using PCEP to one
of the Network Elements (NEs), and that NE uses a control plane to
establish the requested connections and reserve the network
resources.

```
                ---------------------------------------------
                | Orchestrator / Service Manager / OSS / NMS |
                ---------------------------------------------
                        ^
                        |
                        v
                  -----------
                 |           |     -----
                 | PCE-based |<---| TED |
                 | Controller |     -----
                 |           |
                  -----------
                    ^
                 PCEP|
                    v
                  ----           ----        ----         ----
                 | NE |<------->| NE |<--->| NE |<--->| NE |
                  ----  Control  ----        ----         ----
                        Plane
```
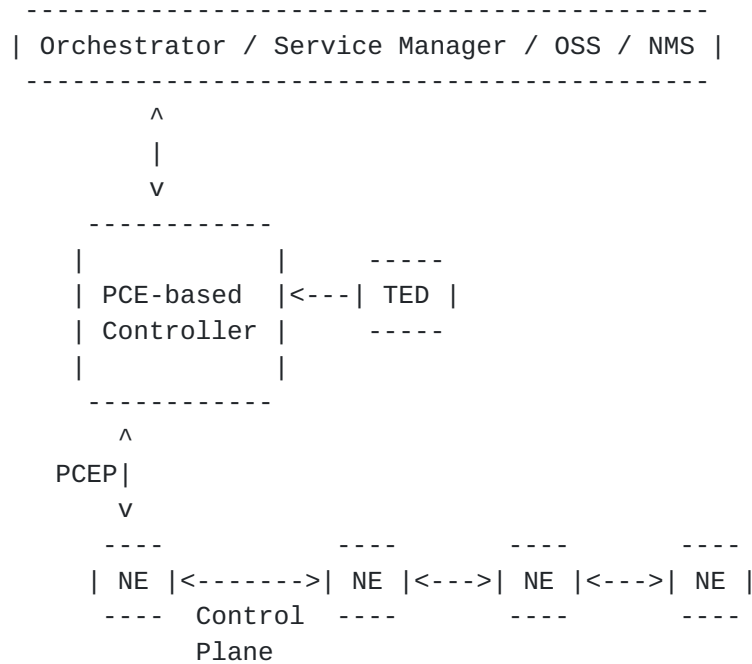
          Figure 1: Architecture for Central Controller with Control Plane

   Although the architecture shown in Figure 1 represents a form of SDN,
   one objective of SDN in some environments is to remove the dependency
   on a control plane.  A transition architecture toward this goal is
   presented in [RFC7491] and is shown in Figure 2.  In this case,
   services are still requested in the same way, and the PCE-based
   controller still requests use of the network using PCEP.  The main
   difference is that the consumer of the PCEP messages is a Network
   Controller that provisions the resources and instructs the data plane
   using Southbound Interface (SBI) that provides an interface to each
   NE.

```
                -----------------------------------------------
                | Orchestrator / Service Manager / OSS / NMS |
                -----------------------------------------------
                                 ^
                                 |
                                 v
                         ------------
                        |            |    -----
                        | PCE-based  |<---| TED |
                        | Controller |    -----
                        |            |
                         ------------
                             ^
                             | PCEP
                             v
                         ------------
                        |  Network   |
                        | Controller |
                        /------------\
                 SBI /    ^        ^   \
                    /     |        |    \
                   /      v        v     \
              ----/     ----     ----    \----
             | NE |    | NE |   | NE |   | NE |
              ----      ----     ----     ----
```
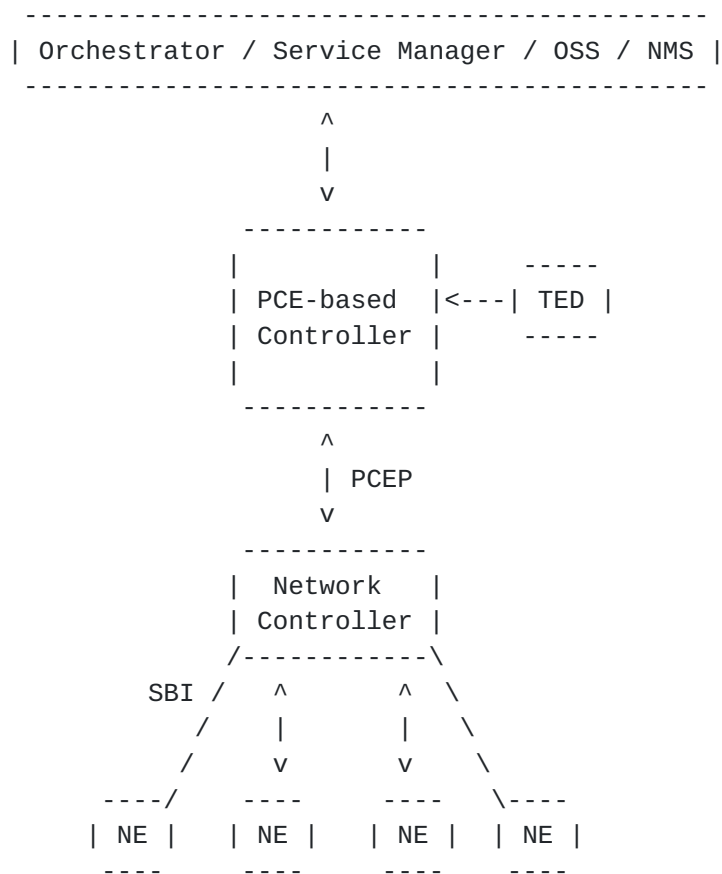
                Figure 2: Architecture Including a Network Controller

