PACE: Next Steps in PAth Computation Element (PCE) Architectures: From Software-Defined Concepts to Standards, Interoperability and Deployment.

Coordination and Support Action - Funded by the European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreement no. 619712.

Strategic objective: ICT-2013.1.1 Future Networks
Start date of project: November 1, 2013 (24 months duration)

D2.3
Advisory "Handbook" for New EU Projects & PCE Standards "Cookbook"

Version 1

Due date: June 30, 2015

Submission date: July 29, 2015

Deliverable leader: ODC

Author list: D. King, A. Farrel (ODC)
F. Paolucci (SSSA)
T. Das (TUBS)
R. Casellas (CTTC)
O. Dios Gonzales (TID)
O. Dungeon, J. Meuric (ORANGE)

Dissemination Level

☒	PU:	Public
☐	PP:	Restricted to other programme participants (including the Commission Services)
☐	RE:	Restricted to a group specified by the consortium (including the Commission Services)
☐	CO:	Confidential, only for members of the consortium (including the Commission Services)

Abstract

The Path Computation Element has been established as a key component of the Software Defined Networking (SDN) architecture. The PACE (Next Steps for the Path Computation Element) project was initiated to continue European technical leadership of PCE technology and related components, provide a collaborative environment for PCE knowledge sharing, and help seed ideas and facilitate standardisation of PCE and related technologies in Standards Development Organisations (SDO).

This is an advisory handbook for new EU projects and provides the role of a Standards cookbook for the Path Computation Element (PCE) and related technologies. The document provides links and reference to various PCE and related technology primers. The document also outlines what are the existing PCE Standards and how to involve your project in PCE and related technology standardisation.

This document has been developed by the project partners using various PACE project deliverables from across the Work Pages and Tasks, as well as output from internal and external PACE project workshops. Where applicable, it cites the key SDO descriptions and references for PCE requirements, architectures, components and procedures.

The long-term intention for this document is to move it onto a publically available wiki for ongoing development and refresh by future users and inventors of PCE technologies.

Contents

Summary of Tables

1. PACE Introduction

A number of challenges exist in the relatively new area of Software Defined Networks (SDN). A key principle of SDN is centralised control of network resources. However, the Open Networking Foundation (ONF) specifications lack sophisticated control systems and well-defined interfaces with external (non-ONF) functions for path computation, traffic engineering, and network optimisation. Additionally, new initiatives, such as the Network Functions Virtualization (NFV) effort originating in the European Telecommunications Standards Institute (ETSI), are leveraging virtualization concepts for moving dedicated hardware functions into a virtualised environment, placing new requirements on network control and optimisation.

It has been identified that a key technology, the Path Computation Element (PCE), will provide a significant benefit to both SDN and NFV environments, and other new technology waves including the Internet of Things (IoT).

1.1 What is the PACE project?

PACE is an FP7 Coordination and Support Action project (Call 11) focusing on research, development, and standards in the broad area of Path Computation Element (PCE) - based architectures. This project has received funding from the European Union’s Seventh Framework Programme for research, technological development and demonstration under grant agreement no.619712.

PACE has been tasked with ensuring that the different aspects of PCE are not developed in isolation, while addressing interoperability issues, and thus avoiding any delays in innovation, and in sealing European leadership in the sector. PACE provides an education repository, community and support forum, open source, as well as sustained collaborative action within a concentrated community of industrial leaders, developers, and academics.

1.2 What are the PACE project application areas?

The PACE project promotes the idea of PCE as a widely applicable tool in the areas of:

- Core packet Internet
- Mobile backhaul networks
- Content Distribution Networks (CDN)
- Next generation optical transport networks
- Sensor networks, wireless mesh, and Internet of Things (IoT)

The following specific challenges and standardisation areas are of interest:

- Open Networking Foundation (ONF) - Design control systems and well-defined interfaces with external (non-ONF) functions, representing a not-to-be-missed opportunity for the third-party network control and management systems, such as PCE.
- Network Functions Virtualization (NFV) - ETSI initiative leveraging virtualization concepts, with the growing role of PCE-like concepts.
- Application-based Network Operations (ABNO) - An architecture proposed, developed and led within the Internet Engineering Task Force (IETF) by leading members of our consortium – uses a variety of PCE-based tools and techniques.

1.3 Where can I find out more about PACE?

There are two areas for information on PACE and EC-funded projects developing technologies around PCE:

PACE Website: http://www.ict-pace.net/

PACE Wiki: http://ict-one.eu/pace/public_wiki/mediawiki-1.19.7/index.php?title=Main_Page

2. The Path Computation Element

2.1 What is the PCE?

Path computation may be based on a set of constraints, such as hop count, delay, bandwidth and Quality of Service (QoS). The application of traffic engineering in MPLS and GMPLS networks utilises constraint-based path computation to optimise the use of resources.

Path computation objectives include: improve network efficiency, increase traffic performance, reduce costs, and increase profitability. Typically, path computation is performed by the connection head-end or by an Operational Support System (OSS). In large single-domain environments, or across multiple domains, end-to-end path computation becomes very complex as multiple constraints are applied and additional computationally resources are requested. As the demand for more complex path computation has increased, the need for a dedicated platform, or a controller, to perform path computations and to be adaptive to network changes.

The Path Computation Element (PCE) was developed as an entity capable of computing paths for a single or set of services. The PCE might be a network node, network management station, controller, or dedicated computational platform that is resource- and network-aware and has the ability to consider multiple constraints for much more complex path computation.

2.1.1 Standards for PCE

The main source of architectural and protocol standards related to PCE is the IETF. Typically, as in the usual IETF standard-building mechanism, drafts are submitted by single author or a group of authors. Then, after discussions, some of them are tagged as officially submitted by the Working Group and are subject to improvement steps until they are published as "Request For Comment (RFC) documents that are widely recognised as protocol standards.

In particular, the IETF's PCE Working Group, active since 2005, is responsible for coordinating the submission and discussion of proposals related to PCE and the PCE Protocol (PCEP).

2.2 What are the components of PCE?

The core of the PCE architecture and PCEP protocol is represented by RFC 4655 (PCE architecture) [1], RFC 4657 (PCEP general requirements) [2], and RFC 5440 (PCEP specification) [3]. In particular, the PCE general architecture described in RFC 4655 includes the following features:

- Definition of PCE
- Motivations of PCE
- Architectural options (composite PCE node, multiple PCE path computation, inter-PCE path computation)
- Architectural considerations (centralised and distributed models, stateless/stateful, Traffic Engineering Database (TED) synchronisation)

The general requirements of PCEP are described in RFC 4657 including the following topics:

- Client-server communication
- Requests/responses, cancellations, multiple requests, reliable message exchange, prioritisation
- Notifications, asynchronous communication
- Objective functions support
- Aliveness detection and recovery requirements

PCEP is formalised and descried in RFC 5440 and includes the following information:

- Architectural protocol overview, including handshake, keepalive, path computation request and reply, notification, errors, termination
- Transport protocol (running on TCP, port 4189)
- Definition of the main PCEP messages (common Header, main messages)
- Definition of the main PCEP objects (common header, main objects)

The main PCEP messages and objects defined in RFC 5440 are summarised in the following table (the Keepalive message is omitted since it contains only the common header). An 'X' indicates a mandatory object in a message while the use of square brackets '[X]' indicates an optional object.

Message Object	**Open**	**Close**	**PCReq**	**PCRep**	**PCNtf**	**PCErr**
OPEN	X					[x]
RP			X	X	[x]	[x]
NO-PATH				[x]		
END-POINTS			X			
BANDWIDTH			[x]	[x]		
METRIC			[x]	[x]		
ERO				X		
RRO			[x]			
LSPA			[x]	[x]		
IRO			[x]	[x]		
SVEC			[x]			
NOTIFICATION					X	
ERROR						X
LOAD-BALANCING			[x]			
CLOSE		X				

Table 1 PCEP core messages and objects from RFC 5440

The use of the Synchronization Vector (SVEC) object to enable synchronised dependent path computation is defined in RFC 6007 [4]. Standards have been defined to describe protocols manageability in RFC 6123 [5] and policy-enabled path computation in RFC 5394 [6].

