Path Computation Element Implementation Agreement

OIF-PCE-IA-01.0

June 2013

Implementation Agreement created and approved
by the Optical Internetworking Forum
www.oiforum.com

The OIF is an international non-profit organization with over 90 member companies, including the world's leading carriers and vendors. Being an industry group uniting representatives of the data and optical worlds, OIF's purpose is to accelerate the deployment of interoperable, cost-effective and robust optical internetworks and their associated technologies. Optical internetworks are data networks composed of routers and data switches interconnected by optical networking elements.

With the goal of promoting worldwide compatibility of optical internetworking products, the OIF actively supports and extends the work of national and international standards bodies. Working relationships or formal liaisons have been established with IEEE 802.1, IEEE 802.3ba, IETF, IP-MPLS Forum, IPv6 Forum, ITU-T SG13, ITU-T SG15, MEF, ATIS-OPTXS,ATIS-TMOC, TMF and the XFP MSA Group.

For additional information contact:
The Optical Internetworking Forum, 48377 Fremont Blvd.,
Suite 117, Fremont, CA 94538
510-492-4040 Φ info@oiforum.com

www.oiforum.com

Working Group: **Network & Operations**

TITLE: **PCE IA**

SOURCE:

Lyndon Y. Ong
Technical Editor
Ciena
920 Elkridge Landing Rd
Linthicum, MD 21090
Phone: +1 408 962 4929
Email: lyong@ciena.com

Jonathan Sadler
Technical Committee Chair
Tellabs
1415 W. Diehl Rd
Naperville IL 60563
Phone: +1.630.798.6182
Email: jonathan.sadler@tellabs.com

Rémi Theillaud
Networking & Operations Working Group Chair
Marben Products
176 rue Jean Jaurès
92800 Puteaux, France
Phone: +33 1 7962 1022
Email: remi.theillaud@marben-products.com

ABSTRACT: This Implementation Agreement defines Path Computation Element procedures and parameters for intra-carrier inter-domain use.

1 Table of Contents

2 List of Figures

3 List of Tables

4 Introduction

This Implementation Agreement is intended to define methods to apply Path Computation Element work from the IETF to networks that are based on ITU-T ASON architecture and OIF protocols, including multi-layer networks and multi-level hierarchy as defined in the ASON architecture. It uses PCE protocol specifications defined by the IETF and does not define new extensions to PCE protocol, although it makes use of PCE mechanisms for vendor-specific encoding. It identifies potential extensions to and compatibility with existing OIF IAs (routing, signaling security). This IA is scoped to intra-carrier use as inter-carrier routing issues have not been fully considered.

As Automatically Switched Optical Networks (ASONs) are deployed into new and existing networks, it cannot be assumed that such networks will be homogeneous (e.g., with respect to transport technologies, vendors, or approach to management and control). This is true even within a single carrier's network. To support deployment of an optical control plane into a heterogeneous environment, it is essential to introduce and support the concept of control domains, and in particular, the specification of the signaling and routing information exchanged between such domains.

A control domain is an architectural construct from ITU-T Recommendation [G.8080] that provides for encapsulation and information hiding, and the characteristics of the control domain are the same as those of its constituent set of distributed architectural components. The E-NNI reference point is defined to exist between control domains. The nature of the information exchanged between control domains across the E-NNI reference point captures the common semantics of the information exchanged amongst its constituent components, while allowing for different representations inside each control domain. Control domains are generally derived from architectural component types that serve a particular purpose; e.g., signaling control domains or routing control domains. Typically, signaling and routing control domains are expected to be congruent within ASON networks. The E-NNI reference point becomes an E-NNI signaling and routing interface when instantiated by signaling and routing protocols.

Figure 1 illustrates a simple example of a control plane subdivided into routing control domains interconnected by routing E-NNI interfaces. This example shows different domains potentially utilizing different I-NNI routing protocols communicating across the E-NNI interfaces by using a common set of signaling and routing protocols.

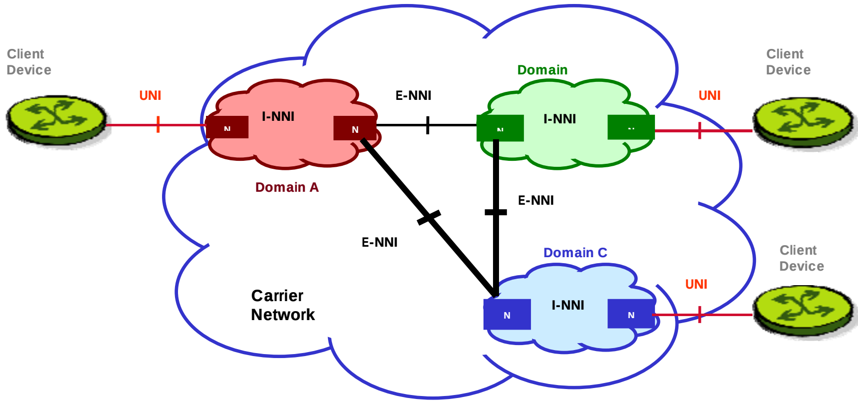


Figure 1: Example of Control Plane Configuration with Different Routing Control Domains

4.1 Problem Statement

The E-NNI routing protocol disseminates topology information across multiple domains, which allows the source node to compute an end-to-end path for a particular connection. However, an individual Routing Controller may advertise an abstracted topology for its domain, resulting in some loss of accuracy of end-to-end path computation.

Very large scale multi-domain ASON environments may need the use of PCE techniques to support precise path computation without impacting routing scalability. PCE enables path computation across multiple domains without a requirement to advertise the detailed topology of each domain across the E-NNI. PCE also supports more complex path computation or path computation that is more precisely done on a global basis rather than on a distributed basis.

PCE may support better optimized placement of connections, but it may also involve greater use of processing power and time as a byproduct. While in some cases there is a clear advantage to refine the path computation for a multi-domain service creation, this approach may not be suitable for restoration or other procedures where decision times are critical.

Some of the possible optimizations based on PCE are:

- When alternative paths exist, assess the real path cost before the signaling phase.

- A PCE attached to a domain may take into account during the intra domain path computation phase some local parameters and reply with a path and an E2E significant metric reflecting them.
- A PCE may take into account the diversity constraint when building a service with diversity constraint prior to the signaling phase.

A procedure on the methods of applying PCE in an ASON environment is needed.

4.2 Scope

The scope of this agreement includes the following items:

- Applying PCE to ASON and OIF E-NNI routing/signaling
  - Definition of PCE use cases and requirements
  - Use of components and identifiers
- PCE and ENNI routing relationship
  - Associating PCE with abstracted topology elements
  - Method for PCE discovery
    - static provisioning
    - dynamic discovery using OSPF
  - Determination of the appropriate next PCE to query
- Use of PCE protocol
  - PCE objective functions and constraints
  - PCE computation procedures, messages and objects
- PCE security & Network confidentiality
- Compatibility with E-NNI Routing and Signaling Implementation Agreements

The base protocols used by this document are OSPFv2 [RFC2328] with extensions for Traffic Engineering [RFC3630], GMPLS [RFC4202, RFC4203], and PCE Discovery [RFC5088], together with RSVP-TE [RFC3209, RFC3473] and PCEP [RFC5440].

This agreement does not address location of PCE functionality (e.g., server-based vs. network element-based). This is left to implementation.

This agreement does not address procedures to handle PCE Server failure. This is left to the implementation. Procedures to monitor the availability of a chain of PCE Servers are outside the scope of this agreement.

The mechanism by which a PCC discovers its local PCE server within a vendor domain is a matter for the I-NNI routing protocol and as such is outside the scope of this agreement.

4.3 Relationship to Other Standards Bodies

This document, to the maximum extent possible, uses standards and specifications already available from other organizations. Specifically,

- ITU SG15 – G.8080, G.7715 and G.7715.2 high level architecture
- IETF WGs – any related GMPLS & PCE extension work
- TMF MTNM – informational

This version of the implementation agreement also documents private extensions, codepoints and formats of these extensions based on the E-NNI 2.0 Routing and Signaling implementation agreements [E-NNI-R], [E-NNI-S].

It is the intent of OIF to develop E-NNI protocols in close alignment with ITU-T Recommendations, and foundation IETF RFCs. As such, the OIF has aligned formats with IETF and ITU-T standard specifications where possible and will continue to pursue alignment with standards in its future work. As additional standard specifications become available that address functions included in this Implementation Agreement, additional revisions for further alignment with these standards will be considered.

4.4 Merits to OIF

This document will provide the information necessary to design a PCE and E-NNI routing combined implementation. This document will enable carriers to operate large, multi-domain networks with greater control over connection routing. This meets requirements identified by OIF Carriers.

4.5 Working Groups

Networking and Operations Working Group
Carrier Working Group
Interoperability Working Group

4.6 Document Organization

This document is organized as follows:

- Section 1: Introduction and Scope of the Document

- Section **Error! Reference source not found.**: Terminology and Abbreviations
- Section 3: Basic PCE Components
- Sections 4 through 6: PCE Protocol and Procedures
- Section 7: PCE Security and Logging
- Section 8: Compatibility
- Section 9: References
- Appendices

4.7 Keywords

The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in [RFC2119].