   The approach in Figure 2 delivers the SDN functionality but is overly
   complicated and insufficiently flexible.

   o  The complication is created by the use of two controllers in a
      hierarchical organization, and the resultant use of two protocols
      in a southbound direction.

   o  The lack of flexibility arises from the assumed or required lack
      of a control plane.

   This document describes an architecture that reduces the number of
   components and is flexible to a number of deployment models and use
   cases.  In this hybrid approach (shown in Figure 3) the network
   controller is PCE-enabled and can also speak PCEP as the SBI (i.e.,
   it can communicate with each node along the path using PCEP).  That
   means that the controller can communicate with a conventional control
   plane-enabled NE using PCEP and can also use the same protocol to
   program individual NEs.  In this way the PCE-based controller can

control a wider range of networks and deliver many different
functions as described in Section 3.

PCEP is essentially already capable of acting as an SBI and only
small, use case- specific modifications to the protocol are needed to
support this architecture.  The implications for the protocol are
discussed further in Section 4.

```
                  --------------------------------------------
                 | Orchestrator / Service Manager / OSS / NMS |
                  --------------------------------------------
                                       ^
                                       |
                                       v
                              -----------
                             |           |     -----
                             | PCE-based |<---| TED |
                             | Controller |     -----
                             |           |
                             /------------\
                       PCEP /    ^        ^  \
                           /     |        |   \
                          /      v        v    \
                         /     ----      ----   \
                        /     | NE |    | NE |    \
                   ----/      ----      ----      \----
                  | NE |                          | NE |
                   ----                            ----
                     ^        ----      ----        ^
                     :......>| NE |...| NE |<....:
                 Control Plane ----      ----
```

Figure 3: Architecture for Node-by-Node Central Control

## 2.1.  Resilience and Scaling

Systems with central controllers are vulnerable to two problems:
failure or overload of the single controller.  These concerns are not
unique to the use of a PCE-based controller but need to be addressed
in this document before the PCE-based controller architecture can be
considered for use in all but the smallest networks.

There are three architectural mechanisms that can be applied to
address these issues.  The mechanisms are described separately for
clarity, but a deployment use may any combination of the approaches.

For simplicity of illustration, these three approaches are shown in
the sections that follow without a control plane.  However, the
general, hybrid approach of Figure 3 is applicable in each case.

### 2.1.1.  Partitioned Network

The first and simplest approach to handling controller overload or
scalability is to use multiple controllers, each responsible for a
part of the network.  We can call the resultant areas of control
"domains."

This approach is shown in Figure 4.  It can clearly address some of
the scaling and overload concerns since each controller now only has
responsibility for a subset of the network elements.  But this comes
at a cost because end-to-end connections require coordination between
the controllers.  Furthermore, this technique does not remove the
single-point-of-failure concern even if it does reduce the impact on
the network of the failure of a single controller.

