TEAS Working Group
Internet-Draft
Intended status: Informational
Expires: January 5, 2020

Q. Zhao
Z. Li
B. Khasanov
D. Dhody
Huawei Technologies
K. Ke
Tencent Holdings Ltd.
L. Fang
Expedia, Inc.
C. Zhou
Cisco Systems
B. Zhang
Telus Communications
A. Rachitskiy
Mobile TeleSystems JLLC
A. Gulida
LLC "Lifetech"
July 4, 2019

The Use Cases for Path Computation Element (PCE) as a Central Controller (PCECC).

draft-ietf-teas-pcecc-use-cases-04

Abstract

The Path Computation Element (PCE) is a core component of a Software-Defined Networking (SDN) system. It can compute optimal paths for traffic across a network and can also update the paths to reflect changes in the network or traffic demands. PCE was developed to derive paths for MPLS Label Switched Paths (LSPs), which are supplied to the head end of the LSP using the Path Computation Element Communication Protocol (PCEP).

SDN has a broader applicability than signaled MPLS traffic-engineered (TE) networks, and the PCE may be used to determine paths in a range of use cases including static LSPs, segment routing (SR), Service Function Chaining (SFC), and most forms of a routed or switched network. It is, therefore, reasonable to consider PCEP as a control protocol for use in these environments to allow the PCE to be fully enabled as a central controller.

This document describes general considerations for PCECC deployment and examines its applicability and benefits, as well as its challenges and limitations, through a number of use cases. PCEP extensions required for stateful PCE usage are covered in separate documents.

This is a living document to catalogue the use cases for PCECC. There is currently no intention to publish this work as an RFC.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

An Architecture for Use of PCE and PCEP [RFC5440] in a Network with Central Control [RFC8283] describes SDN architecture where the Path Computation Element (PCE) determines paths for variety of different usecases, with PCEP as a general southbound communication protocol with all the nodes along the path..

[I-D.ietf-pce-pcep-extension-for-pce-controller] introduces the procedures and extensions for PCEP to support the PCECC architecture [RFC8283].

This draft describes the various usecases for the PCECC architecture.

This is a living document to catalogue the use cases for PCECC. There is currently no intention to publish this work as an RFC.

2. Terminology

The following terminology is used in this document.

IGP: Interior Gateway Protocol. Either of the two routing protocols, Open Shortest Path First (OSPF) or Intermediate System to Intermediate System (IS-IS).

PCC: Path Computation Client: any client application requesting a path computation to be performed by a Path Computation Element.

PCE: Path Computation Element. An entity (component, application, or network node) that is capable of computing a network path or route based on a network graph and applying computational constraints.

PCECC: PCE as a central controller. Extension of PCE to support SDN functions as per [RFC8283].

TE: Traffic Engineering.

3. Application Scenarios

In the following sections, several use cases are described, showcasing scenarios that benefit from the deployment of PCECC.

3.1. Use Cases of PCECC for Label Management

As per [RFC8283], in some cases, the PCE-based controller can take responsibility for managing some part of the MPLS label space for each of the routers that it controls, and it may taker wider responsibility for partitioning the label space for each router and allocating different parts for different uses, communicating the ranges to the router using PCEP.

[I-D.ietf-pce-pcep-extension-for-pce-controller] describe a mode where LSPs are provisioned as explicit label instructions at each hop on the end-to-end path. Each router along the path must be told what label forwarding instructions to program and what resources to reserve. The controller uses PCEP to communicate with each router along the path of the end-to-end LSP. For this to work, the PCE-based controller will take responsibility for managing some part of the MPLS label space for each of the routers that it controls. An

extension to PCEP could be done to allow a PCC to inform the PCE of such a label space to control.