5 Terminology and Abbreviations

5.1 Abbreviations

The following abbreviations are used in this implementation agreement.

ASON	Automatically Switched Optical Networks
BN	Border Node
BRPC	Backward-Recursive PCE-based Computation
GMPLS	Generalized Multi-Protocol Label Switching
GRE	Generic Routing Encapsulation
E-NNI	External Network-Network Interface
ERO	Explicit Route Object
ID	Identifier
IETF	Internet Engineering Task Force
I-NNI	Internal Network-Network Interface
IP	Internet Protocol
IPSec	Internet Protocol Security
ISCD	Interface Switching Capability Descriptor
ITU-T	International Telecom Union - Telecommunications
LSA	Link State Advertisement
OSPF	Open Shortest Path First
PC	Protocol Controller
PCC	Path Computation Client
PCE	Path Computation Element
RA	Routing Area
RC	Routing Controller
RCD	Routing Control Domain
SCN	Signaling Communications Network
SN	Subnetwork
SNP	Subnetwork Point
SNPP	Subnetwork Point Pool
SRLG	Shared Risk Link Group
TE	Traffic Engineering
TLV	Type/Length/Value
TNA	Transport Network Assigned Name

5.2 Terminology

The following terms are used in this implementation agreement.

Control Domain	This terminology is adopted from ITU-T [G.8080]. A type of transport domain where the criterion for membership is the scope of a control plane component responsible for the transport resources within the transport domain.
Inter-domain Link	A link with endpoints in two different Routing Areas at a particular level of the routing hierarchy.
Intra-domain Link	A link with both endpoints within the same Routing Area at a particular level of the routing hierarchy.
Layer	This terminology is adopted from ITU-T [G.805]. A layer (network) is a "topological component" that represents the complete set of access groups of the same type which may be associated for the purpose of transferring information.
Level	This terminology is adopted from ITU-T [G.8080]. A routing hierarchy describes the relationships between a RA and a containing RA or contained RAs. RAs at the same depth within the routing hierarchy are considered to be at the same routing level.
Node ID	This terminology is adopted from ITU-T [G.7715.1]. The Node ID identifies a node in the transport topology graph. A node may represent either an abstraction of a Routing Area or a subnetwork.
Protocol Controller	This terminology is adopted from ITU-T [G.8080]. The Protocol Controller provides the function of mapping the parameters of the abstract interfaces of the control components into messages carried by a protocol to support interconnection via an interface.
Routing Area	This terminology is adopted from [G.8080]: A routing area is defined by a set of subnetworks, the SNPP links that interconnect them, and the SNPPs representing the ends of the SNPP links exiting that routing area. A routing area may contain smaller routing areas interconnected by SNPP links. The limit of subdivision results in a routing area that contains a subnetwork.
Routing Controller	This terminology is adopted from [G.7715]. The Routing Controller functional component provides the routing

service interface and is responsible for coordination and dissemination of routing information.

Routing Control Domain This terminology is adopted from [G.8080]. A transport domain is a set of transport resources grouped according to some criteria established by operator policies. An RCD is a type of transport domain where the criterion for membership is assignment to an RC federation for the purposes of transport resource advertisement.

Signaling Control Network The packet network that carries control plane messages between Protocol Controllers

TE Link This definition is per [RFC4203], which defines a TE link as a "logical" link that has TE properties. The TE link is logical in a sense that it represents a way to group or map the information about certain physical resources (and their properties) into the information used by Constrained SPF for path computation.

6 Basic PCE Components

6.1 PCE Functionality

The Path Computation Element architecture is defined in IETF RFC 4655, and consists of a Path Computation Client (PCC) and a Path Computation Element (PCE), and a set of standardized interfaces that allow the PCC to discover available PCEs and for the PCC (or another PCE) to send a request to the PCE for a path to a specified destination satisfying some set of constraints and objectives. This allows the deployment of PCEs that supplement path computation available locally, using additional routing data or more powerful processing available to the PCE.

6.2 PCE Use Cases for E-NNI

The E-NNI routing protocol disseminates topology information across multiple domains, which allows the source node to compute an end-to-end path for a particular connection. However, an individual Routing Controller may advertise an abstracted topology for its domain, resulting in some loss of accuracy of end-to-end path computation. PCE can be used to improve path computation in a number of situations. Two example use cases are discussed below. This specification does not define specific rules for when to use PCE and when not to use PCE – this is a policy decision on the part of the network operator.

6.2.1 Second stage path computation

Where initial path computation using the E-NNI topology shows multiple possible paths with similar cost and other characteristics, PCE can be used to support a second stage of path computation to compare potential paths using topology information internal to the traversed domains.

6.2.2 Complex path computation

For some types of path computation, distributed computation using the E-NNI topology may result in a suboptimal solution because of either lack of a global view or loss of diversity information. An example is computation of working and backup paths for 1+1 service, where sequential computation of working, then protect, path may not lead to successful setup if, for example, the establishment of the shortest working path blocks establishment of a diverse backup path. Instead a more complex computation such as simultaneous computation of both working and backup can be done using PCE mechanisms.

6.3 Drawbacks of PCE

While PCE allows for more precise path computation, the following are impacts of PCE that may make use of PCE inappropriate for some types of connections

- PCE may add latency to connection setup, due to PCEP request and response
    - This may be a concern for connection restoration, where latency is critical
- PCE may add overhead to the network in general
    - PCE may require additional processing capacity in the network
    - PCE discovery and request/response adds to traffic on the SCN
    - PCE protocol requires support of TCP sessions between PCC and PCE

6.4 PCE Components and Identifiers

PCE supports path computation in large, multi-domain networks where this may be complex and may require special computational components and cooperation between the elements in different domains. A domain is defined in [RFC4655] to be any collection of network elements within a common sphere of address management or path computation responsibility. Current PCE protocol specifications have features defined to support typical IP domains such as IGP routing areas and Autonomous Systems that have associated identifiers in an IP network.

6.5 PCE Components and Identifiers as applied to ASON

ITU-T Recommendation G.7715.2 provides a model of path computation that relates PCE to ASON terminology, such as Routing Controllers. The G.7715.2 model allows discovery to be statically provisioned or dynamically determined using protocols, allows for hierarchical and peer level path requests, and describes step-by-step query, simultaneous query to multiple domains, and hierarchical query procedures.

ITU-T Recommendation G.8080 defines the administrative domain as representing a collection of entities that are grouped for a particular purpose, including administrative policy, capabilities, survivability, etc. The administrative domain may not automatically have an identifier that is uniquely associated with it in routing protocol. The implications are that :

- the ASON domain definition is somewhat broader than in RFC4655. ASON refers to a scoping of the transport plane. Instances of control plane components in G.8080 do operate over the transport plane and the scope of those transport plane resources is referred to as a type of domain prefixed by the control plane component type, for example, a routing domain or signaling domain.
- a particular ASON domain may require additional identification in routing since the ASON definition of domain is not identical to the definition of routing area used in IP routing protocols. An ASON routing domain is equivalent to the definition of routing area used in IP routing protocols. It does include hierarchical arrangements which are not found in all IP routing protocols.
- ASON domains may have different capabilities and protocols
  - In ASON domains, the I-NNI control plane is opaque to E-NNI and may use a different protocol
  - Within an I-NNI, Node Identifiers may be assigned from a domain-specific space that may overlap in value or take a different format from Node IDs used at the E-NNI. For the purposes of this specification, a Node ID is always assumed to be what is advertised or used at the E-NNI and is unique across the E-NNI.
- Inter-domain path computation in ASON allows for abstraction rather than complete hiding of TE information across domain boundaries
  - The E-NNI may advertise an abstract node or abstract link model of topology, for example.

- However, the E-NNI routing protocol provides a complete end-to-end topology across the network, including multiple domains.
    - This may provide some benefit in determining which PCE to contact for additional routing information.

PCE path computation operates based on the SNPP as defined in G.8080, and as a result any TNA to SNPP translation is assumed to occur before a request is made to the PCE. In some cases the path request may include multiple SNPPs if a particular TNA translates to multiple SNPPs to allow for diversity. It is possible, however, that due to abstraction the TNA to SNPP translation results in an abstract destination node rather than the actual destination node in the destination domain. As a result, the TNA MAY be included as additional routing information for use in path computation for the destination domain.

7 PCE Impacts on Routing

7.1 Advertisement of PCE Association

Existing E-NNI routing does not explicitly identify when abstraction has been used for advertised topology. Multiple types of abstraction may be in effect:

- Abstract Node – in this case, a domain is represented as a single node. The abstract node forms one endpoint of an advertised E-NNI link.
- Abstract Link – in this case, a domain is represented as virtual links connecting border nodes. The abstract link is not distinguishable from other links using the existing E-NNI routing.
- Abstract Link and Pseudonode – in this case, a domain is represented as a set of links and nodes that are abstracted from the physical topology. Nodes and links advertised may have no corresponding physical entity, but are not distinguishable from physical nodes and links in the existing E-NNI routing advertisement.

The decision to use PCE for an individual connection can be determined by operator policy rather than the use of a mechanism to identify when abstraction is being used.

However, there is a need to identify what PCE should be used and what SCN address is used to send queries to that PCE.