Note that PCEP is designed to work as a PCE-to-PCE protocol as well
as a PCE-to-PCC protocol, so it should be possible to use it to
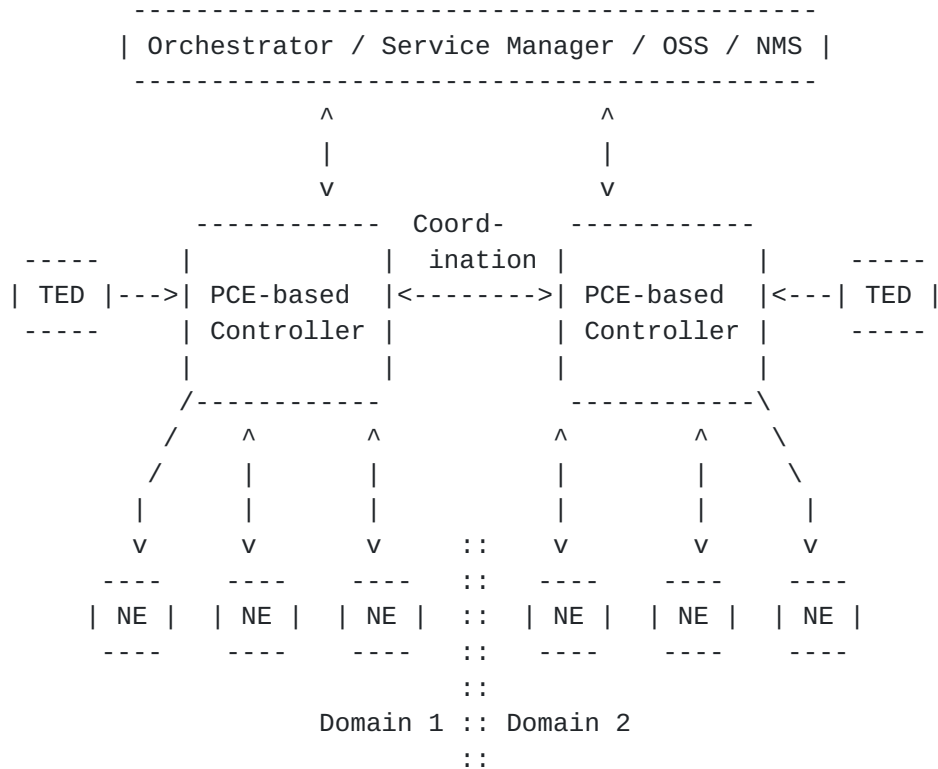coordinate between PCE-based controllers in this model.

```
                 ------------------------------------------------
                 | Orchestrator / Service Manager / OSS / NMS |
                 ------------------------------------------------
                              ^                        ^
                              |                        |
                              v                        v
                       ------------  Coord-    ------------
          -----        |          |  ination |          |        -----
         | TED |--->|  PCE-based |<-------->|  PCE-based |<---| TED |
          -----        | Controller |          | Controller |        -----
                       |          |   |        |   |          |
                     /------------            ------------\
                    /     ^        ^          ^        ^     \
                   /      |        |          |        |      \
                   |      |        |          |        |      |
                   v      v        v    ::    v        v      v
                  ----   ----     ----   ::  ----     ----   ----
                 | NE | | NE |   | NE |  ::  | NE |   | NE |  | NE |
                  ----   ----     ----   ::  ----     ----   ----
                                         ::
                               Domain 1 :: Domain 2
                                         ::
```

          Figure 4: Multiple Controllers on a Partitioned Network

## 2.1.2.  Multiple Parallel Controllers

   Multiple parallel controllers may be deployed as shown in Figure 5.
   Each controller is capable of controlling all of the network elements
   thus the failure of any one controller will not leave the network
   unmanageable and, in normal circumstances, the load can be
   distributed across the controllers.

   To achieve full redundancy and to be able to continue to provide full
   function in the event of the failure a controller, the controllers
   must synchronize with each other.  This is nominally a simple task if
   there are just two controllers, but can actually be quite complex if
   state changes in the network are not to be lost.  Furthermore, if
   there are more than two controllers, the synchronization between
   controllers can become a hard problem.

   Synchronization issues are often off-loaded as "database
   synchronization" problems because distributed database packages have
   already had to address these challenges.  In networking the problem
   may also be addressed by collecting the state from the network
   (effectively using the network as a database) using normal routing
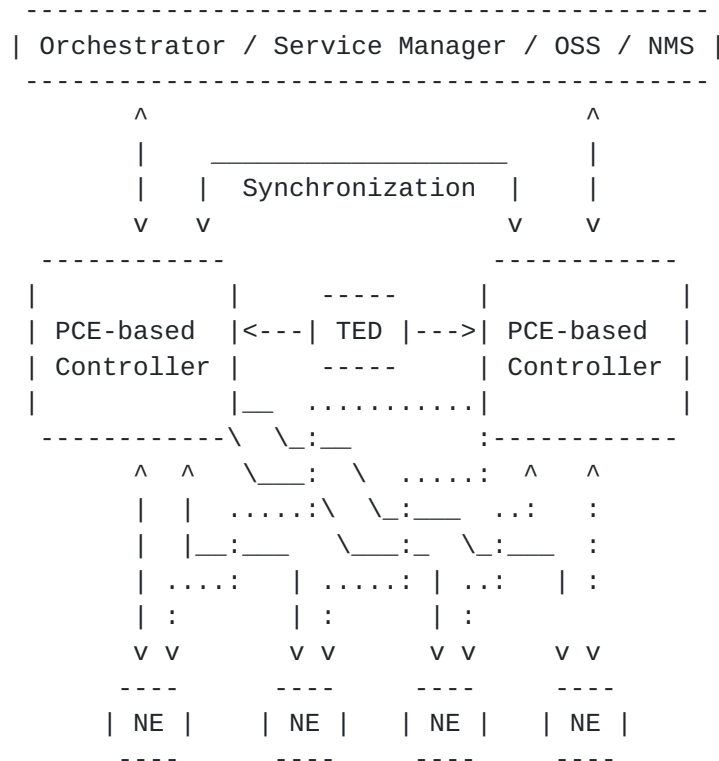   protocols such as OSPF, IS-IS, and BGP.

```
                    ---------------------------------------------
                    | Orchestrator / Service Manager / OSS / NMS |
                    ---------------------------------------------
                           ^                              ^
                           |    _____       |
                           |   |  Synchronization |       |
                           v   v                  v       v
                    -----------              -----------
                    |         |   -----      |         |
                    | PCE-based |<---| TED |--->| PCE-based  |
                    | Controller |    -----      | Controller |
                    |         |__  ..........|         |
                    -----------\  \_:__        :-----------
                       ^   ^   \___:  \  .....:  ^    ^
                       |   |  .....:\  \_:___   ..:   :
                       |   |__:___   \___:_  \_:___  :
                       | ....:   | .....: | ..:    | :
                       | :       | :      | :
                       v v       v v      v v       v v
                       ----      ----     ----      ----
                      | NE |    | NE |   | NE |    | NE |
                       ----      ----     ----      ----
```