[I-D.ietf-pce-segment-routing] specifies extensions to PCEP that allow a stateful PCE to compute, update or initiate SR-TE paths. [I-D.zhao-pce-pcep-extension-pce-controller-sr] describes the mechanism for PCECC to allocate and provision the node/prefix/adjacency label (SID) via PCEP. To make such allocation PCE needs to be aware of the label space from Segment Routing Global Block (SRGB) or Segment Routing Local Block (SRLB) [RFC8402] of the node that it controls. A mechanism for a PCC to inform the PCE of such a label space to control is needed within PCEP. The full SRGB/SRLB of a node could be learned via existing IGP or BGP-LS mechanism too.

[I-D.li-pce-controlled-id-space] defines a PCEP extension to support advertisement of the MPLS label space to the PCE to control.

There have been various proposals for Global Labels, the PCECC architecture could be used as means to learn the label space of nodes, and could also be used to determine and provision the global label range.

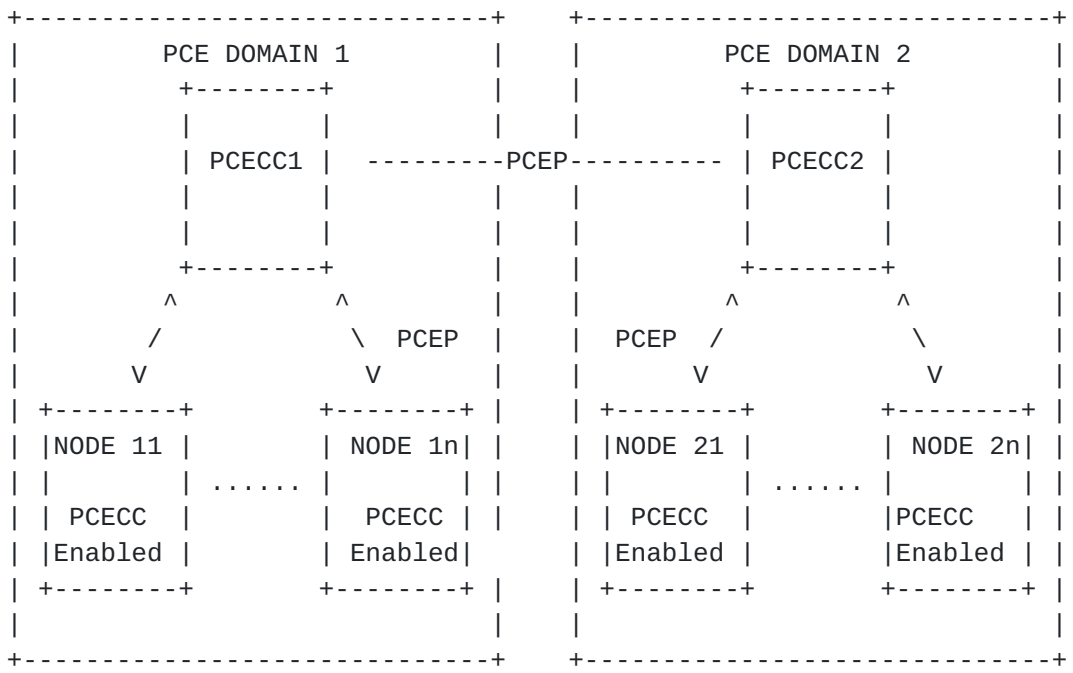


Figure 1: PCECC for Label Management

o PCC would advertise the PCECC capability to the PCE (central controller-PCECC) [I-D.ietf-pce-pcep-extension-for-pce-controller].

- The PCECC could also learn the label range set aside by the PCC ([I-D.li-pce-controlled-id-space]).

- Optionally, the PCECC could determine the shared MPLS global label range for the network.

  - In the case that the shared global label range need to be negotiated across multiple domains, the central controllers of these domains would also need to negotiate a common global label range across domains.

  - The PCECC would need to set the shared global label range to all PCC nodes in the network.