As discussed in section 7.5, the ASON routing domain is equivalent to the routing area used in IP routing, and accordingly the Routing Area ID can be used to identify the ASON routing domain for PCE purposes. It should be noted that this may require additional administration on the part of the carrier in order to assign RA ID values to domains if these do not already exist.

In order to support the association of topology elements advertised in the E-NNI such as links and nodes and the domain that they belong to, the Inter-RA Export Upward sub-TLV defined in [RFC6827] is used. The Inter-RA Export Upward sub-TLV contains a single Associated Routing Area ID associated with the topology element.

The Inter-RA Export Upward sub-TLV MUST be included in the following advertisements:

- Link TLV for intra-domain links (associates the link with its containing Routing Area)
- Node_Attributes TLV (associates the Node with its containing Routing Area)

The Inter-RA Export Upward sub-TLV are not included with Link TLV advertisements for E-NNI links, for which the Local and Remote Endpoints belong in different Routing Areas.

7.2 PCE Discovery

7.2.1 Static Methods

As a network option, the choice of PCE and address for reaching that PCE may be determined by static configuration rather than a dynamic mechanism. This reduces the overhead on the SCN that would otherwise be needed to support a dynamic discovery protocol.

Static configuration may be simplified by deploying PCE functionality on nodes that are already known in the E-NNI routing protocol by advertisement of routing functionality or dataplane node location, such as:

a) the Routing Controller or Controllers for a domain
b) the ingress border nodes to a domain
c) the abstract node representing a domain

7.2.2 Dynamic Methods

Static configuration methods for PCE discovery are not as flexible as dynamic methods, which allow PCE functionality to be located anywhere in the network, and support information about PCE supported capabilities and current availability. Dynamic PCE discovery may be supported by having the PCE advertise its presence, capabilities and location using ASON routing protocol so that the path computation client is aware of potential PCEs that can be used for path computation. This adds overhead to the routing protocol and requires path computation clients to support PCE discovery extensions, but provides a mechanism for dynamically introducing new PCEs into the network and providing information about their capabilities and status.

7.2.3 Routing Extensions for PCE discovery

For dynamic PCE discovery, standards for PCE discovery using OSPF [RFC 5088] MUST be used in the E-NNI routing protocol. Either the PCE servers in each domain should participate directly in the E-NNI routing protocol (so the PCE servers learn about each other directly from E-NNI routing), or else there must be a mechanism to allow PCE discovery information to be redistributed to and from the I-NNI routing protocol (in which case the PCE servers learn about each other from the I-NNI). PCE discovery relies on the association of a PCE and the domain that it supports, identified by Routing Area ID; E-NNI Routing extensions for advertisement of Routing Area ID are defined in section 8.1.1.

[RFC5088] defines an extension to the OSPF Router-Information LSA which a PCE uses to advertise its existence and functionality. This uses a new Type 6 OSPF PCED TLV and five sub-TLVs:

Sub-TLV type	Length	Name
1	variable	PCE-ADDRESS sub-TLV
2	4	PATH-SCOPE sub-TLV
3	4	PCE-DOMAIN sub-TLV
4	4	NEIG-PCE-DOMAIN sub-TLV
5	variable	PCE-CAP-FLAGS sub-TLV

For ASON application of PCE, these sub-TLVs are used as follows:

- The PCE-ADDRESS sub-TLV is mandatory and contains the SCN address of the advertising PCE in either IPv4 or IPv6 format as needed; only one sub-TLV is allowed per address type (i.e., IPv4 or IPv6)
- the PATH-SCOPE sub-TLV is mandatory and indicates the scope of path computation supported by the PCE. At a minimum, intra-area scope must be supported. Only one sub-TLV SHOULD be included, only the first occurrence is processed.
- the PCE-DOMAIN sub-TLV is not mandatory in [RFC5088] but is required for ASON application, and is set to the Routing Area ID of the domain for which the PCE provides path computation. Multiple instances are allowed.
- the NEIG-PCE-DOMAIN sub-TLV is optional and may be used to indicate the ability of the PCE to select egress paths to a particular neighboring domain or domains, identified by their Routing Area ID. Multiple instances are allowed.
    - If this sub-TLV is not present, it is assumed that the PCE is capable of selecting egress paths to all neighboring domains in the E-NNI topology.
- the PCE-CAP-FLAGS sub-TLV is optional and identifies path computation capabilities supported by the PCE. At most one instance is allowed. If no PCE-CAP-FLAGS sub-TLV is advertised, the following capabilities are required at a

minimum for ASON applications, and should be assumed to be supported by the advertising PCE:

- path computation with GMPLS link constraints
- bidirectional path computation

In case of dynamic PCE discovery, PCE Discovery procedures MUST be used as defined in [RFC5088].

8 PCE Query and Reply

8.1 Synchronized Path Computation

A synchronized path computation is one in which a PCE server computes the best possible result for N paths by taking all paths into consideration at one time, rather than by computing each path in isolation from the others. The results of a synchronized computation for N paths may differ from those obtained by computing each of the N paths in a serialized and independent fashion. This is the case if there is a dependency between the paths that are being computed, or if there is a limited network resource which might block some of the paths. See section 13.3 for some illustrative examples.
[RFC 5440] defines three types of dependency between paths.

1. The paths must be node diverse.
2. The paths must be link diverse.
3. The paths must be SRLG diverse.

For example, paths that are to be used for LSP protection have one of these types of dependency. See section 13.2 for an example of this type of computation. Objective functions that are applied to a set of paths can also introduce dependencies between those paths, because the PCE server may need to simultaneously vary all paths in the set to find a global maximum or minimum of the objective function. See section 8.2 for a discussion of objective functions.
A PCC requests a synchronized path computation by including one or more SVEC objects in its PCReq message.

- [RFC5440] defines the format of the SVEC object
- [RFC6007] defines the procedures for using the SVEC object to request synchronization of explicitly dependent paths.

It is optional for a PCE implementation of multi-domain E-NNI routing to support synchronized path computations. The PCE Capability Flags sub-TLV advertised in PCE Discovery provide one method of indicating if synchronized path computation is supported by the advertising PCE. This could alternatively be known by configuration at the PCC.

8.2 PCE Objective Functions

[RFC 5541] defines 3 objective functions for single path computation

- minimum cost path,
- minimum load path,
- maximum residual bandwidth path

minimum cost path MUST be supported for multi-domain E-NNI routing using PCE. The remaining objective functions may be optionally supported.

[RFC 5541] defines 3 objective functions for synchronized computation of multiple paths:

- minimize aggregate bandwidth consumption,
- minimize load of the most loaded link,
- minimize cumulative cost

If synchronized computation of multiple paths is supported by a PCE for ASON networks, at least the objective function to minimize cumulative cost MUST be supported as a minimum.

If a particular objective function is requested by the PCC and is not supported by the PCE, corresponding error procedures in [RFC 5441] MUST be followed.

[RFC 5440] defines the following metric types which can be used to determine the cost of a single path:

- TE metric
- IGP metric
- hop count

In addition, [pce-svc-aware] defines the following metric types for single paths.

- latency
- latency variation (jitter)
- packet loss

PCE server implementations of multi-domain E-NNI routing MUST support the use of TE metric to determine the cost of a single path.

[RFC 5541] defines the following metric types which can be used to determine the cost of synchronized paths:

- aggregate bandwidth consumption in the network
- load of the most loaded link in the network
- cumulative IGP cost of all paths in the synchronized computation
- cumulative TE cost of all paths in the synchronized computation

PCE server implementations of multi-domain E-NNI routing that support synchronized path computations MUST support the use of TE metric to determine the cumulative cost of all paths in a synchronized computation.

8.3 Inter-Domain PCE Computation Procedures

There are many methods of coordinating the actions of multiple PCEs in order to obtain an end-to-end path, including:

- per-domain (forward) path computation
- backwards recursive PCE-based computation (BRPC)
- hierarchical path computation

This specification does not mandate a specific method of coordination but provides guidelines for how some methods may be used for inter-domain path computation in ASON networks.

The sequence of domains to be traversed in the path may be provided as part of the constraints for path computation. The method for specifying the sequence of domains has been defined in [pce-seq] using the IRO object.

8.3.1 Per-Domain (Forward) Path Computation

Forward path computation is defined in [RFC5152] and uses path computation at each successive domain's PCE as the connection progresses from source to destination. It uses path request and response from the ingress node of each domain to its PCE and does not require PCE-to-PCE communication. Support of PCE within a domain depends on internal domain architecture and is out of scope from an E-NNI perspective.

8.3.2 Backwards Recursive PCE-based Computation

BRPC supports precise path computation without full topology distribution and guarantees computation of the least cost path where the sequence of domains to be traversed is known in advance. This may be a common case, for example, where the provider network is partitioned into metro and core domains where all connections between any two metro domains must pass through the common core, in which case the sequence of domains is fixed.

BRPC operation is defined in [RFC5441] and relies on the exchange of path computation requests from source PCC through intermediate PCEs to the destination domain's PCE, and a return of the reply containing a Virtual Shortest Path Tree (VSPT) that can be used to select the least cost end-to-end path. No changes are required to BRPC operation for ASON networks; supporting PCEs MUST be capable of determining the next hop PCE (if any) from the E-NNI topology together with static or dynamic PCE discovery methods discussed previously. The PCReq message MUST include the VSPT flag as part of the RP Object and the PCRep message MUST contain a set of EROs comprising the branches of the VSPT, if a path is available.