2.2.1 General PCEP extensions

After the release of the first definition of PCEP, a number of extensions have been promoted as RFC to introduce additional path computation features:

- Diffserv TE support (RFC 5455) [7]
- Global Concurrent Optimization (GCO) (RFC 5557) [8]
- Monitoring (RFC 5886) [9]

- Objective Function (RFC 5541) [10]
- Point-to-MultiPoint path computation: requirements in RFC 5862 [11], application in RFC 5671 [12], extensions in RFC 6006 [12]
- Excluding Route Object support (RFC 5521) [13]
- PCEP vendor constraints (RFC 7470) [14]
- Encrypted ERO, path key (RFC 5520) [15]

The following table lists the extended objects introduced by the aforementioned standards and the hosted by PCEP messages. The same notation is used as in Table 1.

Message Object	**PCMonReq (RFC 5886)**	**PCMonRep (RFC 5886)**	**PCReq**	**PCRep**	**PCNtf**	**PCErr**
CLASSTYPE (RFC 5455)			[x]			
GLOBAL CONSTRAINT (RFC 5557)			[x]			
MONITORING (RFC 5886)	X	X	[x]	[x]		
PCC-ID-REQ (RFC 5886)	X		[x]			
PCE-ID (RFC 5886)	[x]		[x]			
PROC-TIME (RFC 5886)		[x]		[x]		
OVERLOAD (RFC 5886)		[x]		[x]		
OF (RFC 5541)			[x]	[x]		
P2MP END-POINTS (RFC 6006)			[x]			
UNREACH-DESTINATION (RFC 6006)				[x]		
Secondary ERO (RFC 6006)				[x]		
Secondary RRO (RFC 6006)			[x]	[x]		
Branch Node Capability			[x]			
XRO (RFC 5521)			[x]	[x]		
VENDOR_INFORMATION (RFC 7470)			[x]	[x]		

Table 2 General PCEP extensions and objects

Extensions to other protocols related to PCE/PCEP operation and functions are listed below:

- Management Information Base (MIB) support (RFC 7420) [16]
- PCE Discovery: requirements in RFC 4674 [17], OSPF extensions in RFC 5088 [18], IS-IS extensions in RFC 5089 [19]

2.2.2 Multi-layer architectures

In multi-layer networks, path computation has to be enabled by considering the different nature (e.g., switching capability, granularity) of the layers composing the network. In this regard, the general inter-layer path computation framework is provided in RFC 5623 [21]. In particular the following topics are defined and discussed:

- Inter-layer path computation models (single PCE, multiple PCE)
- Virtual Network Topology (VNT) management through a Virtual Network Topology Manager (VNTM) component
- Inter-layer path control models (PCE with VNTM; higher-layer signalling triggers; Network Management System (NMS) with VNTM)

The inter-layer requirements are reported in RFC 6457 [22], including the following topics:

- Control of inter-layer path computation
- Communication of inter-layer constraints
- Adaptation capabilities
- Cooperation between PCEs
- Inter-layer diverse paths
- Capabilities advertisements for PCE discovery

The PCEP extensions providing proper inter-layer path computation function are not standardised yet, being reported in the Internet-Draft draft-ietf-pce-inter-layer-ext that is work in progress for the IETF's PCE Working Group. Proposed extended objects include INTER-LAYER, SWITCH-LAYER, REQ-ADAP-CAP, and new inter-layer metrics (see the relationships with PCEP messages in the table below):

	PCMonReq (RFC 5886)	**PCMonRep (RFC 5886)**	**PCReq**	**PCRep**	**PCNtf**	**PCErr**
INTER-LAYER			[x]	[x]		
SWITCH-LAYER			[x]	[x]		
REQ-ADAP-CAP			[x]	[x]		

Table 3 PCEP extensions proposed for inter-layer path computation

2.2.3 Multi-area and multi-domain architectures

In multi-area and multi-domain scenarios path computation mechanisms are more complex. This is due to the fact that, if each domain or area is controlled by a PCE, inter-PCE path computation procedures have to be deployed and more coordination is required. This leads to additional protocol requirements and extensions.

For multi-area networks, the requirements specified for proper path computation are reported in RFC 4927 [23] and include the following aspects:

- Control and recording of area crossing
- Strict explicit and loose path

- PCE list enforcement and recording
- Inclusion of area-id in requests
- Area inclusion/exclusion
- Inter-area diverse computation and policies
- Loop avoidance

In multi-domain networks, if inter-PCE path computation is performed, two architectural schemes based on PCE are defined: the Backward Recursive PCE-based Computation (BRPC), detailed in RFC 5441 [31], and the Hierarchical PCE, detailed in RFC 6805 [24]. In BRPC, each domain is controlled by a PCE and an inter-PCE computation chain that is established among the selected domain sequence (which is assumed to be known in advance). In H-PCE, each domain is controlled by a child PCE and inter-domain computation is performed by communicating with a parent PCE that has access to the multi-domain TED. BRPC requires a minimum set of PCEP extensions (flags and some TLVs). H-PCE needs a number of PCEP extensions, not yet standardised, that are set out in the Internet-Draft draft-ietf-pce-hierarchy-extensions [27] that is working progress in the IETF's PCE Working Group.

2.2.4 PCE for GMPLS, WSON, and Flex-grid

In the particular context of GMPLS networks, path computation and related protocol attributes have to be extended accordingly. Lightpath computation requirements in GMPLS networks are detailed in RFC 7025 [25], including:

- Switching capabilities check
- Unnumbered interfaces
- Asymmetric bandwidth path computation
- Label constraints

In the context of GMPLS, Wavelength Switched Optical Networks (WSON) require the definition of extended path computation that also includes wavelength assignment. Abbreviations are defined for Routing (R) and Wavelength Assignment (WA) and may be combined to indicate different combinations and phasing of computation. Requirements specified in RFC 7449 [26] discuss the following aspects:

- Path computation type option (RWA, R+WA, R-candidate+WA)
- Bulk RWA
- RWA re-optimisation
- Wavelength range constraints and preferences

Extensions to PCEP enabling lightpath computation are being developed in the Internet-Draft draft-ietf-pce-gmpls-pcep-extensions [28] that is work in progress for the IETF's PCE working group and that details extensions for the following existing objects:

- Generalized END-POINTS
- Generalized BANDWIDTH
- Generalized LOAD-BALANCING
- ERO/IRO/XRO Label subobject
- Extended LSPA and NO-PATH

2.2.5 Stateful PCE and instantiation capabilities

The stateful approach proposed for the PCE enables, on the one hand, more powerful path computation, on the other hand, increased control on path status report and modification. Moreover, instantiation capabilities provide PCE with capabilities typical of an integrated provisioning interface.

The extensions required for the stateful support are reported in the Internet-Draft draft-ietf-pce-stateful-pce [29] that is work in progress in the IETF's PCE working group. Two novel messages are defined, the PCReport and the PCUpdate messages. The PCReport is used to report the status of a Label Switched Path (LSP) (periodic report, report after synchronisation, report after action), while the PCUpdate message is used to trigger the modification of the attributes of an LSP (e.g., route). Two main objects are defined: the SRP object, equivalent to RP but used for stateful operation, and the LSP object, providing unique identification of the LSP across data plane and signalling protocols. Additional extensions (e.g., error reasons, stateful capabilities in the Open message, flags) are also included.

Instantiation capabilities are a further option and create an active stateful PCE. These capabilities are detailed in the Internet-Draft draft-ietf-pce-pce-initiated-lsp document [30] that is a work in progress for the IETF's PCE working group. This document introduces a novel PCEP message, called the Initiate message, and novel flags and TLVs to handle setup and delete requests. The table below summarises the relationships between mandatory and optional novel objects and messages enabling active stateful PCE and instantiation. It is worth noting that other existing PCEP objects may also be used in the novel messages (for example, in the PCInitiate, the ENDPOINTS object is mandatory).

Message Object	**PCReport**	**PCUpdate**	**PCInitiate**
SRP	X	X	X
LSP	X	X	X

Table 4 PCEP messages and objects enabling stateful capabilities

2.2.6 PCE in Application-Based Network Operations (ABNO)

One of the most recent IETF architectures incorporating the PCE as a key module is Application Based Network Operation (ABNO), defined in RFC 7491 [33].