Figure 5: Multiple Redundant Controllers

## 2.1.3.  Hierarchical Controllers

   Figure 6 shows an approach with hierarchical controllers.  This
   approach was developed for PCEs in [RFC6805] and appears in various
   SDN architectures where a "parent PCE", an "orchestrator", or "super
   controller" takes responsibility for a high-level view of the network
   before distributing tasks to lower level PCEs or controllers.

   On its own, this approach does little to protect against the failure
   of a controller, but it can make significant improvements in loading
   and scaling of the individual controllers.  It also offers a good way
   to support end-to-end connectivity across multiple administrative or
   technology-specific domains.

   Note that this model can be arbitrarily recursive with one PCE-based
   controller acting as the parent of of another set of PCE-based
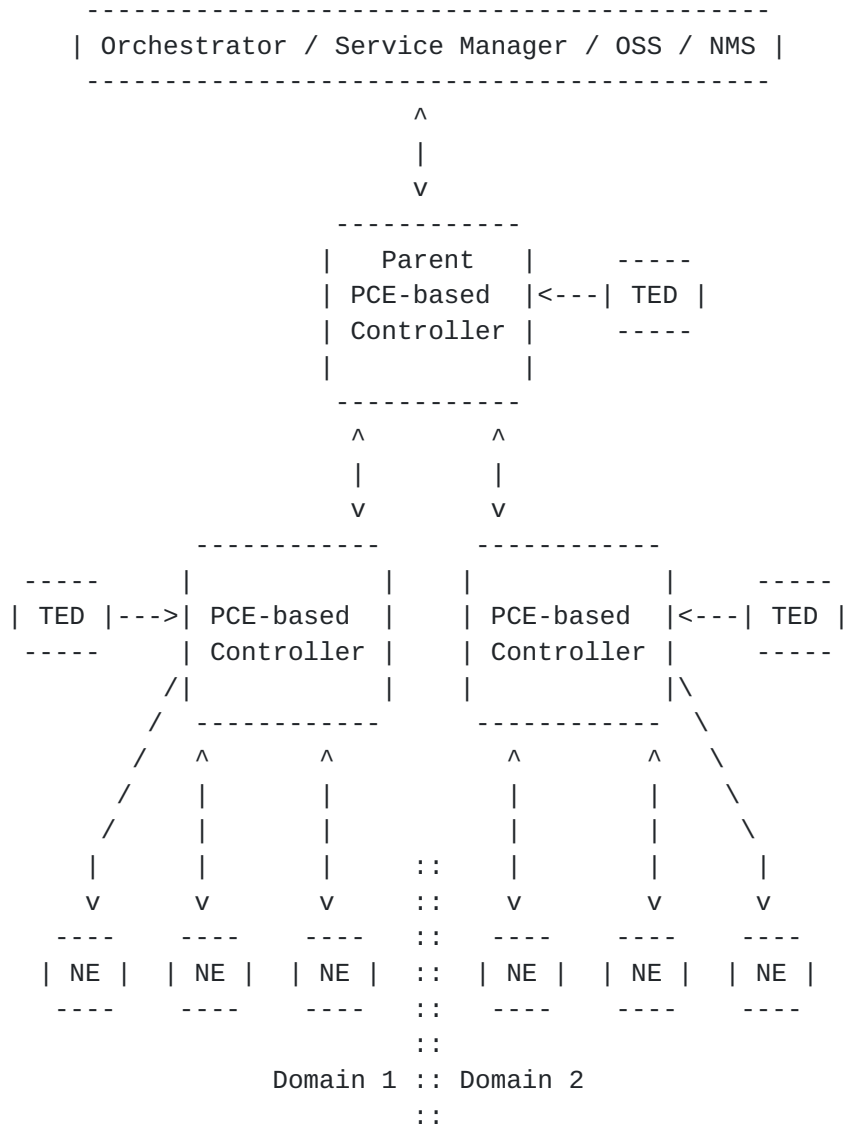   controllers.

```
         ---------------------------------------------
        | Orchestrator / Service Manager / OSS / NMS |
         ---------------------------------------------
                              ^
                              |
                              v
                         -----------
                        |   Parent  |     -----
                        | PCE-based |<---| TED |
                        | Controller |    -----
                        |           |
                         -----------
                          ^         ^
                          |         |
                          v         v
                     -----------    -----------
       -----        |           |  |           |         -----
      | TED |--->| PCE-based  |  | PCE-based  |<---| TED |
       -----        | Controller |  | Controller |      -----
              /|           |  |           |\
             /  -----------    -----------  \
            /  ^       ^          ^       ^   \
           /   |       |          |       |    \
          /    |       |          |       |     \
         |     |       |    ::    |       |      |
         v     v       v    ::    v       v      v
        ----  ----   ----   ::   ----   ----   ----
       | NE | | NE | | NE |  ::  | NE | | NE | | NE |
        ----  ----   ----   ::   ----   ----   ----
                            ::
                  Domain 1 :: Domain 2
                            ::
```