In BRPC operation, each PCE independently determines the succeeding PCE to consult either by local configuration or by discovery, using its knowledge of the E-NNI topology and addressing information. Some constraints on the choice of succeeding PCEs MAY be included in the PCReq message by specifying a sequence of Routing Areas or Nodes as discussed in section 9.3 above.

8.3.3 Hierarchical PCE

Hierarchical PCE is designed for environments where the sequence of domains to be crossed between ingress and egress is not known in advance and selection of the sequence may be complex.

8.3.3.1 Description of IETF hierarchical PCE architecture

The IETF hierarchical PCE architecture is defined in [pce-hier]. This subsection presents a summary of this architecture.

A multi-level hierarchy is introduced such that level N of the hierarchy is represented as a graph whose vertices are partitioned into non-overlapping domains and whose links are either intra-domain links or inter-domain links. The internal topology of each domain is not known to the other domains. Level N+1 of the hierarchy is represented as a graph whose vertices are the domains of level N, and whose links are the inter-domain links of level N. The vertices at level 1 are the nodes of the physical network, whereas the vertices above level 1 are abstract and represent collections of physical nodes. However, the links at all levels of the hierarchy represent physical links. There can be arbitrarily many levels of hierarchy above level 1.

Each domain at level N contains one or more PCE servers which know the internal topology of that domain at their level of the hierarchy. That is, they know all the level N vertices and links in their domain's graph, but if N>1, they do not know the level N-1 topology contained within each level N vertex or the Level N+1 topology.

There is a relationship between a PCE server at level N and the PCE servers in the level N+1 domain which contains the domain of the level N PCE server as a vertex. The level N PCE server is called a child PCE, and the level N+1 PCE servers with which it has a relationship are called parent PCEs. In general, there is a many-to-many relationship between child PCEs and parent PCEs. A PCE server at level N, where N>1, can have both child PCEs at level N-1 and parent PCEs at level N+1.

Path computation in the hierarchical PCE architecture proceeds as follows. A PCE server at level 1 receives a path computation request from a PCC. If the destination is contained within its topology graph, then it computes a path and returns a response. Otherwise, it forwards the request to one of its parent PCEs. The procedure continues as follows.

- A PCE server at level N receives a path computation request from one of its child PCEs.
- If the destination is contained within its topology graph, then it computes the best path as follows.
    - It selects a set of candidate paths within its level N topology graph.
    - For each selected path, it sends a path computation request to a child PCE in each level N-1 domain along the path, requesting a level N-1 path segment within the child's domain.
    - Once all responses are received, the PCE server constructs the optimum level N-1 path (chosen from amongst all candidate level N paths) by concatenating the path segments received from its child PCEs, and returns it to the child PCE that originally the request.
- Otherwise, if the destination is not contained within its topology graph, then it forwards the request to one of its parent PCEs, and the procedure completes recursively.

IETF PCEP extensions to support the hierarchical PCE architecture defined at the time of this document can be found in [pce-hier-pcep].

8.3.3.2 Comparison with ASON architecture

There is a close relationship between the domains at level N of the IETF hierarchical PCE architecture, and the level N routing domains of ASON. However, the ASON definition of a domain is broader than that allowed by the IETF hierarchical PCE architecture, since in ASON each level N routing domain can be represented at level N+1 by a more complex abstraction than a single vertex. In consequence:

- The links in the level N routing topology are not all physical links; some may be abstract links. Strictly, this differs from the assumptions of the IETF hierarchical PCE architecture. However, intra-domain links are always physical links at every level of the routing hierarchy.

- The vertices of a path in a level N routing domain do not necessarily have a 1:1 relationship with level N-1 routing domains. It is possible that consecutive vertices in the path may belong to the same level N-1 routing domain.

A key problem that a parent PCE must solve is to determine which vertex in its topology graph contains the destination endpoint (the source endpoint is assumed to be contained within the domain of the child PCE that sent the request.) The IETF hierarchical PCE architecture leaves this question open, and simply states that a mechanism must exist for the parent PCE to do this. In the ASON architecture the destination endpoint is an SNPP (see Section 7.5) and this destination is visible at each routing level. Hence in ASON, the parent PCE is always able to locate the destination SNPP in its E-NNI routing database.

8.3.3.3 Applicability of hierarchical PCE to ASON

If a PCE receives a path computation request from within its domain that it cannot satisfy using level N routing information (that is, the I-NNI routing information of that domain), then it should perform a hierarchical path computation at level N+1, as follows.

- It selects a level N+1 PCE to act as the parent PCE.
- The parent PCE constructs a set of candidate paths at level N+1.
- It sends path computation queries to the child PCEs for each E-NNI routing domain that is traversed by each candidate path, specifying the ingress and egress E-NNI link IDs of the child PCE's domain in the END-POINTS object of the PCReq. It assembles the results into an optimal path at level N.

When the parent PCE selects the child PCEs to send its path segment queries to, it first derives from each candidate path the sequence of domains that are to be traversed, and from this derive a sequence of child PCEs to query.
As discussed in sections 7.2.1, 7.2.2 and 7.2.3, there are multiple mechanisms by which the parent and child PCEs can learn each other's identities at each level of the hierarchy, for example:

- Each parent / child relationship can be configured by the network administrator.
- PCE discovery advertisements from the parent area may be redistributed into the child area so that the parent/child relationship is dynamically learned.

8.4 Multilayer path computation

The Multilayer Amendment to E-NNI 2.0 Routing and Signaling [E-NNI-ML] introduces some routing and signaling extensions for multilayer networks. The extensions related to path computation consist of new routing constructs (transitional links, pseudo-links and pseudo-nodes) and new signaling constructs (inverse multiplexing and nested ERO).

There are two basic models for PCE in a multilayer network: PCE for multiple layers and PCE per layer.

In the first model, a single PCE has responsibility for multiple layers. The PCE computes paths across multiple layers. This is useful when a sequence of domains is traversed as in BRPC where some domains have multilayer capabilities as is illustrated in **Error! Reference source not found.**.

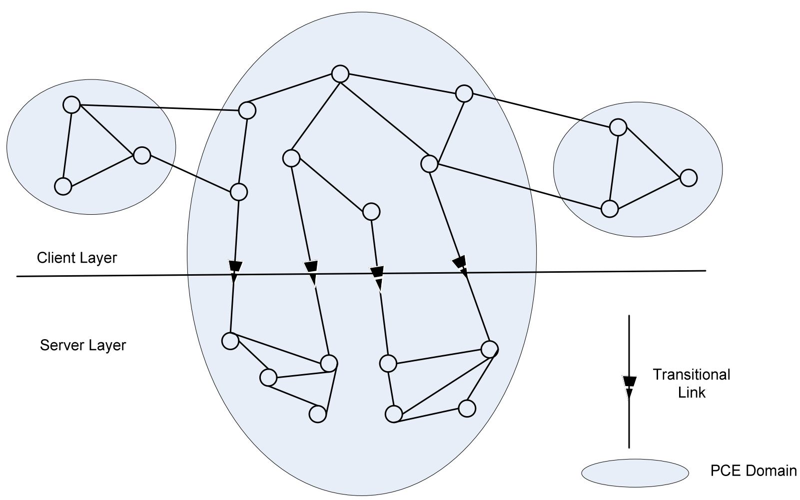


Figure 2: Multilayer PCE Example

In this case, the PCE for the multilayer area may return a path with a Multilayer ERO but is otherwise similar to single layer PCE. This type of configuration allows domains that don't have multilayer path computation capabilities to rely on PCE for more comprehensive multilayer path computation provided the other domains involved in the call establishment are able to forward the Multilayer ERO extensions in signaling.

In the second model, each PCE is responsible for path computation in a single layer. Multilayer path computation requires hierarchical PCE interactions where the client layer PCE computes paths through server layers and then requests server layer PCEs to compute paths in their respective server layer domains in order for the client layer PCE to determine the optimal multilayer end-to-end path. In Figure 3, an example multilayer network based on a PCE per layer is shown. The server layer domains are represented using a combination of pseudo-nodes and pseudo-links in the client layer. In this example, a different PCE is associated with each pseudo-node but it is possible to have the same PCE be responsible for several pseudo-nodes. The client layer PCE discovers the server layer PCEs through a static or dynamic method, similar to single layer PCE discovery described in section 9. Path computation is performed similarly to the hierarchical PCE described in section 9.3.3 with the following differences:

- The PCC resides in the top level, i.e. client layer.
- For each pseudo-node that is expanded via a lower layer PCE:
    - The PCReq from the client layer PCE includes the server layer source and destination TNA, or server layer SNPP source and destination, obtained from translating the client layer SNPP pair.
    - The PCRep may include a nested ERO or inverse multiplexed ERO in an OIF Vendor private extension (see [ML]).
    - The client layer PCE replaces the original pseudo-links and/or pseudo-node with the ERO from the PCRep. .