A modular control plane architecture automating application-driven provisioning and proactive monitoring is envisioned. The architecture provides all the required instruments to handle policies, admission, pre-emption, routing, path computation, signalling and network operation, and administration maintenance (OAM).

The PCE is a key component of the ABNO architecture. In the ABNO architecture, PCE can be connected to the ABNO Controller, to a VNTM (in a multi-layer scenario), with databases (TED and LSP-DB), with the Provisioning Interface, and with Client and Server Network Layers. PCEP can be used between the ABNO Controller and PCE, between the PCEs in Client and Server Network Layers, as well as with the Provisioning Interface. Furthermore, PCEP is used between the VNTM (acting as PCC) and the PCE. LSP path computation/modification is based on requests coming either from the ABNO Controller (connected via a northbound interface to the Application Layer) or directly from network nodes. The active stateful functionality with instantiation is enabled through connection with the Provisioning Interface.

The ABNO RFC describes a number of use cases of ABNO. In particular the following use cases are related to the use of PCE within the architecture:

- Inter-AS connectivity through H-PCE (ABNO Controller over parent PCE), including active provisioning procedures
- Multi-layer networking with VNTM (ABNO Controller over packet-layer PCE, packet-layer PCE over VNTM, VNTM over optical-layer PCE)
- Data centre interconnection (ABNO Controller over PCE, PCE over multi-protocol provisioning manager)
- Make-before-break procedures (through OAM Handler and PCE)
- Global Concurrent Optimization (ABNO Controller over GCO-capable PCE)
- Cross-stratum optimisation

2.3 Where can I find out about the PCE?

For a fundamental grounding in PCE and related technologies, please refer to our PACE PCE primer:

www.ict-pace.net/files/3313/8929/2782/PCE_Primer.pdf

The PCE Primer is split into the following sections:

- PCE Definitions and Main Drivers
- Basics of the PCE Architecture
- PCE Protocol (PCEP)
- Path Computation in Wavelength/Spectrum Switched Optical Networks
- PCE and Quality of Transmission (QoT)
- PCE for Flexible Grid Spectrum Switched Optical Networks
- PCE for Restoration
- Multi-domain Path Computation
- Backwards Recursive Path Computation
- Hierarchical PCE
- Path Computation in Multi-Layer Networks
- The Traffic Engineering Database
- Stateful PCE
- PCE deployment models in Software Defined Networks

2.4 Where was PCE developed?

The PCE was developed within the Internet Engineering Task Force (IETF).

> *"The mission of the IETF is to make the Internet work better by producing high quality, relevant technical documents that influence the way people design, use, and manage the Internet."*
> *www.ietf.org*

Newcomers to the IETF should start here: www.ietf.org/newcomers.html

The PCE Working Group (WG) website is located at datatracker.ietf.org/wg/pce/charter/. The PCE Working Group is chartered to specify the required protocols so as to enable a Path Computation Element (PCE)-based architecture for the computation of paths for MPLS and GMPLS Point to Point and Point to Multi-point Traffic Engineered LSPs.

In this architecture path computation does not necessarily occur on the head-end (ingress) Label Switching Router (LSR), but on some other path computation entity that may physically not be located on each head-end LSR.

The PCE WG works on applications of this model within a single domain or within a group of domains (where a domain is a layer, IGP area or Autonomous System with limited visibility from the head-end

LSR). At this time, applying this model to large groups of domains such as the Internet is not thought to be possible, and the PCE WG does not spend energy on that topic.

The WG specifies the PCE communication Protocol (PCEP) and needed extensions for communication between LSRs (termed Path Computation Clients - PCCs) and PCEs, and between cooperating PCEs. Security mechanisms such as authentication and confidentiality are included.

The WG determines requirements for extensions to existing routing and signalling protocols in support of the PCE architecture and the signalling of inter-domain paths (e.g., RSVP-TE and its GMPLS variations). Any necessary extensions will be produced in collaboration with the Working Groups responsible for the protocols.

The WG also works on the mechanisms to for multi-layer path computation and PCEP extensions for communication between several network layers, and for a variety of forwarding technologies.

2.5 What about other standards for general Path Computation?

2.5.1 ITU-T and OIF

The main path computation functions and architectures standardised by the ITU-T are reported in the G.7715.2 Recommendation [34] in the context of the Automatically Switched Optical Network (ASON) routing paradigms. The functional element responsible for routing and path computation is referred to as Routing Controller (RC). The document describes the requirements and the general architectures of remote routing queries, but does not specify any protocol solution. Routing algorithms are beyond the scope of the document.

Path computation may be performed by a single RC if the requested connection remains inside the same Routing Area (RA) controlled by the one Controller. Otherwise, end-to-end path computation may be performed in multiple steps, and cooperative path computation may use multiple Controllers. RCs may be federated and provide cooperative path computation by using remote query operations. Different functional procedures are used:

- Step-by-step remote computation
- Hierarchical remote computation
- Combination of step-by-step and hierarchical

Step-by-step remote computation enables segmented path computation iteratively computed by a chain of RCs.

Hierarchical remote computation identifies a child RC for each RA, and a parent RC, placed at the next higher level of the ASON hierarchy. The description of this architecture is general and recursive in the sense that many hierarchy layers may exist and the same RC may act as child RC for one layer and parent RC for the lower layer.

The combination of the two aforementioned procedures include a number of routing federations in which a step-by-step approach is deployed for RCs of the same hierarchy acting as parent RC for a single routing federation. Inside each routing federation, the parent RC implements hierarchical computation with lower layer child RCs responsible of different segments (RAs) of the routing federation.

Additional RC requirements related to path computation are detailed and listed below:

- Discovery mechanisms (address and name): static or protocol -based
- Capabilities attributes advertised (e.g., supported path computation algorithms, constraints, policy-based
- Route request and response messages

- Routing adjacencies between routing controllers
- Communication channel over data communication network

The Optical Internetworking Forum (OIF) is the main source of interworking agreements for optical equipment vendors. It specialises in documenting how to interpret and apply standards specifications that pertain to optical networks and covers architecture, data plane specifications, and control plane standards. The OIF adopts many of the architectural and data plane concepts standardised by the ITU-T and applies the control plane standards from the IETF to describe profiles of the technology space and to describe interoperable deployments.

2.5.2 Correlation and Overlap with IETF

The PPCE defined by the IETF may be associated to a RC implementing path computation algorithms and exchanging route query/response messages with another RC.

The main correlations and overlap of the two standardisation proposal are described below:

1. Step-by-step remote computation is strictly correlated and fully compliant to the BRPC mechanism detailed in RFC 5441 [31]. In particular, two main aspects are completely overlapping: a) the backward nature of the mechanism, and b) the transparent chain-based mechanism, where information exchanged by two controllers (i.e., PCEs) is transparent to the other controllers involved in the chain. In addition, route information exchanged between two controllers may be encrypted to preserve confidentiality and IETF provides the path-key mechanism to address such requirement in RFC 5520 [15].
2. Hierarchical remote computation is strictly correlated to H-PCE architecture defined in RFC 6805 [24]. Parent and child attributes and roles are practically the same and overlap is significant.
3. The route query interface is correlated with PCEP. Route queries from connection controllers and from other RCs equate to the relationship between PCC-PCE and between PCE-PCE. Single/multiple controller path computation corresponds with single and multiple PCE-based path computation described in RFC 4655 [1].

2.5.3 Open Networking Foundation (ONF)

The main source of standardisation in Software Defined Networking (SDN) is the Open Network Foundation (ONF). Path computation is one of the main functions designed and implemented within the SDN Controller, the main centralised element implementing control functions within the architecture.

As reported in the ONF SDN Architecture document [35], path computation is an optional internal function of the SDN Controller. However, it may be also considered as an independent application. More in general, the path computation engine in SDN may exist in three different configurations:

- As internal and private component of the Controller
- As internal component of the Controller, but exposed as service to be invoked by external users (i.e., applications)
- As an application that can be utilised by a Controller

Following these options, the path computation protocol may be used:

- As a communication protocol between internal modules of the Controller, with the alternative of a direct path computation function call
- As a communication protocol between Controller and application (Controller acts as PCE)
- As a communication protocol between application and Controller (Controller acts as PCC)

The ONF's SDN architecture does not specify the internal design of an SDN Controller, and in particular does not specify specific architectures and procedures for path computation. This suggests that path computation, that is an optional procedure of the controller, may be adopted in the general SDN framework inside the Controller (internal function) or as specific application which an SDN Controller may resort to.