Figure 6: Hierarchical Controllers

## 3. Applicability

   This section gives a very high-level introduction to the
   applicability of a PCE-based centralized controller.  There is no
   attempt to explain each use case in detail, and the inclusion of a
   use case is not intended to suggest that deploying a PCE-based
   controller is a mandatory or recommended approach.  The sections
   below are provided as a stimulus to discussion of the applicability
   of a PCE-based controller and it is expected that separate documents
   will be written to develop the use cases in which there is interest
   for implementation and deployment.  As described in Section 4
   specific enhancements to PCEP may be needed for some of these use

cases and it is expected that the documents that develop each use
case will also address any extensions to PCEP.

The rest of this section is divided into two sub-sections.  The first
approaches the question of applicability from a consideration of the
network technology.  The second looks at the high-level functions
that can be delivered by using a PCE-based controller.

As previously mentioned, this section is intended to just make
suggestions.  Thus the material supplied is very brief.  The omission
of a use case is in no way meant to imply some limit on the
applicability of PCE-based control.

## 3.1.  Technology-Oriented Applicability

This section provides a list of use cases based on network
technology.

### 3.1.1.  Applicability to Control Plane Operated Networks

This mode of operation is the common approach for an active, stateful
PCE to control a traffic engineered MPLS or GMPLS network
[I-D.ietf-pce-stateful-pce].  Note that the PCE-based controller
determines what LSPs are needed and where to place them.  PCEP is
used to instruct the head end of each LSP, and the head end signals
in the control plane to set up the LSP.

### 3.1.2.  Static LSPs in MPLS

Static LSPs are provisioned without the use of a control plane.  This
means that they are established using management plane or "manual"
configuration.

Static LSPs can be provisioned as 1-hop, micro-LSPs at each node
along the path of an end-to-end path LSP.  Each router along the path
must be told what label forwarding instructions to program and what
resources to reserve.  The PCE-based controller keeps a view of the
network and determines the paths of the end-to-end LSPs just as it
does for the use case described in Section 3.1.1, but the controller
uses PCEP to communicate with each router along the path of the end-
to-end LSP.  In this case the PCE-based controller will take
responsibility for managing some part of the MPLS label space for
each of the routers that it controls, and may taker wider
responsibility for partitioning the label space for each router and
allocating different parts for different uses communicating the
ranges to the router using PCEP.

### 3.1.3.  MPLS Multicast

   Multicast LSPs may be provisioned with a control plane or as static
   LSPs.  No extra considerations apply above those in Section 3.1.1 and
   Section 3.1.2 except, of course, to note that the PCE must also
   include the instructions about where the LSP branches, i.e., where
   packets must be copied.

### 3.1.4.  Transport SDN

   Transport SDN (T-SDN) is the application of SDN techniques to
   transport networks.  In this respect a transport network is a network
   built from any technology below the IP layer and designed to carry
   traffic transparently in a connection-oriented way.  Thus, an MPLS
   traffic engineering network is a transport network although it is
   more common to consider technologies such as Time Division
   Multiplexing (TDM) and Optical Transport Networks (OTN).

   Transport networks may be operated with or without a control plane
   and may have point-to-point or point-to-multipoint connections.
   Thus, all of the considerations in Section 3.1.1, Section 3.1.2, and
   Section 3.1.3 apply.  It may be the case that additional technology-
   specific parameters are needed to configure the NEs and these
   parameters will need to be carried in the PCEP messages.