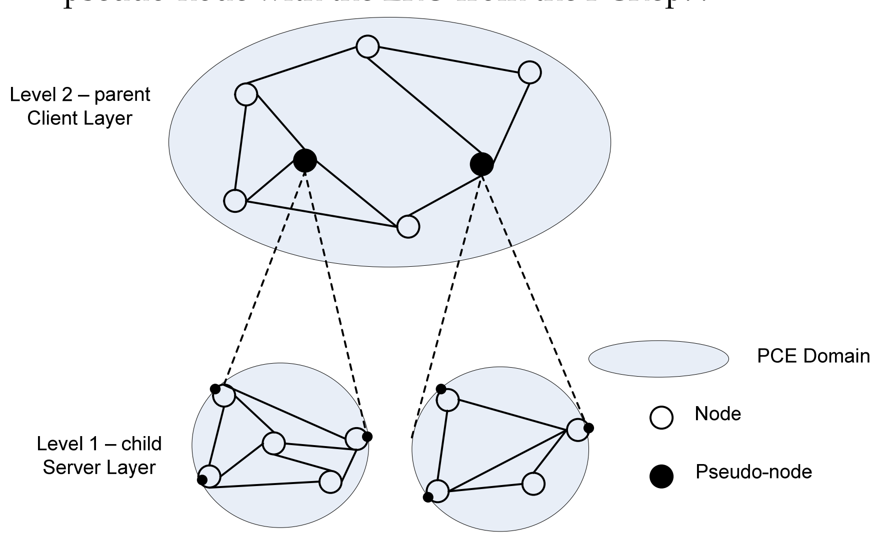


Figure 3: PCE per Layer Example

At this time, multilayer policies are expected to be configured in the PCEs. Example multilayer path computation policies are whether multilayer path computation is allowed or not, a list of layers to include or exclude from path computation and a maximum number of layers/adaptations to include in the path computation.

8.5 PCE Protocol Messages and Procedures

Basic procedures for PCE Protocol are defined in [RFC5440]. Initially a TCP session is established between PCC/PCE using port 4189 as the server port (the client port may be any port number). A PCE Protocol session is created using the TCP session for transport. Only one TCP and PCEP session are allowed between peers. Once a TCP session has been established between peers, the same session can be used by both sides for supporting their path computation requests.

All PCEP messages MUST be supported as specified in the standard [RFC5440]. ASON path computation requires that necessary information is sent with the query to the PCE.

A number of associated PCE objects are defined in the PCEP standard as well as in extensions to PCEP for support of GMPLS that are required for path computation in ASON networks.

For ASON Path Computation the following objects are defined for the PCReq message:

Table 1: PCReq Message Objects

Object	Reference	M/O	Contents
RP	[pce-gmpls]	M	Path computation characteristics
END-POINTS	[pce-gmpls]	M	Path source and destination
LSPA	[pce-gmpls]	O	Indicates protection requirements
GEN-BANDWIDTH	[pce-gmpls]	M	Bandwidth requirements
METRIC	[RFC5440]	M	Indicates which path metric to minimize and, optionally, an upper bound.
OF	[RFC5441]	O	Indicates Objective Function
PATH-KEY	[RFC5520]	O	Used when requesting expansion of Key
IRO	[pce-gmpls] [pce-seq]	O	Indicates required path members
XRO	[pce-gmpls]	O	Indicates exclusion required
SVEC	[RFC5440] [RFC6007]	O	Indicates synchronization and dependency requirements
VENDOR-CONSTRAINT	[pce-vendor]	O	Indicates OIF-specific information, e.g., G_UNI object

For ASON Path Computation the following objects are defined for the PCRep message:

Table 2: PCRep Message Objects

Object	Reference	M/O	Contents

RP	[pce-gmpls]	M	Path computation characteristics
NO-PATH	[RFC5440]	O	Indicates no path computed
LSPA	[pce-gmpls]	O	Indicates protection requirements
GEN-BANDWIDTH	[pce-gmpls]	O	Bandwidth requirements
METRIC	[pce-gmpls]	O	Contains computed metric
PATH-KEY	[RFC5520]	O	Used when returning a Key in place of a full ERO
IRO	[pce-gmpls] [pce-seq]	O	Indicates elements that caused path computation to fail
ERO	[pce-gmpls]	O	Used when returning a detailed path; when BRPC procedures are used, multiple EROs may be returned, each representing a branch of the VSPT.
VENDOR-CONSTRAINT	[pce-vendor]	O	Indicates OIF-specific information, e.g., G_UNI object
OF	[RFC5441]	O	Indicates the Objective Function that the PCE Server used to compute the path.

8.5.1 PCEP Scalability

The maximum size of a single PCEP message is 65,532 bytes. Since there is an upper bound on the size of a PCRep, it limits the amount of information that can be sent in a path computation response. In particular, it limits both the size of the ERO that can be returned, and the size of the VSPT that can be computed using BRPC procedures.

The required size of the VSPT is bounded by the maximum number of interconnecting links between any two consecutive domains in the pre-ordained domain sequence. It is left to the implementation to handle a VSPT that would exceed the message size limit.

It should be noted that Path-Key encryption provides a very compact representation of a segment of an ERO. Use of Path-Key encryption can lead to a smaller ERO and a smaller VSPT.

8.6 PCEP Objects

8.6.1 RP Object

The RP Object (Class = 2, Type = 1) contains a set of flags specifying characteristics of path computation requested. The following flags SHOULD be set:

- Strict path computation
- Bidirectional path computation

Additionally the VSPT Flag (bit 25) MUST be set if BRPC is being requested, as in [RFC5441].

8.6.2 END-POINTS Object

The END-POINTS Object (Class = 4, Type = 1 for IPv4, Type = 2 for IPv6; Type = 3 for Generalized Endpoints) contains the source and destination address of the requested path at the E-NNI level. As in [pce-gmpls], options are provided for indicating the Node ID, unnumbered endpoint/interface, label, etc. depending on the particular path request.

8.6.3 LSPA Object

The LSPA Object (Class = 9, Type = 1) carries LSP Attributes. For ASON connections, the L flag should be set to zero. As in [pce-gmpls] this is extended with a Protection TLV that carries the protection characteristics requested for the path.

8.6.4 GEN-BANDWIDTH Object

The GEN-BANDWIDTH Object (Class =tbd, Type=tbd) carries the TSpec for the requested path. As in [pce-gmpls], the type of TSpec is indicated and types are defined for Intserv, SONET/SDH, OTN and Ethernet. For G.709ed3 an additional type has been defined in [sig-g709v3].

8.6.5 METRIC Object

The METRIC Object (Class = 9, Type = 1) indicates the type of metric to be optimized as well as any bounds to be used when computing the path. For ASON connections, the TE-Metric MUST be supported. Other flags and information are provided as needed.

8.6.6 ERO Object

The ERO Object (Class=7, Type=1) carries the Explicit Route returned by path computation. Multiple ERO Objects can be carried in the PCRep message. The contents are encoded as in [OIF E-NNI 2.0 Signaling].

Note: the ERO Object returned by path computation can include Path Key information as specified in [RFC5553].

8.6.7 PATH-KEY Object

The PATH-KEY Object (Class = 16, Type = 1) is carried in the PCReq message when the PCC is requesting expansion of a received Path Key into the detailed intra-domain path segment. It can also be carried in the PCRep message as part of a returned ERO. Further details are defined in [RFC5520].

8.6.8 OF Object

The OF Object (Class = 21, Type = 1) may be carried in the PCReq to specify a particular objective function to be used during path computation, e.g., minimum cost, minimum load, etc. If no objective function is specified for path computation, minimum cost is assumed.

8.6.9 IRO Object

The IRO Object (Class = 10)[1] may be carried in the PCReq to specify a particular set of nodes, links or domains that must be included in the computed path. The base IRO format is defined in [RFC5440], and enhancements to allow a sequence of domains to be specified are defined in [pce-seq]. The requesting PCC may use the enhanced IRO of [pce-seq] to specify the sequence of domains that a computed inter-domain path must traverse by including a sequence of ASON routing area IDs in the IRO.

8.6.10 XRO Object

The XRO Object (Class = 17, Type = 1) may be carried in the PCReq to specify a particular set of nodes, links or domains that must be excluded from the computed path. The order in which the excluded objects are given is not significant. Further information on this object can be found in [RFC5521].

8.6.11 SVEC Object

The SVEC Object (Class = 11, Type = 1) may be carried in the PCReq when the PCC is requesting that a set of path computations be synchronized. It contains the request IDs of the requests to synchronize. If the requests are being synchronized to compute a path with protection, then the PCC must set one or more of the S, N and L flags to indicate whether the protecting path must be SRLG, node or link diverse, respectively. See section 8.1 for a discussion of synchronized path computations.

---

[1] The requesting PCC may use a Type 2 IRO object to specify a sequence of domains in a PCReq, in such a case the order in which the [pce-seq] IGP Area sub-objects are encoded is meaningful.