However, when administrative domains are considered, the ONF document specifies the possibility of communication among SDN Controllers. For example, a client Controller (or a client application) may orchestrate a number of server Controllers. This principle is called "common controller coordination" and relies on the logical centralisation (and, partially, on hierarchy) of the Controller. Furthermore, Controller-to-Controller coordination is recommended when a service traverses multiple data plane networks (possibly of different technologies). In this case, a peer-to-peer relationship is established. The ONF document states that existing protocols may be used to address peer-to-peer Controller coordination and use case requirements do not need additional ad-hoc protocols. In this regard, in the context of inter-domain path computation, PCEP is explicitly mentioned as a possible interface between Controllers, and BRPC is a compatible peer-to-peer path computation mechanism.

2.6 What Standards exist for the PCE?

A full list of IETF PCE standards with direct links to the documents is available at:

ict-one.eu/pace/public_wiki/mediawiki-1.19.7/index.php?title=Main_Page#IETF_RFCs

The IETF maintains a list of relevant RFCs and Internet-Drafts that are work progress for the PCE working group at:

https://datatracker.ietf.org/documents/pce

All current PCE standards are outlined in the sections below and reference from the IETF repository:

2.6.1 PCE Frameworks, Architectures & Requirements

"A Path Computation Element (PCE)-Based Architecture" RFC 4655 [1]

This document specifies the architecture for a Path Computation Element (PCE)-based model to address this problem space. https://tools.ietf.org/html/rfc4655

"Path Computation Element (PCE) Communication Protocol Generic Requirements" RFC 4657 [2]

This document specifies generic requirements for a communication protocol between PCCs and PCEs, and also between PCEs where cooperation between PCEs is desirable. Subsequent documents will specify application-specific requirements for the PCE communication protocol. https://tools.ietf.org/html/rfc4657

"Path Computation Element Communication Protocol (PCECP) Specific Requirements for Inter-Area MPLS and GMPLS Traffic Engineering" RFC 4927 [23]

This document lists a detailed set of PCECP-specific requirements for support of inter-area TE-LSP path computation. It complements the generic requirements for a PCE Communication Protocol. https://tools.ietf.org/html/rfc4927

"Inter-AS Requirements for the Path Computation Element Communication Protocol (PCECP)" RFC 5376 [23]

Generic requirements for the PCECP are set out in "Path Computation Element (PCE) Communication Protocol Generic Requirements", RFC 4657. This document extends those requirements to cover the use of PCECP in support of inter-AS MPLS TE. https://tools.ietf.org/html/rfc5376

"Framework for PCE-Based Inter-Layer MPLS and GMPLS Traffic Engineering" RFC 5623 [21]

This document describes a framework for applying the PCE-based architecture to inter-layer Multiprotocol Label Switching (MPLS) and Generalized MPLS (GMPLS) traffic engineering. It provides suggestions for the deployment of PCE in support of multi-layer networks. This document also describes network models where PCE performs inter-layer traffic engineering, and the relationship between PCE and a functional component called the Virtual Network Topology Manager (VNTM). https://tools.ietf.org/html/rfc5623

"Path Computation Clients (PCC) - Path Computation Element (PCE) Requirements for Point-to-Multipoint MPLS-TE" RFC 5862 [11]

This document complements the generic requirements and presents a detailed set of PCC-PCE communication protocol requirements for point-to-multipoint MPLS/GMPLS traffic engineering. https://tools.ietf.org/html/rfc5862

"The Application of the Path Computation Element Architecture to the Determination of a Sequence of Domains in MPLS and GMPLS" RFC 6805 [24]

This document examines techniques to establish the optimum path when the sequence of domains is not known in advance. The document shows how the PCE architecture can be extended to allow the optimum sequence of domains to be selected, and the optimum end-to-end path to be derived through the use of a hierarchical relationship between domains. https://tools.ietf.org/html/rfc6805

"Requirements for GMPLS Applications of PCE" RFC 7025 [25]

The initial effort of the PCE (Path Computation Element) WG focused mainly on MPLS. As a next step, this document describes functional requirements for GMPLS applications of PCE. https://tools.ietf.org/html/rfc7025

"PCC-PCE Communication and PCE Discovery Requirements for Inter-Layer Traffic Engineering" RFC 6457 [22]

This document complements the generic requirements and presents detailed sets of PCC-PCE communication protocol requirements and PCE discovery protocol requirements for inter-layer traffic engineering. https://tools.ietf.org/html/rfc6457

"Conveying Vendor-Specific Constraints in the Path Computation Element Communication Protocol" RFC 7150 [22]

This document defines a facility to carry vendor-specific information in PCEP using a dedicated object and a new Type-Length-Variable that can be carried in any existing PCEP object. https://tools.ietf.org/html/rfc7150

"PCE-Based Computation Procedure to Compute Shortest Constrained Point-to-Multipoint (P2MP) Inter-Domain Traffic Engineering Label Switched Paths" RFC 7334 [22]

This document describes an experiment to provide procedures and extensions to the PCE Communication Protocol (PCEP) for the computation of inter-domain paths for P2MP TE LSPs. https://tools.ietf.org/html/rfc7334

"Unanswered Questions in the Path Computation Element Architecture" RFC 7399 [22]

This document does not update the PCE architecture documents and does not define how PCE protocols or components must be used. It does, however, suggest how the architectural components might be combined to provide advanced PCE function. It identifies the key gaps, questions and discusses them in the PCE architectural context with reference to other PCE architectural components, existing protocols, and recent IETF work efforts. https://tools.ietf.org/html/rfc7399

"A Backward-Recursive PCE-Based Computation (BRPC) Procedure to Compute Shortest Constrained Inter-Domain Traffic Engineering Label Switched Paths" RFC 5441 [31]

This document specifies a procedure relying on the use of multiple Path Computation Elements (PCEs) to compute such inter-domain shortest constrained paths across a predetermined sequence of domains, using a backward-recursive path computation technique. https://tools.ietf.org/html/rfc5441

2.6.2 PCE Protocol & Extensions

"Requirements for Path Computation Element (PCE) Discovery" RFC 4674 [17]

This document presents a set of requirements for a Path Computation Element (PCE) discovery mechanism that allows a Path Computation Client (PCC) to discover dynamically and automatically a set of PCEs along with certain information relevant for PCE selection. https://tools.ietf.org/html/rfc4674

"OSPF Protocol Extensions for Path Computation Element (PCE) Discovery" RFC 5088 [18]

This document defines extensions to the Open Shortest Path First (OSPF) routing protocol for the advertisement of PCE Discovery information within an OSPF area or within the entire OSPF routing domain. https://tools.ietf.org/html/rfc5088

"IS-IS Protocol Extensions for Path Computation Element (PCE) Discovery" RFC 5089 [19]

This document defines extensions to the Intermediate System to Intermediate System (IS-IS) routing protocol for the advertisement of PCE Discovery information within an IS-IS area or within the entire IS-IS routing domain. https://tools.ietf.org/html/rfc5089

"Path Computation Element (PCE) Communication Protocol (PCEP)" RFC 5440 [3]

This document specifies the Path Computation Element (PCE) Communication Protocol (PCEP) for communications between a Path Computation Client (PCC) and a PCE, or between two PCEs. https://tools.ietf.org/html/rfc5440.

"Diffserv-Aware Class-Type Object for the Path Computation Element Communication Protocol" RFC 5455 [7]

This document specifies a CLASSTYPE object to support Diffserv-Aware Traffic Engineering (DS-TE) where path computation is performed with the aid of a Path Computation Element (PCE). https://tools.ietf.org/html/rfc5455

"Preserving Topology Confidentiality in Inter-Domain Path Computation Using a Path-Key-Based Mechanism" RFC 5520 [15]

Where the TE LSP crosses multiple domains, such as Autonomous Systems (ASes), the path may be computed by multiple PCEs that cooperate, with each responsible for computing a segment of the path. However, in some cases (e.g., when ASes are administered by separate Service Providers), it would break confidentiality rules for a PCE to supply a path segment to a PCE in another domain, thus disclosing AS-internal topology information. This issue may be circumvented by returning a loose hop and by invoking a new path computation from the domain boundary Label Switching Router (LSR) during TE LSP setup as the signalling message enters the second domain, but this technique has several issues including the problem of maintaining path diversity.