### 3.1.5.  Segment Routing

   Segment routing is described in [I-D.ietf-spring-segment-routing].
   It relies on a series of forwarding instructions being placed in the
   header or a packet: at each hop in the network a router looks at the
   first instruction and may continue to forward the packet unchanged,
   strip the top instruction and forward the packet, or strip the top
   instruction, insert some additional instructions, and forward the
   packet.

   The segment routing architecture supports operations that can be used
   to steer packet flows in a network thus providing a form of traffic
   engineering.  A PCE-based controller can be responsible for computing
   the paths for packet flows in a segment routing network, for
   configuring the forwarding actions on the routers, and for telling
   the edge routers what instructions to attach to packets as they enter
   the network.  These last two operations can be achieved using PCEP
   and the PCE-based controller will assume responsibility for managing
   the space of labels or path identifiers used to determine how packets
   are forwarded.

### 3.1.6.  Service Function Chaining

Service Function Chaining (SFC) is described in [RFC7665].  It is the
process of directing traffic in a network such that it passes through
specific hardware devices or virtual machines (known as service
function nodes) that can perform particular desired functions on the
traffic.  The set of functions to be performed and the locations at
which they are to be performed is known as service function chain.
Each packet is marked as belonging to a specific chain and that
marking lets each successive service function node know which
functions to perform and to which service function node to send the
packet next.

To operate an SFC network the service function nodes must be
configured to understand the packet markings and the edge nodes must
be told how to mark packets entering the network.  Additionally it
may be necessary to establish tunnels between service function nodes
to carry the traffic.

Planning an SFC network requires load balancing between service
function nodes and traffic engineering across the network that
connects them.  These are operations that can be performed by a PCE-
based controller, and that controller can use PCEP to program the
network and install the service function chains and any required
tunnels.

### 3.2.  High-Level Applicability

This section provides a list of the high-level functions that can be
delivered by using a PCE-based controller.

### 3.2.1.  Traffic Engineering

According to [RFC2702], Traffic Engineering (TE) is concerned with
performance optimization of operational networks.  In general, it
encompasses the application of technology and scientific principles
to the measurement, modeling, characterization, control of Internet
traffic, and the application of such knowledge and techniques to
achieve specific performance objectives.

From a practical point of view this involves having an understanding
of the topology of the network, the characteristics of the nodes and
links in the network, and the traffic demands and flows across the
network.  It also requires that actions can be taken to ensure that
traffic follows specific paths through the network.

PCE was specifically developed to address TE in an MPLS network, and
so a PCE-based controller is well suited to analyze TE problems and

supply answers that can be installed in the network using PCEP.  PCEP
can be responsible for initiating paths across the network through a
control plane, or for installing state in the network node by node
such as in a Segment Routed network (see Section 3.1.5) or by
configuring IGP metrics.

## 3.2.2.  Traffic Classification

Traffic classification is an important part of traffic engineering.
It is the process of looking at a packet to determine how it should
be treated as it is forwarded through the network.  It applies in
many scenarios including MPLS traffic engineering (where it
determines what traffic is forwarded onto which LSPs), segment
routing (where it is used to select which set of forwarding
instructions to add to a packet), and service function chaining
(where it indicates along which service function chain a packet
should be forwarded).

Traffic classification is closely linked to the computational
elements of planning for the network functions just listed because it
determines how traffic load is balanced and distributed through the
network.  Therefore, selecting what traffic classification should be
performed by a router is an important part of the work done by a PCE-
based controller.

Instructions can be passed from the controller to the routers using
PCEP.  These instructions tell the routers how to map traffic to
paths or connections.  The instructions may use the concept of a
Forwarding Equivalence Class (FEC).

## 3.2.3.  Service Delivery

Various network services may be offered over a network.  These
include protection services (including end-to-end protection
[RFC4427], restoration after failure, and fast reroute [RFC4090]),
Virtual Private Network (VPN) service (such as Layer 3 VPNs [RFC4364]
or Ethernet VPNs [RFC7432]), or Pseudowires [RFC3985].

Delivering services over a network in an optimal way requires
coordination in the way that network resources are allocated to
support the services.  A PCE-based central controller can consider
the whole network and all components of a service at once when
planning how to deliver the service.  It can then use PCEP to manage
the network resources and to install the necessary associations
between those resources.

## 4.  Protocol Implications

   PCEP is push-pull protocol that is designed to move requests and
   responses between a server (the PCE) and Path Computation Clients
   (PCCs - the network elements).  In particular, it has a message
   (PCInitiate [I-D.ietf-pce-pce-initiated-lsp]) that can be sent by the
   PCE to install state or cause actions at the PCC, and a response
   message (PCRpt) that is used to confirm the request.