8.6.12 PCEP Vendor-Specific Constraints Extension

The Vendor Information Object (Class=tbd, 23 has been requested, Type=tbd, 1 has been requested) may be carried in the PCReq to convey vendor-specific information used in path computation.
In the context of this document, the Vendor Information Object MAY be used to carry the contents of the Generalized_UNI object for use in path computation. In this case, the format is as follows:

```
  0                   1                   2                   3
  0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |          Enterprise Number = 26041 for OIF                    |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 ~                 Enterprise-Specific Information               ~
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The Enterprise-Specific Information field contains the Generalized_UNI object as defined in [UNI 2.0-RSVP] section 9.2.5 (Class=229, Type=1). Subobjects of the Generalized_UNI object that may be contained include:

- Source and destination TNA
- Egress Label or SPC Label
- Service Level

Sub-Objects of the G_UNI object MAY be excluded if the requestor (PCC or PCE) determines that they would not be used in the requested path computation (e.g., source and destination TNA would not be included if the path computation request is for a segment of the end-to-end connection that does not include the source and destination TNA).

Other OIF-specific objects may be carried in the Vendor Information Object as needed for OIF procedures, e.g., extensions required to support multilayer operation.

8.6.13 PCEP versus RSVP-TE objects

Note that some PCEP object formats are shared with the RSVP-TE protocol. Some care is required to ensure that objects are formatted correctly as there are subtle differences in the two protocols.

The object-class and object-type fields of similar PCEP and RSVP-TE objects differs according to the table below.

Table 3: Similar PCEP and RSVP-TE Objects

Object	PCEP Class	PCEP C-Type	RSVP-TE Class	RSVP-TE C-Type

ERO	7	1	20	1
RRO	8	1	21	1
XRO	17	1	232	1

There is a difference in the format of PCEP's XRO object and RSVP-TE's XRO object. The PCEP object contains an extra 32 bits of information between the object header and the first subobject, which is composed of 16 reserved bits and 16 flag bits.

Each of the above objects, and also the PCEP IRO object, contains subobjects. A subobject contains a type field, a length field and a variable amount of subobject data. This resembles the TLV structures used on other PCEP objects, but there are two important differences.

1. In a subobject, the length field gives the length of the entire subobject, whereas in a PCEP TLV the length field gives the length of the value portion only.

2. In a subobject, the variable-length subobject data must be a multiple of 4 bytes in length, whereas in a PCEP TLV the value portion can be any length. Hence the length field of a subobject always contains a multiple of 4, whereas the length field of a PCEP TLV does not necessarily contain a multiple of 4. Note that sufficient 0x00 bytes are appended to a PCEP TLV to ensure that the next element of the PCEP message begins on a 4-byte aligned boundary.

9 PCE Impacts on Signaling

9.1 Incorporation of PCE Reply into the PATH message

The PCE Reply received at the source PCC, if path computation has been successful, will contain an ERO giving the end-to-end path to the specified destination. This ERO is then used in the PATH message to initiate call setup.

The ERO in the PCE Reply MAY contain one or more Path Key sub-objects if these were created during the PCE path computation [RFC 5553], including the PCE ID of the PCE generating the Path Key. These are incorporated into the PATH message ERO as received. Note: the PCE ID can take either a 32-bit or 128-bit format as defined in [RFC 5553], however the interpretation is specific to the domain originating the Path Key.

The ERO in the PCE Reply MAY contain one or more OIF-specific ERO objects as defined in [RSVP-PVT]. These are incorporated into the PATH message ERO as received.

An example sequence of PCE interaction is given in Appendix A.

10 Security and Logging for PCE

The Security Considerations in Section 10 of [RFC 5440] apply to PCE for the OIF E-NNI as defined in this IA, with the following changes:

- Terminology changes for ASON apply, esp.: “LSP” in [RFC 5440] is assumed to apply to optical/circuit switched paths as well as packet switched paths; clients for PCE are assumed to be ASON elements rather than “routers”.

- Section 4.4 of [CPSec] defines protocol security extensions applicable to PCEP and to the OSPFv2 extensions used in this IA. PCEP implementations running over IPv4 SHOULD support these security extensions. PCEP implementations running over IPv6 MUST support them (since implementation of IPsec ESP is already a requirement of IPv6).

- PCEP implementations MAY support security for PCEP over TLS as defined in [SecMang], especially if the security extensions in [CPSec] are not available. Note, however, that TLS cannot protect OSPFv2 and does not prevent TCP-level attacks such as SYN floods.

- PCEP implementations MAY use TCP-AO [RFC 5925] to secure PCEP, however this is not recommended as it lacks protection against some denial of service attacks and does not support a method of key distribution and rollover. Furthermore, use of TCP-AO would add another security method to what is already supported for OIF interfaces, with additional administration and operation overhead.

Additional security concerns apply for PCE mechanisms involving interaction between more than two entities, such as the BRPC mechanism shown in the figure of Appendix I. In this figure, the PCE for Domain 3 exchanges path information with the PCE for Domain 2 and the PCE for Domain 2 extends this information and passes this to the PCE for Domain 1. If Domain 3 and Domain 2 do not wish to reveal the results of every path computation to the lower-numbered domains, or allow the results to be changed, two possible options are:

- an implementation MAY support a policy module governing the conditions under which a PCE should participate in the BRPC procedure, as defined in [RFC 5441], or
- an implementation MAY use the PathKey mechanism defined in [RFC 5520] to restrict the information exchanged with other domains to a fixed length key.

PCC implementations SHOULD implement the security mechanisms defined in [SecMang] for their management interfaces.

PCC implementations SHOULD log the use of PCEP and the OSPFv2 extensions defined in this IA and any resulting error conditions with the methods in [LogAud]. PCEs SHOULD record comparable information in their logs sufficient to identify and reconcile any discrepancies.

11 Compatibility

11.1 Signaling Backwards Compatibility