This document defines a mechanism that allows a segment of a path within one domain to be represented by a key within another domain. The key can be "unlocked" when the signalling message reaches the domain in which the key was computed. https://tools.ietf.org/html/rfc5520

"Extensions to the Path Computation Element Communication Protocol (PCEP) for Route Exclusions" RFC 5521 [13]

This document presents PCEP extensions for route exclusions. https://tools.ietf.org/html/rfc5521

"Encoding of Objective Functions in the Path Computation Element Communication Protocol (PCEP)" RFC 5541 [10]

This document defines extensions to the PCE communication Protocol (PCEP) to allow a PCE to indicate the set of objective functions it supports. Extensions are also defined so that a PCC can indicate in a path computation request the required objective function, and a PCE can report in a path computation reply the objective function that was used for path computation. This document defines objective function code types for six objective functions previously listed in the PCE requirements work, and provides the definition of four new metric types that apply to a set of synchronised requests. https://tools.ietf.org/html/rfc5541

"Extensions to the Path Computation Element Communication Protocol (PCEP) for Point-to-Multipoint Traffic Engineering Label Switched Paths" RFC 6006 [12]

This document describes extensions to the PCE communication Protocol (PCEP) to handle requests and responses for the computation of paths for point-to-multipoint (P2MP) TE LSPs. https://tools.ietf.org/html/rfc6006

"Use of the Synchronization VECtor (SVEC) List for Synchronized Dependent Path Computations" RFC 6007 [4]

This informational document clarifies the use of the SVEC list for synchronised path computations when computing dependent requests. The document also describes a number of usage scenarios for SVEC lists within single-domain and multi-domain environments. https://tools.ietf.org/html/rfc6007

2.6.3 PCE Applicability Statements

"Applicability of the Path Computation Element (PCE) to Point-to-Multipoint (P2MP) MPLS and GMPLS Traffic Engineering (TE)" RFC 5671 [12]

This document examines the applicability of PCE to path computation for P2MP TE LSPs in MPLS and GMPLS networks. It describes the motivation for using a PCE to compute these paths and examines which of the PCE architectural models are appropriate. https://tools.ietf.org/html/rfc5671

2.6.4 PCE Applications

"Path Computation Element Communication Protocol (PCEP) Requirements and Protocol Extensions in Support of Global Concurrent Optimization" RFC 5557 [8]

This document provides application-specific requirements and the path computations from Path Computation Elements (PCEP) extensions in support of Global Concurrent Optimization (GCO) applications. https://tools.ietf.org/html/rfc5557

2.6.5 PCE Management & Policy

"Policy-Enabled Path Computation Framework" RFC 5394 [6]

This document introduces the use of the Policy Core Information Model (PCIM) as a framework for supporting path computation policy. This document also provides representative scenarios for the support of PCE Policy. https://tools.ietf.org/html/rfc5394

"A Set of Monitoring Tools for Path Computation Element (PCE)-Based Architecture" RFC 5886 [9]

In PCE-based environments, it is critical to monitor the state of the path computation chain for troubleshooting and performance monitoring purposes: liveness of each element (PCE) involved in the PCE chain and detection of potential resource contention states and statistics in terms of path computation times are examples of such metrics of interest. This document specifies procedures and extensions to the Path Computation Element Protocol (PCEP) in order to gather such information. https://tools.ietf.org/html/rfc5886

"An Inclusion of Manageability Sections in Path Computation Element (PCE) Working Group Drafts" RFC 6123 [5]

Previously, the PCE Working Group used the recommendations contained in this document to guide authors of Internet-Drafts on the contents of "Manageability Considerations" sections in their work. This document is retained for historic reference. https://tools.ietf.org/html/rfc6123

2.7 What are Application-Based Network Operations (ABNO)?

One of the most recent IETF architectures incorporating PCE as a key module is the Application Based Network Operation (ABNO), defined in [33].

ABNO is a modular control plane architecture automating application-driven provisioning and proactive monitoring, as well as providing all the required instruments to handle policies, admission, pre-emption, routing, path computation, signalling and network operation, and administration maintenance (OAM) operation. PCE is a key component of the ABNO architecture. In the ABNO architecture, PCE can be connected to the ABNO Controller, to a VNTM (in multi-layer scenario), with databases (TED and LSP-DB), to the Provisioning Interface and with Client and Server Network Layers. PCEP can be used between the ABNO Controller and PCE, between PCE and Client and Server Network Layers, as well as at the Provisioning Interface. Furthermore, PCEP is used between the VNTM (acting as PCC) and PCE. LSP path computation/modification are based on requests coming either from the ABNO Controller (connected via a northbound interface to the Application Layer) or directly from network nodes. The active stateful functionality with instantiation is enabled through connection at the Provisioning Interface.

The ABNO RFC specifies and describes a number of use cases for the application of ABNO, in particular the following related to PCE architectures:

- Inter-AS connectivity through H-PCE (ABNO Controller over parent PCE), including active provisioning procedures
- Multi-layer networking with VNTM (ABNO Controller over packet-layer PCE, packet-layer PCE over VNTM, VNTM over optical-layer PCE)
- Data centre interconnection (ABNO Controller over PCE, PCE over multi-protocol provisioning manager)
- Make-before-break procedures (through OAM Handler and PCE)
- Global Concurrent Optimization (ABNO Controller over GCO-capable PCE)
- Cross-stratum optimisation

2.8 Open Source Projects using PCE

This section briefly high-lights two of the main Open Source projects that include PCE. Participation in Open Source projects is generally easy with the main requirement being to write code cooperatively and in line with the spirit of the project. However, precise rules (especially with respect to code ownership and IPR) apply differently in each project and should be checked before participation.

2.8.1 Open Daylight

Open Daylight (ODL) [43] is the most popular Open Source platform for SDN and NFV control. Under the auspices of the Linux Foundation, the ODL project maintains a collection of components, including a SDN Controller.

ODL has a BGP and a PCEP library that enables the controller to talk to a wide variety of elements (and not just OpenFlow devices). BGP is used as a source of topology information. ODL PCEP is built based on Yang Models describing the protocol.

Some tests have been conducted at Orange Labs on the conformity of PCEP and BGP-LS standard. For that purpose, an Open Daylight platform was setup in front of a various IP routers coming from Juniper, Cisco and ALU vendors. Regarding the topology acquisition, Open Daylight get in real-time the network topology sent by Juniper or Cisco routers configured as BGP-LS Speaker. Only the BGP-LS Listener mode of Open Daylight has been tested. However, Open Daylight DLux UI interface don't support directly the BGP-LS topology acquisition in order to show a graph of the network. To check that the topology is correctly learnt, you could:

- Use the REST API to get the topology and verify manually that the network topology is correct,
- Use the Yang UI of DLux to browse the topology model. In this mode, you could show some graph of the network, but not in a useful and user friendly mode.

The second test concern the PCEP protocol itself. If establish a PCEP session between a router and Open Daylight is not a problem, playing with it is not really possible. In fact, Open Daylight just implement the PCEP message prototype, but not all state machine behind in order to correctly process the various messages. For example, Open Daylight doesn't support PCReq / PCRep message as per RFC5440. A Wireshark capture shows that the message is correctly sent to Open Daylight by the route acting as a PCC, but there is no answer. Same behaviour occurs when PCC sent a PCRpt message when it would delegate the path computation. Finally, Open Daylight just implement the PCE Initiate mode, but without any path computation. Indeed, to setup a tunnel, you need to use the REST API and provide all information including the ERO. Browsing the code confirms that no path computation algorithm is provided.

We found that Open Daylight is just a translator of BGP-LS and PCEP protocols into REST API and vice-versa, but, in any case could be tagged as a runnable PCE. To achieve this later, you need to enhance the code with messages processing and path computation algorithm, or use the REST API to implement them externally if you don't want to understand the Open Daylight code. Note that as the code is generated (through Open Daylight tools parsing YANG model), it is very hard to understand and follow it in order to add your own piece of code.