   As such, there is an expectation that only relatively minor changes
   to PCEP are required to support the concept of a PCE-based
   controller.  The only work expected to be needed is small extensions
   to carry additional or specific information elements for the
   individual use cases.  Where possible, consistent with the general
   principles of how protocols are extended, any additions to the
   protocol should be made in a generic way such that they are open to
   use in a range of applications.

   It is anticipated that new documents will be produced for each use
   case dependent on support and demand.  Such documents will explain
   the use case and define the necessary protocol extensions.

## 5.  Security Considerations

   Security considerations for a PCE-based controller are little
   different from those for any other PCE system.  That is, the
   operation relies heavily on the use and security of PCEP and so
   consideration should be given to the security features discussed in
   [RFC5440] and the additional mechanisms described in
   [I-D.ietf-pce-pceps].

   It should be observed that the trust model of a network that operates
   with out a control plane is different from one with a control plane.
   The conventional "chain of trust" used with a control plane is
   replaced by individual trust relationships between the controller and
   each individual NE.  This model may be considerably easier to manage
   and so is more likely to be operated with a high level of security.
   However debate will rage over overall system security and the
   opportunity for attacks in an architecture with a central controller
   since the network can be vulnerable to denial of service attacks on
   the controller, and the forwarding system may be harmed by attacks on
   the messages sent to individual routers.  In short, while the
   interactions with a PCE-based controller are not substantially
   different from those in any other SDN architecture, the security
   implications of SDN are still open for discussion.  The IRTF's SDN
   Research Group (SDNRG) continues to discuss this topic.

It is expected that each new document that is produced for a specific
use case will also include considerations of the security impacts of
the use of a PCE-based central controller on the network type and
services being managed.

## 6.  Manageability Considerations

The architecture described in this document is a management
architecture: the PCE-based controller is a management component that
controls the network through a southbound management protocol (PCEP).

RFC 5440 [RFC5440] contains a substantive manageability
considerations section that examines how a PCE-based system and a
PCE-enabled system may be managed.  A MIB module for PCEP was
published as RFC 7420 [RFC7420] and a YANG module for PCEP has also
been proposed [I-D.pkd-pce-pcep-yang].

## 7.  IANA Considerations

This document makes no requests for IANA action.

## 8.  Contributors

The following people contributed to discussions that led to the
development of this document:


        Cyril Margaria
        Email: cmargaria@juniper.net

        Sudhir Cheruathur
        Email: scheruathur@juniper.net

        Dhruv Dhody
        Email: dhruv.dhody@huawei.com

        Daniel King
        Email: daniel@olddog.co.uk

        Iftekhar Hussain
        Email: IHussain@infinera.com

        Anurag Sharma
        Email: AnSharma@infinera.com

        Eric Wu
        Email: eric.wu@huawei.com

## 9.  Acknowledgements

The ideas in this document owe a lot to the work started by the
authors of [I-D.zhao-teas-pcecc-use-cases] and
[I-D.zhao-pce-pcep-extension-for-pce-controller].  The authors of
this document fully acknowledge the prior work and thank those
involved for opening the discussion.  The individuals concerned are:
King Ke, Luyuan Fang, Chao Zhou, Boris Zhang, Zhenbin Li.

This document has benefited from the discussions within a small ad
hoc design team the members of which are listed as document
contributors.

Thanks to Michael Scharf and Andy Malis for a lively discussion of
this document.

## 10.  References

### 10.1.  Normative References

[RFC4655]  Farrel, A., Vasseur, J., and J. Ash, "A Path Computation
           Element (PCE)-Based Architecture", RFC 4655,
           DOI 10.17487/RFC4655, August 2006,
           <http://www.rfc-editor.org/info/rfc4655>.

### 10.2.  Informative References

[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-07 (work in
           progress), July 2016.

[I-D.ietf-pce-pceps]
           Lopez, D., Dios, O., Wu, W., and D. Dhody, "Secure
           Transport for PCEP", draft-ietf-pce-pceps-10 (work in
           progress), July 2016.

[I-D.ietf-pce-stateful-pce]
           Crabbe, E., Minei, I., Medved, J., and R. Varga, "PCEP
           Extensions for Stateful PCE", draft-ietf-pce-stateful-
           pce-15 (work in progress), July 2016.

[I-D.ietf-spring-segment-routing]
           Filsfils, C., Previdi, S., Decraene, B., Litkowski, S.,
           and R. Shakir, "Segment Routing Architecture", draft-ietf-
           spring-segment-routing-09 (work in progress), July 2016.

[I-D.pkd-pce-pcep-yang]
          Dhody, D., Hardwick, J., Beeram, V., and j.
          jefftant@gmail.com, "A YANG Data Model for Path
          Computation Element Communications Protocol (PCEP)",
          draft-pkd-pce-pcep-yang-06 (work in progress), July 2016.