This IA does not introduce any new signaling messages and fully supports all UNI signaling messages and attributes defined in E-NNI 2.0. This IA introduces the use of subobjects of the ERO object that are not defined in the E-NNI 2.0 specification and potentially impact E-NNI signaling as these may be returned in the PCEReply message with the intention of being inserted into the ERO of a subsequent connection request. It is assumed that border nodes of the domain of the PCE which returned these subobjects are capable of processing them on receipt of the ERO. Nodes outside of that domain are only required to understand the first non-local subobject of the ERO as received and are not required to "look ahead" into other subobjects of the ERO, as specified in [RFC5553], section 3.1, and are as a result not affected by the additional subobjects.

11.2 Routing Backwards Compatibility

This IA introduces new routing elements for PCE path computation and PCE discovery. The backwards compatibility issues for these constructs are described separately below.
In order to support PCE path computation it is necessary for RCs to include the Inter-RA Export Upward sub-TLV in Link TLVs and Node_Attributes TLVs

advertised for the RCs domain topology. If this sub-TLV is not advertised by the RC, then PCE-based enhanced path computation cannot be used for that associated domain and normal non-PCE-based path computation is used for paths using that domain. If the sub-TLV is received by an RC that does not recognize it, according to normal procedures [RFC3630] the sub-TLV continues to be flooded to other RCs and may be ignored internally by the receiving RC.

In order to support dynamic PCE discovery, the Type 6 PCED TLV must be advertised by the RC or RCs for the domain supporting the PCE and recognized by domains wishing to use dynamic PCE discovery. RCs which receive the PCED TLV and do not recognize it will flood the TLV according to normal procedures while silently ignoring the information internally. Procedures for compatibility of the PCED TLV are given in more detail in [RFC5088].

11.3 Multilayer Compatibility

This IA introduces no new routing or signaling elements for multilayer path computation. For support of multilayer it assumes that extensions and compatibility procedures defined in [ML] have been implemented.

12 References

12.1 ITU-T

[G.7715.2] – ITU-T Recommendation G.7715.2/Y.1706.2, "ASON architecture and requirements for remote route query", Feb. 2007.

12.2 OIF

[E-NNI-R] Optical Internetworking Forum Implementation Agreement, External Network-Network Interface (E-NNI) OSPF-based Routing - 2.0 (Intra-Carrier), OIF-ENNI-OSPF-02.0, July 2011.

[CPSec] Optical Internetworking Forum Implementation Agreement, Security Extension for UNI and NNI version 2.0, OIF-SEP-03.1, November 2011.

[LogAud] Optical Internetworking Forum Implementation Agreement, OIF Control Plane Logging and Auditing with Syslog version 1.1, OIF OIF-SLG-01.2, November 2011.

[SecMang] Optical Internetworking Forum Implementation Agreement, Security for Management Interfaces to Network Elements, OIF-SMI-03.0, November 2011.

[RSVP-PVT] Optical Internetworking Forum Implementation Agreement, OIF Application of Vendor Private Extensions in RSVP Implementation Agreement, OIF-RSVP-PVT-EXT-01.0, October 2011

[E-NNI-ML] OIF Implementation Agreement, “Multilayer Amendment to E-NNI 2.0 – Common Part”, OIF-ENNI-ML-AM-01.0, April 2013.

12.3 IETF

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2328] Moy, J., "OSPF Version 2", STD 54, RFC 2328, April 1998.

[RFC 3209] Awduche, D., et al, “RSVP-TE: Extensions to RSVP for LSP Tunnels”, Dec. 2001.

[RFC 3473] Berger, L. (ed), “Generalized Multi-Protocol Label Switching (GMPLS) Signaling Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Extensions”, Jan 2003.

[RFC 3630] Katz, D., et al, ” Traffic Engineering (TE) Extensions to OSPF Version 2”, Sept. 2003.

[RFC 4202] Kompella, K. and Rekhter, Y. (eds), “Routing Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)”, Oct. 2005.

[RFC 4203] Kompella, K. and Rekhter, Y. (eds), “OSPF Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)”, Oct. 2005.

[RFC 4655] – Farrel, A. et al, A Path Computation Element (PCE)-based Architecture”, August 2006.

[RFC 5088] – LeRoux, J.L. and Vasseur, J.P. (eds), “OSPF Extensions for Path Computation Element (PCE) Discovery”, January 2008.

[RFC 5152] – Vasseur, J.P. and Ayyangar, A. (eds), “A Per Domain Path Computation Method”, Feb. 2008.

[RFC 5440] – Vasseur, J.P. and LeRoux, J (eds), “Path Computation Element (PCE) Communication Protocol (PCEP)”, March 2009.

[RFC 5441] – Vasseur, J.P. (ed), “A Backward-Recursive PCE-based Computation (BRPC) Procedure”, April 2009.

[RFC 5520] – Bradford, R. (ed), “Preserving Topology Confidentiality in Inter-Domain Path Computation using a Path-Key-based Mechanism”, April 2009.

[RFC 5521] – Oki, E., T. Takeda and A. Farrel, “Extensions to the Path Computation Element Communication Protocol (PCEP) for Route Exclusions”, April 2009.

[RFC 5541] – LeRoux, J., et al, “Encoding of Objective Functions in the Path Computation Element Communication Protocol (PCEP)”, June 2009.

[RFC 5553] – Farrel, A. (ed), "Resource Reservation Protocol (RSVP) Extensions for Path Key Support", May 2009.

[RFC 5925] Touch, J., A Mankin, and R. Bonica, "The TCP Authentication Option," IETF RFC 5925, June 2010.

[RFC 6007] – Nishioka, I. And King, D., "Use of the Synchronization VECtor (SVEC) List for Synchronized Dependent Path Computations", September 2010.

[RFC 6805] – King, D. (ed.), Farrel, A. (ed.), "Application of the PCE Architecture to the Determination of a Sequence of Domains in MPLS and GMPLS", November, 2012.

[pce-hier-pcep] – Zhang, F., et al, draft-zhang-pce-hierarchy-extensions-03.txt, "Extensions to Path Computation Element Communication Protocol (PCEP) for Hierarchical Path Computation Elements (PCE)", Feb. 2, 2013, Work in Progress.

[pce-gmpls] –Margaria, C. (ed), draft-ietf-pce-gmpls-pcep-extensions-07.txt, "PCEP Extensions for GMPLS", October 22, 2012, Work in Progress.

[pce-vendor] –Zhang, F., et al, draft-ietf-pce-vendor-constraints-10.txt, "Conveying Vendor-Specific Constraints in the Path Computation Element Protocol", April 22, 2013, Work in Progress.

[sig-g709v3] Zhang, F., et al, draft-ietf-gmpls-signaling-g709v3-10.txt, "Generalized Multi-Protocol Label Switching (GMPLS) Signaling Extensions for the evolving G.709 Optical Transport Networks Control ", June 18, 2013, Work in Progress.

[pce-seq] –Dhody, et al, draft-ietf-pce- pcep-domain-sequence-02.txt, "Standard Representation Of Domain-Sequence", Feb. 18, 2013, Work in Progress; Dhody, et al, draft-dhody-ccamp-rsvp-te-domain-subobjects-01.txt, "Domain Subobjects for Resource ReserVation Protocol - Traffic Engineering (RSVP-TE)", April 2, 2013, Work in Progress.

[pce-svc-aware] – Dhody, et al, draft-dhody-pce-pcep-service-aware-05.txt, "Extensions to the Path Computation Element Communication Protocol (PCEP) to compute service aware Label Switched Path (LSP)", Feb. 25, 2013, Work in Progress.

13 Appendix A: PCE Examples

13.1 Simple PCE Example Sequence

In the simple example sequence below, Node A generates a PCE Request message to determine an end-to-end path to destination Node V, specifying backwards recursive computation.

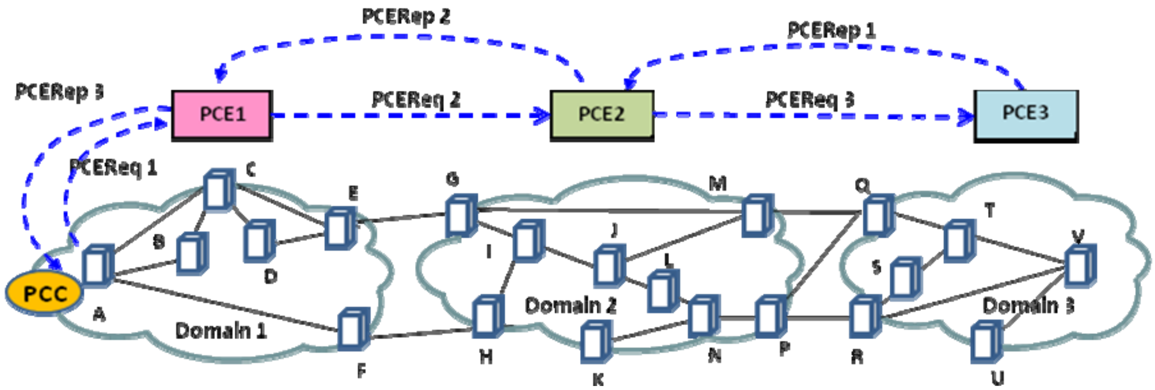


Figure 4: Simple PCE Example Sequence

Example message contents are given below for an ODU0 path computation request:

Table 4: PCE Request Sequence