In summary, Open Daylight is a base to build a complete SDN PCE based system. Such platform is provided by vendors such as Cisco, Brocade and Ericsson which used Open Daylight as the basis of their respective commercial solution.

2.8.2 ONOS

ONOS [40] is an Open Source SDN-based network operating system that has been focused on creating northbound and southbound abstractions to control OpenFlow devices [41]. In the case of IP-Optical scenarios, ONOS maintains a global network graph, which is used by its Path Computation Engine (PCE) to find available capacity [42]. The operating system is also able, by means of the southbound interfaces, to create the computed capacity on demand. The PCE is also used in case of failures or problems in the network.

The ONOS PCE is multi-layer, constructing a graph per-layer and the correspondence among them, which allows maintenance of the global network graph. ONOS has also been adapted for Segment Routing (SR). The architecture for SR, at its current stage is based on three modules, a Routing Service, a Discovery Service, and a Forwarding Service, that cooperate to process the service requests.

3 Contributing to PCE Development

3.1 How to incorporate standardisation in to European Projects

Contribution to standardisation is an important outcome for a European Project in order to advance the state of the art, have a measurable impact on industry, and show leadership in the relevant technology sector.

Furthermore, standardisation is a valuable step in ensuring that the effort expended on the project has lasting results both in a public record of the work done and, more importantly, in making sure that future development in the space deriving from the project is interoperable and fully flexible in the marketplace.

Although the standardisation process does not take a large amount of effort, it can take a long time to produce standards. This is partly because of the processes involved, but also happens because the methods used are collaborative and demand wide discussion.

The natural approach is to wait until the project has developed and stabilised the protocols that it will use before starting work on standardisation. While this appears to be a sound way to proceed it misses the key benefit of a standardisation process which is the input from other participants in the standards body at a time in the project when it would be most use. Furthermore, this level of delay is likely to mean that the full standardisation process is not completed before the end of the project resulting either in unsatisfactory outcomes for the project or the requirement for additional effort by the project consortium members after the end of the project.

Therefore, a project aiming for standardisation outcomes needs to become active in the relevant standards bodies as early in the project as is feasible. It is recommended that projects plan their intended standardisation outcomes at the time of the project proposal and that they start work on standardisation on day one of the project. It needs to be understood that initial input to standardisation does not need to be and is not expected to be mature and complete. It is perfectly acceptable for early standardisation from a project to be sketchy and preliminary: this will enable early feedback from others in the industry, validation of the approach being used, and advance towards published standards.

3.2 Participating in Standards-based PCE

3.2.1 Contributing to the IETF PCE Working Group

Participation in the IETF is open and free. There is no membership fee or sign-up process, and anyone can contribute. The main mode of participation is via mailing lists: there is one mailing list for each working group where all topics relevant to the working group are discussed.

The working groups proceed by developing documents that will be progressed towards becoming standards, or as informational or experimental records. All work in progress and all published documents are freely downloadable from the IETF web site.

Within the IETF standards are published as a Request for Comment (RFC) documents. Each potential RFC begins life as an Internet Draft (I-D). The procedure in submitting, developing, adopting and then finally publishing an IETF document is as follows:

1. Submit an individual 00 draft which is published as an Internet Draft (I-D).
2. Receive comments and contributions from partners and working group participants.
3. Update the draft and based on feedback, new insights, and ongoing PCE activity contributed through the working group mailing list.
4. Present the I-D at a working group session.
5. Repeat steps 2-3-4 until the draft is efficiently discussed, and adopted by the working group.

6. Once the draft reaches sufficient maturity and all open issues have been addressed, the document can be sent for working group Last Call (LC) and any final issues raised are addressed.
7. The draft is then submitted for publication, and processing is handed over to the Internet Engineering Steering Group (IESG).
8. The IESG carries out its own reviews and sends the draft for an IETF-wide Last Call. Further updates may be needed to address additional comments.
9. If the IESG approves the draft to become an RFC it is advanced through the publication process as an RFC.

European projects are encouraged to become directly involved and contribute to the PCE Working Group and to include acknowledgement of their funding source within the Acknowledgements section of any I-Ds they author or to which they contribute.

Participants should make themselves aware of the IETF's rules with respect to IPR. All active contributors to the IETF process (that is, people who contribute to I-Ds or to the development process by discussing the work in an IETF forum) are expected to disclose any relevant IPR about which they are aware. They may disclose any appropriate licensing terms, but failure to disclose in a timely manner will be frowned on by the IETF and may impact the future ability to enforce the IPR.

3.2.2 Contribution to the ITU-T

Participation in the ITU-T generally requires association with a member organisation that pays fees to contribute to the standards process. However, there are other participation avenues through academic bodies and by direct invitation as an expert in a particular technology.

While the ITU-T's standards (Recommendations) are available for free download from the ITU-T's web site, work in progress is normally held in confidence by the ITU-T study groups. Under some circumstances, the ITU-T may share work in progress with other standards bodies, groups, or individuals in order to get reviews, advice, or input.

The ITU-T develops its standards on mailing lists and at face-to-face meetings. Access to these venues is also limited to member organisations and invited participants.

3.3 IETF Working Groups Relevant to PCE

Within the IETF, the PCE working group is the main source of PCE-related standards. However, there is a set of working groups in the Routing Area directly related to PCE, mainly being consumers of PCE outcomes.

TEAS

The Traffic Engineering Architecture and Signaling (TEAS) working group is responsible for architectures for traffic engineering and for generic signalling protocol extensions for use traffic engineered networks.

The architectures in TEAS recognise PCE as a fundamental component for determining paths across traffic engineered networks.

MPLS

The Multiprotocol Label Switching (MPLS) working group has responsibility for the forwarding plane architecture of MPLS networks and for the control plane protocols used to establish LSPs in those networks. Part of this extends to connection-oriented MPLS where PCE is seen as a key component for determining paths across MPLS networks.

CCAMP

The Common Control and Measurement Plane (CCAMP) working group is in charge of defining a common control plane (GMPLS) for the non-packet technologies which are deployed by network providers. The technologies, whose data plane is standardised in other SDOs such as ITU-T or IEEE, include OTN, SDH, and WDM. Devices such as Reconfigurable Optical Add-Drop Multiplexers (ROADMs) or electrical TDM switches can be controlled.

CCAMP uses PCE as an element for determining the paths to be signalled in GMPLS networks using information distributed by extensions to the OSPF and IS-IS protocols. The special requirements in these networks comprise additional network parameters specific to the data plane technologies and to the demands of the services operated in transport networks.

IDR

The Interdomain Routing Protocol (IDR) working group concentrates on maintenance and extension of the Border Gateway Protocol (BGP). Included in this work is Link State BGP (BGP-LS) that exports TE information from a network. This export is commonly seen as a way to populate the TED used by a PCE, and so BGP-LS is an important related technology.

SPRING

The Source Packet Routing in Networking (SPRING) working group is investigating a new paradigm in packet forwarding where the full path of a packet through the network is placed in each packet before it is sent into the network. This approach allows traffic engineering to be applied to micro-flows or even on a per-packet basis, and to be quickly modified in reaction to changes within the network, without utilising any signalling mechanisms. A PCE may be a valuable tool to determine the paths for packets and there is cooperative work between SPRING and the PCE working group to investigate what PCEP extensions may be required to support this approach.

SFC

The Service Function Chaining (SFC) working group is developing solutions to allow routing of packets to traverse service functions that may be present at different physical or virtual locations within the network. Two parts of this solution space may involve PCE as the paths between service function nodes may be achieved using traffic engineered tunnels, and the service function nodes themselves must be selected to build a service function chain. This is, in many ways, a two-layer networking path computation problem.

DetNet and 6TiSCH

New work is arriving from the Deterministic Networking (DetNet) and IPv6 over the TSCH mode of IEEE 802.15.4e (6TiSCH) work efforts in the IETF. These work efforts are closely related to the Internet of Things (IoT) and consider elements of network resource availability and demands on specific service attributes for end-to-end traffic flows.

In both cases elements of planned paths are desirable and this requires computation of paths based on network graphs and knowledge of the capability of network elements (links and nodes) as well an understanding of resource availability and characteristics. A PCE is a potential solution.