3.2. Using PCECC for SR

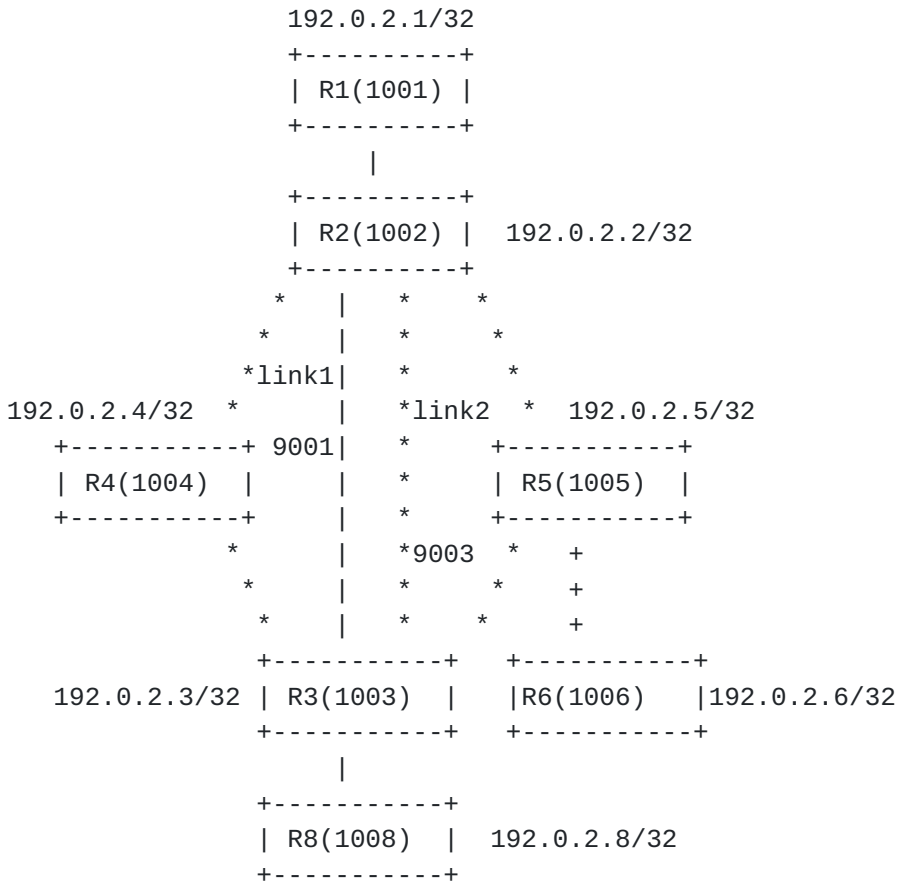
Segment Routing (SR) leverages the source routing paradigm. Using SR, a source node steers a packet through a path without relying on hop-by-hop signaling protocols such as LDP or RSVP-TE. Each path is specified as an ordered list of instructions called "segments". Each segment is an instruction to route the packet to a specific place in the network, or to perform a specific service on the packet. A database of segments can be distributed through the network using a routing protocol (such as IS-IS or OSPF) or by any other means. PCEP (and PCECC) could be one such means.

[I-D.ietf-pce-segment-routing] specify the SR specific PCEP extensions. PCECC may further use PCEP protocol for SR SID (Segment Identifier) distribution to the SR nodes (PCC) with some benefits. If the PCECC allocates and maintains the SID in the network for the nodes and adjacencies; and further distributes them to the SR nodes directly via the PCEP session has some advantage over the configurations on each SR node and flooding via IGP, especially in a SDN environment.

When the PCECC is used for the distribution of the node segment ID and adjacency segment ID, the node segment ID is allocated from the SRGB of the node. For the allocation of adjacency segment ID, the allocation is from the SRLB of the node as described in [I-D.zhao-pce-pcep-extension-pce-controller-sr].

[RFC8355] identifies various protection and resiliency usecases for SR. Path protection lets the ingress node be in charge of the failure recovery (used for SR-TE). Also protection can be performed by the node adjacent to the failed component, commonly referred to as local protection techniques or fast-reroute (FRR) techniques. In case of PCECC, the protection paths can be pre-computed and setup by the PCE.