[I-D.zhao-pce-pcep-extension-for-pce-controller]
          Zhao, Q., Li, Z., Dhody, D., and C. Zhou, "PCEP Procedures
          and Protocol Extensions for Using PCE as a Central
          Controller (PCECC) of LSPs", draft-zhao-pce-pcep-
          extension-for-pce-controller-03 (work in progress), March
          2016.

[I-D.zhao-teas-pcecc-use-cases]
          Zhao, Q., Li, Z., Khasanov, B., Ke, Z., Fang, L., Zhou,
          C., Communications, T., Rachitskiy, A., and A. Gulida,
          "The Use Cases for Using PCE as the Central
          Controller(PCECC) of LSPs", draft-zhao-teas-pcecc-use-
          cases-01 (work in progress), July 2016.

[RFC2702]  Awduche, D., Malcolm, J., Agogbua, J., O'Dell, M., and J.
          McManus, "Requirements for Traffic Engineering Over MPLS",
          RFC 2702, DOI 10.17487/RFC2702, September 1999,
          <http://www.rfc-editor.org/info/rfc2702>.

[RFC3985]  Bryant, S., Ed. and P. Pate, Ed., "Pseudo Wire Emulation
          Edge-to-Edge (PWE3) Architecture", RFC 3985,
          DOI 10.17487/RFC3985, March 2005,
          <http://www.rfc-editor.org/info/rfc3985>.

[RFC4090]  Pan, P., Ed., Swallow, G., Ed., and A. Atlas, Ed., "Fast
          Reroute Extensions to RSVP-TE for LSP Tunnels", RFC 4090,
          DOI 10.17487/RFC4090, May 2005,
          <http://www.rfc-editor.org/info/rfc4090>.

[RFC4364]  Rosen, E. and Y. Rekhter, "BGP/MPLS IP Virtual Private
          Networks (VPNs)", RFC 4364, DOI 10.17487/RFC4364, February
          2006, <http://www.rfc-editor.org/info/rfc4364>.

[RFC4427]  Mannie, E., Ed. and D. Papadimitriou, Ed., "Recovery
          (Protection and Restoration) Terminology for Generalized
          Multi-Protocol Label Switching (GMPLS)", RFC 4427,
          DOI 10.17487/RFC4427, March 2006,
          <http://www.rfc-editor.org/info/rfc4427>.

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

   [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,
              <http://www.rfc-editor.org/info/rfc6805>.

   [RFC7025]  Otani, T., Ogaki, K., Caviglia, D., Zhang, F., and C.
              Margaria, "Requirements for GMPLS Applications of PCE",
              RFC 7025, DOI 10.17487/RFC7025, September 2013,
              <http://www.rfc-editor.org/info/rfc7025>.

   [RFC7399]  Farrel, A. and D. King, "Unanswered Questions in the Path
              Computation Element Architecture", RFC 7399,
              DOI 10.17487/RFC7399, October 2014,
              <http://www.rfc-editor.org/info/rfc7399>.

   [RFC7420]  Koushik, A., Stephan, E., Zhao, Q., King, D., and J.
              Hardwick, "Path Computation Element Communication Protocol
              (PCEP) Management Information Base (MIB) Module",
              RFC 7420, DOI 10.17487/RFC7420, December 2014,
              <http://www.rfc-editor.org/info/rfc7420>.

   [RFC7432]  Sajassi, A., Ed., Aggarwal, R., Bitar, N., Isaac, A.,
              Uttaro, J., Drake, J., and W. Henderickx, "BGP MPLS-Based
              Ethernet VPN", RFC 7432, DOI 10.17487/RFC7432, February
              2015, <http://www.rfc-editor.org/info/rfc7432>.

   [RFC7491]  King, D. and A. Farrel, "A PCE-Based Architecture for
              Application-Based Network Operations", RFC 7491,
              DOI 10.17487/RFC7491, March 2015,
              <http://www.rfc-editor.org/info/rfc7491>.

   [RFC7665]  Halpern, J., Ed. and C. Pignataro, Ed., "Service Function
              Chaining (SFC) Architecture", RFC 7665,
              DOI 10.17487/RFC7665, October 2015,
              <http://www.rfc-editor.org/info/rfc7665>.

Authors' Addresses

   Adrian Farrel (editor)
   Juniper Networks

   Email: adrian@olddog.co.uk

Quintin Zhao (editor)
Huawei Technologies
125 Nagog Technology Park
Acton, MA   01719
USA

Email: quintin.zhao@huawei.com


Robin Li
Huawei Technologies
Huawei Bld., No.156 Beiqing Road
Beijing   100095
China

Email: lizhenbin@huawei.com


Chao Zhou
Cisco Systems

Email: chao.zhou@cisco.com