3.4 Using Open Source PCE

An easy way to experiment with PCE and test new functionalities, without the need of obtaining a commercial product, is to use Open Source PCE tools. The PACE project has released a new open source PCE (netphony), described in detail in D4.2, that can be used for these purposes.

3.4.1 Testing Open Source PCE

The Netphony PCE can get a topology from multiple sources. The most convenient for a simple test is to describe the topology in an xml file, with nodes, links, and TE parameters. That information is enough to test the computing capabilities of the PCE with the help of a test PCC.

The next step is to use the capabilities to dynamically obtain the topology. The PCE can listen to OSPF-TE messages and build a Traffic Engineering Database. However, the preferred option is to take the advantage of BGP-LS and activate the BGP-LS speaker, configure the remote peer and get the TE information.

3.4.2 Contributing to Open Source PCE

The Open source Netphony PCE code is available in github in a set of repositories. In order to contribute to it, the developer needs first a github account and fork the repository. The, in his own repository, a new branch needs to be created with his changes. Then, once the code is completed, a pull request against the development branch of the original repository needs to be done.

4 Deploying PCE

Actual deployments of PCE solutions (with PCE defined in its strict sense, not covering path computation functions in its widest sense) in production networks is very limited. There are limited deployments of PCE in Transport SDN (T-SDN) architectures. It was noted that, from an operator's perspective, the lack of deployment is not necessarily (always) due to technical reasons.

A point to note that constraints the use of PCE in transport networks is the actual deployments of GMPLS as a control plane. According to a vendor, grossly speaking, only 20% of transport networks have deployed GMPLS, 80% are purely managed networks from a centralised Network Management System. Conversely, a service provider notes that whenever they have a mesh transport network they deploy a GMPLS control plane. It was noted that this is just a sample and the difficulty of obtaining actual information on deployments.

4.1 PCE Recipes

PCE solutions may be deployed in a range of scenarios, either as a component of an NMS (it was pointed out that it already is, although the interfaces may not be external) or as a component of T-SDN.

For currently existing GMPLS Networks, it is not realistic to migrate deployments to a GMPLS/Passive PCE architecture because the deployed nodes do not support PCEP. However, since a provisioning interface supports a strict ERO it can benefit indirectly from a PCE, where the path is computed and sent to the ingress node using a provisioning interface.

Although it is accepted to be a topic of debate, the active stateful capabilities were accepted as part of a PCE feature set.

Finally, the following sub-sections highlight the current PCE applicability ("PCE Recipes") documents:

4.1.1 Applicability of the Path Computation Element to Inter-Area and Inter-AS MPLS and GMPLS Traffic Engineering

This document examines the applicability of the PCE architecture, protocols, and protocol extensions for computing multi-area and multi-AS paths in MPLS and GMPLS networks:

https://tools.ietf.org/id/draft-ietf-pce-inter-area-as-applicability

This PCE applicability IETF I-D is being developed by the PACE partners and the PACE project is suitable acknowledged in the document.

4.1.2 Applicability of the Path Computation Element (PCE) to Point-to-Multipoint (P2MP) MPLS and GMPLS Traffic Engineering (TE)

This document examines the applicability of PCE to path computation for P2MP TE LSPs in MPLS and GMPLS networks. It describes the motivation for using a PCE to compute these paths and examines which of the PCE architectural models are appropriate:

https://tools.ietf.org/html/rfc5671

4.1.3 The Applicability of the PCE to Computing Protection and Recovery Paths for Single Domain and Multi-Domain Networks

This document examines the applicability of the PCE architecture, protocols, and procedures for computing protection paths and restoration services, for single and multi-domain networks. This document also explains the mechanism of Fast Re-Route (FRR) where a point of local repair (PLR) needs to find the appropriate merge point (MP) to do bypass path computation using PCE:

https://tools.ietf.org/html/draft-chen-pce-protection-applicability

4.1.4 Applicability of a Stateful Path Computation Element (PCE)

This document describes general considerations for a stateful PCE deployment and examines its applicability and benefits, as well as its challenges and limitations through a number of use cases:

https://tools.ietf.org/html/draft-ietf-pce-stateful-pce-app

5. Missing PCE Ingredients

5.1 Future Research Opportunities

During the life of PACE we have run a number of workshops (in Madrid and Valencia) we identified what is believed to be missing for the adoption and deployment of PCE and PCE based solutions. These are described in the following sub-sections and represent missing “ingredients” for the research opportunities.

5.1.1 North Bound Interfaces

North Bound Interfaces (generically speaking, interfaces that are exported to applications or “Orchestrating” entities) are still to be defined. There are ongoing works at the Open Networking Forum (ONF) and related projects (ODL) for this purpose. The interfaces should allow e.g. specifying requests, or operating on the PCE as a functional entity. Additionally, having interfaces that allow operating on the main PCE databases (create, read, update, delete operations) is considered a strength.

Note: for the purposes of a transport network, the feature set of an AS-PCE and SDN controller are quite similar.

5.1.2 Management Interfaces

Related to the above points, PCE Management interfaces (interfaces that enable e.g. the configuration of the PCE to operate as such) also need to be developed. It was noted that, in general, too many existing and competing management interfaces exist. It is recommended that a management framework be adopted / developed, based, ideally, on open and standard interfaces and potentially reusing existing architectures (Netconf/Yang was mentioned as an example).

5.1.3 Open and standard models for the description of a Transport Networks

For the adoption of PCE, and in view of its integration in SDN architectures for transport networks, it is required to work on standard models for transport network topologies and inventories that build up starting from models for e.g. optical devices and to compose them for networks.

It was noted, though, that it may not be realistic to expect standards to cover all the requirements, and that there are some aspects (e.g. optical impairments) that are still quite vendor specific. It is thus concluded that models should be extensible, avoiding vendor lock-in.

The Traffic Engineering Database (TED) is at the core of the PCE function. The concept of TED is only defined at a macroscopic level, and it is often assumed to be quite bound to existing IGP (e.g. OSPF-TE / IS-IS) routing protocols and their Traffic Engineering attributes. It is widely accepted that this limited view of TED is not enough, and that it does not take into account additional constraints related to diverse aspects such as impairments, collected information not available at the IGP or even OSS/BSS constraints.

Additionally, it is required that the TEDs are defined in view of actual multi-layer deployments, not considering isolated layers. This involves considering e.g. the OTN multi-layer network in an integrated way, covering the electrical / signal and optical layer as well as tributary/client interfaces and adaptation.

5.1.4 Service Provisioning Interface

If defining a network service as a network connectivity service such an Optical LSP, it was clearly identified that for transport networks, there is need for a provisioning interface, preferably open & standard. PCEP is and can be used as a provisioning interface, although there are other candidates (Netconf /Restconf, etc.) resulting in a trade-off covering simplicity, existing deployments, features, etc. That said, PCEP as it is does not cover necessary provisioning operations such as FEC mapping, client adaptation or direct forwarding programming in a similar manner to OpenFlow. It is required further study covering a requirements & gap analysis.

5.1.5 Interfaces

The topic of actual interfaces and protocols to the PCE function was also discussed, considering efficiency and ease of use. PCEP is undeniably bound to the PCE concept, but it may end up being delegated to specific settings. In short, interfaces to the PCE (be management interfaces, north bound interfaces or south-bound / provisioning interfaces) may benefit of having a unified framework and technology. The pros and cons of existing interfaces (e.g., socket based TCP/IP low level interfaces with binary streams and length delimited messages versus high level interfaces based on, for instance, REST / HTTP that can be quick to develop and extend given the convenience of existing frameworks) were discussed. The choice of an interface is clearly constraint by the requirements of the future deployments.

PCEP was mentioned as being initially conceived for the purposes of path computation requests and responses but which was clearly extended to support other functions such as monitoring, topology and service management, etc. At the time being, no further uses of PCEP are considered in a short term, although in general, its use for new functions may be discouraged if there are existing protocols for a given purpose. It is nonetheless quite implementation specific and implies a trade-off between ease of use / features / complexity / time-to-market, etc.

5.1.6 Multi-domain, multi-vendor and multi-technology environments

It is accepted that transport networks are segmented (e.g., in access, aggregation and core segments) and end-to-end services need to be orchestrated along segments. This requires abstraction and orchestration functions.