The following example illustrate the use case where the node SID and adjacency SID are allocated by the PCECC.



3.2.1. PCECC SID Allocation

Each node (PCC) is allocated a node-SID by the PCECC. The PCECC needs to update the label map of each node to all the nodes in the domain. On receiving the label map, each node (PCC) uses the local routing information to determine the next-hop and download the label forwarding instructions accordingly. The forwarding behavior and the end result is same as IGP based Node-SID in SR. Thus, from anywhere in the domain, it enforces the ECMP-aware shortest-path forwarding of the packet towards the related node.

For each adjacency in the network, PCECC can allocate an Adj-SID. The PCECC sends PCInitiate message to update the label map of each Adj to the corresponding nodes in the domain. Each node (PCC) download the label forwarding instructions accordingly. The forwarding behavior and the end result is similar to IGP based "Adj-SID" in SR.

The various mechanism are described in [I-D.zhao-pce-pcep-extension-pce-controller-sr].

3.2.2. Use Cases of PCECC for SR Best Effort (BE) Path

In this mode of the solution, the PCECC just need to allocate the node segment ID and adjacency ID (without calculating the explicit path for the SR path). The ingress of the forwarding path just need to encapsulate the destination node segment ID on top of the packet. All the intermediate nodes will forward the packet based on the destination node SID. It is similar to the LDP LSP.

R1 may send a packet to R8 simply by pushing an SR header with segment list {1008} (Node SID for R8). The path would be the based on the routing/nexthop calculation on the routers.

3.2.3. Use Cases of PCECC for SR Traffic Engineering (TE) Path

SR-TE paths may not follow an IGP SPT. Such paths may be chosen by a PCECC and provisioned on the ingress node of the SR-TE path. The SR header consists of a list of SIDs (or MPLS labels). The header has all necessary information so that, the packets can be guided from the ingress node to the egress node of the path; hence, there is no need for any signaling protocol. For the case where strict traffic engineering path is needed, all the adjacency SID are stacked, otherwise a combination of node-SID or adj-SID can be used for the SR-TE paths.

Note that the bandwidth reservations is only guaranteed at controller and through the enforce of the bandwidth admission control. As for the RSVP-TE LSP case, the control plane signaling also does the link bandwidth reservation in each hop of the path.

The SR traffic engineering path examples are explained as bellow:

Note that the node SID for each node is allocated from the SRGB and adjacency SID for each link are allocated from the SRLB for each node.

Example 1:

R1 may send a packet P1 to R8 simply by pushing an SR header with segment list {1008}. Based on the best path, it could be: R1-R2-R3-R8.

Example 2:

R1 may send a packet P2 to R8 by pushing an SR header with segment list {1002, 9001, 1008}. The path should be: R1-R2-link1-R3-R8.

Example 3:

R1 may send a packet P3 to R8 via R4 by pushing an SR header with segment list {1004, 1008}. The path could be : R1-R2-R4-R3-R8

The local protection examples for SR TE path are explained below:

Example 4: local link protection:

- R1 may send a packet P4 to R8 by pushing an SR header with segment list {1002, 9001, 1008}. The path should be: R1-R2-link1-R3-R8.

- When node R2 receives the packet from R1 which has the header of link1-R3-R8, and also find out there is a link failure of link1, then the R2 can enforce the traffic over the bypass to send out the packet with header of R3-R8 through link2.

Example 5: local node protection:

- R1 may send a packet P5 to R8 by pushing an SR header with segment list {1004, 1008}. The path could be : R1-R2-R4-R3-R8.

- When node R2 receives the packet from R1 which has the header of {1004, 1008}, and also finds out there is a node failure for node4, then it can enforce the traffic over the bypass and send out the packet with header of {1005, 1008} to node5 instead of node4.

3.3. Use Cases of PCECC for TE LSP

In the Section 3.2 the case of SR path via PCECC is discussed. Although those cases give the simplicity and scalability, but there are existing functionalities for the traffic engineering path such as the bandwidth guarantee, monitoring where SR based solution are complex. Also there are cases where the depth of the label stack is an issue for existing deployment and certain vendors.