Object	PCE Req 1	PCE Req 2	PCE Req 3
RP	Priority=0 Reopt = 0 Bidirectional=1 Strict=0 (strict) VSPT=1 Req-ID=001	Priority=0 Reopt = 0 Bidirectional=1 Strict=0 (strict) VSPT=1 Req-ID=101	Priority=0 Reopt = 0 Bidirectional=1 Strict=0 (strict) VSPT=1 Req-ID=110
END-POINTS	Source IPv4=A Dest IPv4=V	Source IPv4=A Dest IPv4=V	Source IPv4=A Dest IPv4=V
GEN-BANDWIDTH	TSpec Signal Type = 10	TSpec Signal Type = 10	TSpec Signal Type = 10
METRIC	T=2 (TE Metric) C=1 (provide metric in response)	T=2 (TE Metric) C=1 (provide metric in response)	T=2 (TE Metric) C=1 (provide metric in response)
IRO	Domain 2 routing area ID; Domain 3 routing area ID	Domain 3 routing area ID	
Vendor-Specific	Source TNA Dest TNA Service Level	Source TNA Dest TNA Service Level	Source TNA Dest TNA Service Level

Table 5: PCE Reply Sequence

Object	PCE Rep 1	PCE Rep 2	PCE Rep 3
RP	Req-ID=110	Req-ID=101	Req-ID=001

ERO*	Q/q2t_IF; T/t2v_IF	G/g2m_IF; M/m2q_IF; Q/q2t_IF; T/t2v_IF	A/a2c_IF; C/c2e_IF; E/e2g_IF; G/g2m_IF; M/m2q_IF; Q/q2t_IF; T/t2v_IF
METRIC	2	4	7
ERO*	R/r2v_IF	H/h2i_IF; I/i2j_IF; J/j2m_IF; M/2q_IF; Q/q2t_IF; T/t2v_IF	<not supplied>
METRIC	1	6	<not supplied>

*Note: the ERO contains Node ID and outgoing interface for each link

13.2 Complex PCE Example Sequence

In the second example sequence below, Node A generates a PCE Request message to determine an end-to-end diverse path for a 1+1 connection to destination Node V, using hierarchical approach. This example assumes that the PCE hosted in Domain 1 acts as a Parent PCE for Domain 0 (PCE1p) and as Child PCE for Domain 1 (PCE1c). Messages exchanged between PCE1p and PCE1c are not described. E-NNI routing is used at the highest level across Domain 0. It is assumed that PCE1p has the knowledge of the Domain 0 topology by using the information exchanged by the E-NNI routing protocol.

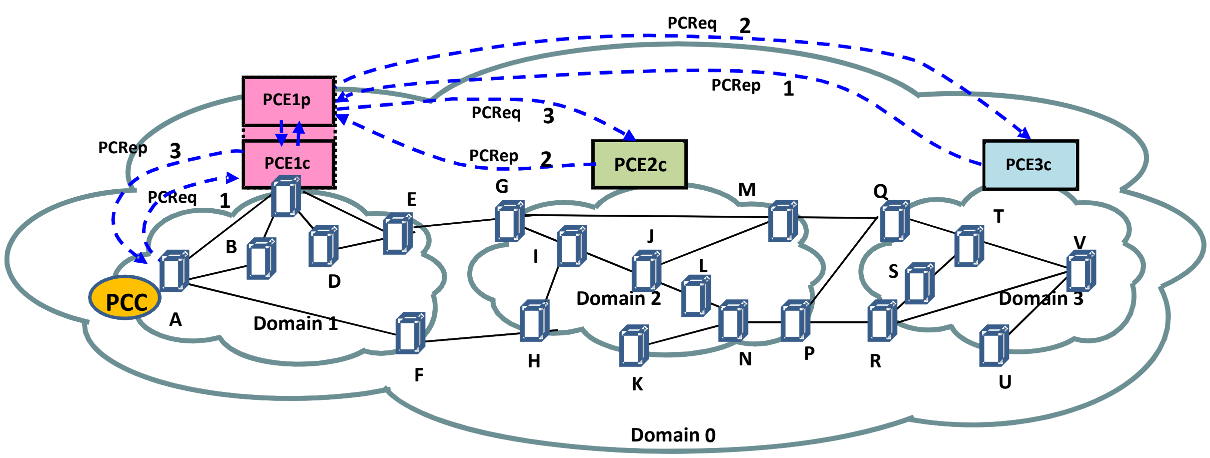


Figure 5: Complex PCE Example Sequence

Example message contents are given below for an ODU0 path computation request:

Table 6: Complex PCE Request Sequence

Object	PCReq 1	PCReq 2	PCReq 3
SVEC	Flags: S & N = 1 Req-ID#001 Req-ID#002	Flags: S & N = 1 Req-ID#010 Req-ID#011	Flags: S & N = 1 Req-ID#100 Req-ID#101

RP	Priority=0 Reopt = 0 Bidirectional=1 Strict=0 (strict) Req-ID=001	Priority=0 Reopt = 0 Bidirectional=1 Strict=0 (strict) Req-ID=010	Priority=0 Reopt = 0 Bidirectional=1 Strict=0 (strict) Req-ID=100
END-POINTS	Source IPv4=A Dest IPv4=V	Source IPv4=Q Dest IPv4=V	Source IPv4=G Dest IPv4=M
GEN-BANDWIDTH	TSpec Signal Type = 10	TSpec Signal Type = 10	TSpec Signal Type = 10
METRIC	T=2 (TE Metric) C=1 (provide metric in response)	T=2 (TE Metric) C=1 (provide metric in response)	T=2 (TE Metric) C=1 (provide metric in response)
Vendor-Specific	Source TNA Dest TNA Service Level	Source TNA Dest TNA Service Level	Service Level
RP	Priority=0 Reopt = 0 Bidirectional=1 Strict=0 (strict) Req-ID=002	Priority=0 Reopt = 0 Bidirectional=1 Strict=0 (strict) Req-ID=011	Priority=0 Reopt = 0 Bidirectional=1 Strict=0 (strict) Req-ID=101
END-POINTS	Source IPv4=A Dest IPv4=V	Source IPv4=R Dest IPv4=V	Source IPv4=P Dest IPv4=H
GEN-BANDWIDTH	TSpec Signal Type = 10	TSpec Signal Type = 10	TSpec Signal Type = 10
METRIC	T=2 (TE Metric) C=1 (provide metric in response)	T=2 (TE Metric) C=1 (provide metric in response)	T=2 (TE Metric) C=1 (provide metric in response)
Vendor-Specific	Source TNA Dest TNA Service Level	Source TNA Dest TNA Service Level	Service Level

Table 7: Complex PC Reply Sequence

Object	PC Rep 1	PC Rep 2	PC Rep 3
RP	Req-ID=10	Req-ID=100	Req-ID=001
ERO*	[Q/q2t_IF]; [T/t2v_IF]	[G/g2m_IF];	[A/a2c_IF]; [C/c2e_IF]; [E/e2g_IF]; [G/g2m_IF];

			[M/m2q_IF]; [Q/q2t_IF]; [T/t2v_IF]
METRIC	2	1	7
RP	Req-ID=11	Req-ID=101	Req-ID=001
ERO*	[R/r2v_IF]	[H/h2i_IF]; [I/i2j_IF] ; [J/j2l_IF]; [L/l2n_IF]; [N/n2p_IF]	[A/a2f_IF]; [F/f2h_IF]; [H/h2i_IF]; [I/i2j_IF] ; [J/j2l_IF]; [L/l2n_IF]; [N/n2p_IF]; [P/p2r_IF]; [R/r2v_IF]
METRIC	1	5	9

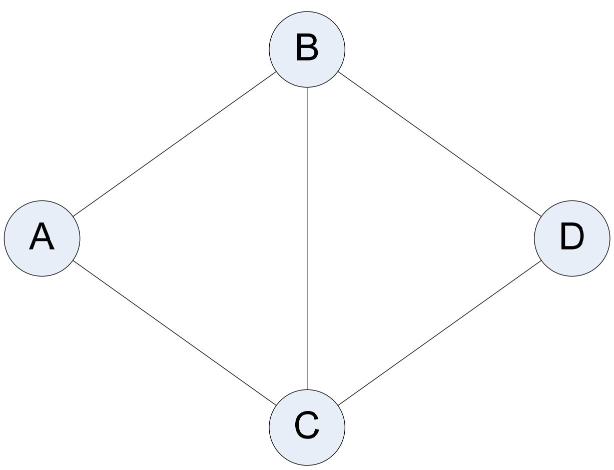
*Note: the ERO contains Node ID and outgoing interface for each link (might be encoded in a PATH_KEY)

13.3 Synchronized path computation examples

This section contains examples of dependent and independent synchronized path computations.

13.3.1 Example 1: dependent, synchronized computation

In the network graph below, two link-diverse paths are required from A to D which minimize the TE metric.



Link	TE metric
AB	1
AC	3
BC	1
BD	3
CD	1

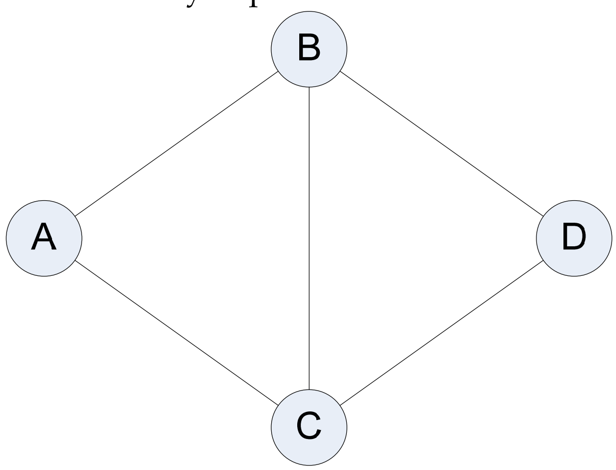
Figure 6: Dependent, Synchronized Computation

If the two paths were computed in a serialized, independent fashion, then the PCE server would compute the solution {AB, BC, CD} for the first path, with a globally minimized cumulative TE metric of 3. Unfortunately, the second path computation would then fail because there is no alternate link-diverse path from

A to D. If, however, the two path computations were synchronized, then the PCE server could compute the solutions {AB, BD} and {AC, CD}.

13.3.2 Example 2: independent, synchronized computation

In the network graph below, two 10Mbps paths are required from A to D which each has a minimum TE metric. The paths are independent; in particular, there is no diversity requirement.



Link	TE metric	Residual b/w Mbps
AB	1	10
AC	3	10
BC	1	10
BD	3	10
CD	1	10

Figure 7: Independent, Synchronized Computation

If the two paths were computed in a serialized, independent fashion, then the PCE server would compute the solution {AB, BC, CD} for the first path, with a globally minimized cumulative TE metric of 3. Unfortunately, the second path computation would then either fail or lead to a failure in LSP signaling, because all paths from A to D now contain at least one link with a residual bandwidth of zero. If, however, the two path computations were synchronized, then the PCE server could compute the solutions {AB, BD} and {AC, CD}.

14 Appendix B: List of companies belonging to OIF when document was approved

Acacia Communications
ADVA Optical Networking
Agilent Technologies
Alcatel-Lucent
Altera
AMCC
Amphenol Corp.
Anritsu
Applied Communication Sciences
AT&T
Avago Technologies Inc.
Broadcom
Brocade
Centellax, Inc.
China Telecom
Ciena Corporation
Cisco Systems
ClariPhy Communications
Comcast
Coriant
Cortina Systems
CPqD
CyOptics
Department of Defense

Deutsche Telekom
EigenLight.com
Emcore
Ericsson
ETRI
EXFO
FCI USA LLC
Fiberhome Technologies Group
Finisar Corporation
Fujitsu
Furukawa Electric Japan
GigOptix Inc.
Hewlett Packard
Hitachi
Hittite Microwave Corp
Huawei Technologies
IBM Corporation
Infinera
Inphi
Intel
IPtronics
JDSU
Juniper Networks
Kandou
KDDI R&D Laboratories
Kotura, Inc.
LeCroy
LSI Corporation
Luxtera
M/A-COM Technology Solutions
Marben Products
Metaswitch
Mindspeed
Mitsubishi Electric Corporation
Molex
MoSys, Inc.
MultiPhy Ltd
NEC
NeoPhotonics
NTT Corporation
Oclaro
Optoplex
Orange
PETRA
Picometrix
PMC Sierra
QLogic Corporation
Reflex Photonics
Semtech
Skorpios Technologies
Sumitomo Electric Industries
Sumitomo Osaka Cement
TE Connectivity
Tektronix
Tellabs
TELUS Communications, Inc.
TeraXion
Texas Instruments
Time Warner Cable
TriQuint Semiconductor
u2t Photonics AG
US Conec
Verizon
Xilinx
Xtera Communications
Yamaichi Electronics Ltd.