5.1.7 Wider Selection of Programming Languages

As developers, the choice of programming language is not seen as a critical aspect and developments of PCE based solutions in C, C++, Java and python have been reported. It is widely accepted that it is very dependent on deployment requirements such as the number of requests per unit time and desired performance. Therefore, further investigation and implementation of PCE and related technologies in a wider array of programming languages is required.

5.1.8 Fault Tolerance and Control plane design

The suitability of existing Data Communications Network (DCN) for network management for the purposes of SDN like control needs to be further investigated, taking into account aspects such as latency and bandwidth requirements.

5.1.9 Architectural Considerations

It was stated that new architectures (including refinements of existing ones) should be conceived for a better integration of the active stateful capabilities (also referred to as Push-model).

Appendix

Reference Documents

[1] A. Farrel, J.P. Vasseur, and J. Ash, "A Path Computation Element (PCE)-Based Architecture," IETF RFC 4655, Aug. 2006

[2] J. Ash et al., "Path Computation Element (PCE) Communication protocol Generic Requirements", IETF RFC 4657, Sept. 2006

[3] J. P. Vasseur Ed. and JL. Le Roux Ed., "Path computation element (PCE) communication protocol (PCEP)", IETF RFC 5440, March 2009

[4] I. Nishioka et al., "Use of the Synchronization Vector (SVEC) List for Synchronized Dependent Path Computation", IETF RFC 6007, Sept. 2010

[5] A. Farrel, "Inclusion of Manageability Sections in Path Computation Element (PCE) Working Group Drafts", IETF RFC 6123, Feb. 2011

[6] I. Briskin et al., "Policy-enabled Path Computation Framework", IETF RFC 5394, Dec. 2008

[7] S. Sivabalan et al., "Diffserv-Aware Class-Type Object for the Path Computation Element Communication Protocol", IETF RFC 5455, March 2009

[8] Y. Lee et al., "Path Computation Element Communication Protocol (PCEP) Requirements and Protocol Extensions in Support of Global Concurrent Optimization", IETF RFC 5557, July 2009

[9] JP. Vasseur et al., "A Set of Monitoring Tools for Path Computation Element (PCE)-Based Architecture", IETF RFC 5886, June 2010

[10] JL. Le Roux et al., "Encoding of Objective Functions in the Path Computation Element Communication Protocol (PCEP)", IETF RFC 5541, June 2009

[11] S. Yasukawa et al., "Path Computation Clients (PCC) - Path Computation Element (PCE) Requirements for Point-to-MultiPoint MPLS-TE", IETF RFC 5862, June 2010

[12] Q. Zhao et al., "Extensions to the Path Computation Element Communication Protocol (PCEP) for Point-to-Multipoint Traffic Engineering Label Switched Paths", IETF RFC 6006, September 2010

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

[14] F. Zhang et al., "Conveying Vendor-Specific Constraints in the Path Computation Element Communication Protocol", IETF RFC 7470, March 2015

[15] R. Bradford et al., "Preserving Topology Confidentiality in Inter-Domain Path Computation Using a Path-Key-Based Mechanism", IETF RFC 5520, April 2009

[16] A. Koushik et al., "Path Computation Element Communication Protocol (PCEP) Management Information Base (MIB) Module", IETF RFC 7420, Dec. 2014

[17] JL. Le Roux, "Requirements for Path Computation Element (PCE) Discovery, IETF RFC 4674, Oct. 2006

[18] JL. Le Roux et al., "OSPF Protocol Extensions for Path Computation Element (PCE) Discovery", IETF RFC 5088, Jan. 2008

[19] JL. Le Roux et al., "IS-IS Protocol Extensions for Path Computation Element (PCE)", IETF RFC

5089, Jan. 2008

[20] N. Bitar et al., "Inter-AS Requirements for the Path Computation Element Communication Protocol (PCECP)", IETF RFC 5376, Nov. 2008.

[21] E. Oki et al., "Framework for PCE-Based Inter-Layer MPLS and GMPLS Traffic Engineering", IETF RFC 5623, Sept. 2009

[22] T. Takeda et al., "PCC-PCE Communication and PCE Discovery Requirements for Inter-Layer Traffic Engineering", IETF RFC 6457, Dec. 2011

[23] JL. LE Roux, "Path Computation Element Communication Protocol (PCECP) Specific Requirements for Inter-Area MPLS and GMPLS Traffic Engineering", IETF RFC 4927, June 2007

[24] D. King et al., "The Application of the Path Computation Element Architecture to the Determination of a Sequence of Domains in MPLS and GMPLS", IETF RFC 6805, Nov. 2012

[25] T. Otani et al., "Requirements for GMPLS Applications of PCE", IETF RFC 7025, Sept. 2013

[26] Y. Lee et al., "Path Computation Element Communication Protocol (PCEP) Requirements for Wavelength Switched Optical Network (WSON) Routing and Wavelength Assignment", IETF RFC 7449, Feb. 2015

[27] F. Zhang et al., draft-ietf-pce-hierarchy-extensions-02, IETF PCE WG, work in progress, Jan. 2015

[28] C. Margaria et al., draft-ietf-pce-gmpls-pcep-extensions-10, IETF PCE WG, work in progress, Oct. 2014

[29] E. Crabbe et al., "PCEP extensions for stateful PCE," IETF draft-ietf-pce-stateful-pce, work in progress, May 2013

[30] E. Crabbe et al., "PCEP extensions for PCE-initiated LSP setup in a stateful PCE model," IETF draft-crabbe-pce-pce-initiated-lsp, work in progress, April 2013

[31] JP. Vasseur et al., "A Backward-Recursive PCE-Based Computation (BRPC) Procedure to Compute Shortest Constrained Inter-Domain Traffic Engineering Label Switched Paths", IETF RFC 5441, April 2009

[32] JP. Vasseur et al., "A Per-Domain Path Computation Method for Establishing Inter-Domain Traffic Engineering (TE) Label Switched Paths (LSPs)", IETF RFC 5152, February 2008

[33] D. King and A. Farrel, "A PCE-based Architecture for Application-Based Network Operation", IETF RFC 7491, March 2015

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

[35] "SDN architecture", ONF TR-502, issue 1, June 2014, Open Networking Foundation

[36] F. Zhang and A. Farrel, "Conveying Vendor-Specific Constraints in the Path Computation Element Communication Protocol", IETF RFC 7150, Mar. 2014

[37] Q. Zhao et al., "PCE-Based Computation Procedure to Compute Shortest Constrained Point-to-Multipoint (P2MP) Inter-Domain Traffic Engineering Label Switched Paths", IETF RFC 7334, Aug. 2014

[38] A. Farrel and D. King, "Unanswered Questions in the Path Computation Element Architecture", IETF RFC 7399, Oct. 2014

[39] S. Yasukawa and A. Farrel, "Applicability of the Path Computation Element (PCE) to Point-to-Multipoint (P2MP) MPLS and GMPLS Traffic Engineering (TE)", IETF RFC 5671, Oct. 2009.

[40] "ONOS project", http://onosproject.org/

[41] "Introducing ONOS – a SDN network operating system for Service Providers", whitepaper, November 2014

[42] "ONOS Packet + Optical Use Case", ONOS Wiki, available at https://wiki.onosproject.org/display/ONOS/Packet+Optical

[43] Opendaylight Project, http://www.opendaylight.org/

Acronyms

ERO	Explicit Route Object
GCO	Global Concurrent Optimization
GMPLS	Generalized Multi-Protocol Label Switching
IGP	Interior Gateway Protocol
LSPDB	LSP Database
NMS	Network Management System
PCC	Path Computation Client
PCE	Path Computation Element
IETF	Internet Engineering Task Force
OIF	Optical Internetworking Forum
ITU	International Telecommunication Union
ROADM	Reconfigurable Optical Add Drop Multiplexer
RSA	Routing and Spectrum Assignment
RSVP-TE	ReSerVation Protocol – Traffic Engineering
RWA	Routing and Wavelength Assignment
SA	Spectrum Assignment
SSON	Spectrum Switched Optical Network
TED	Traffic Engineering Database
TE-LSA	Traffic Engineering – Link State Advertisement