So to address these issues, PCECC architecture also support the TE LSP functionalities. To achieve this, the existing PCEP can be used to communicate between the PCECC and nodes along the path. This is similar to static LSPs, where LSPs can be provisioned as explicit label instructions at each hop on the end-to-end path. Each router along the path must be told what label- forwarding instructions to program and what resources to reserve. The PCE-based controller keeps a view of the network and determines the paths of the end-to-

end LSPs, and the controller uses PCEP to communicate with each router along the path of the end-to-end LSP.

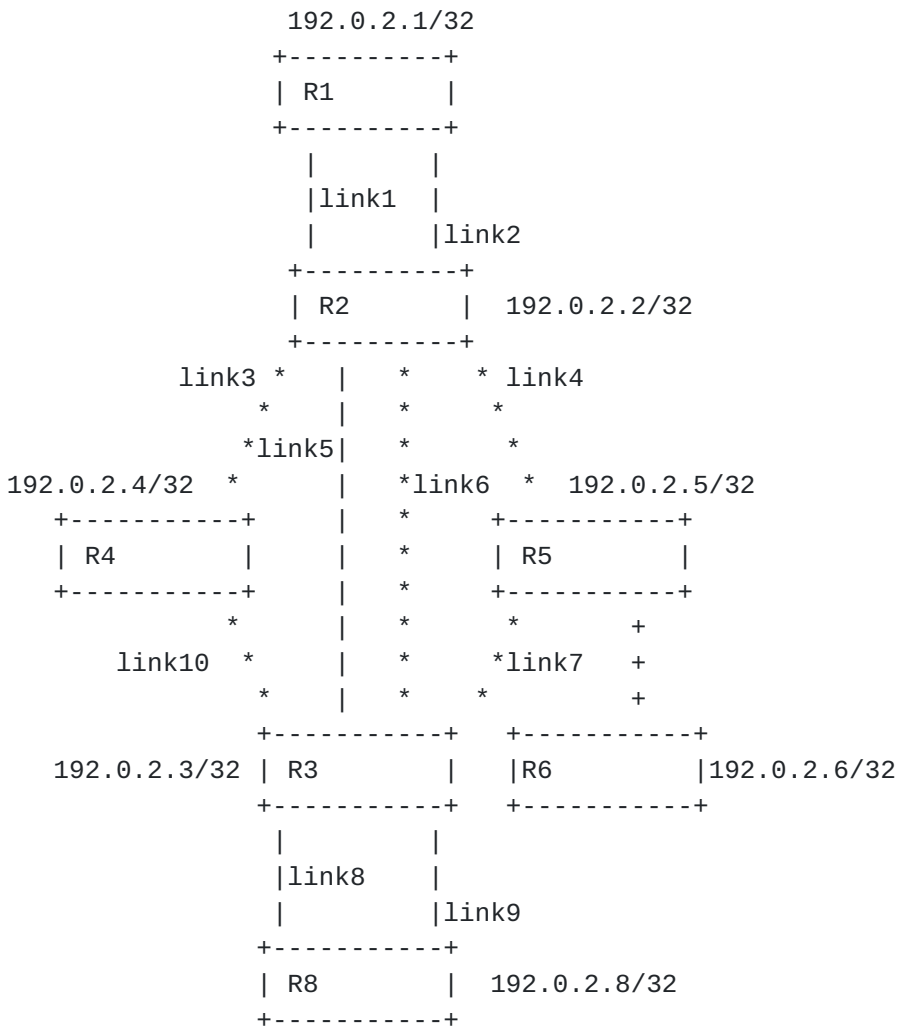


Figure 2: PCECC TE LSP Setup Example

- Based on path computation request / delegation or PCE initiation, the PCECC receives the PCECC request with constraints and optimization criteria.

- PCECC would calculate the optimal path according to given constrains (e.g. bandwidth).

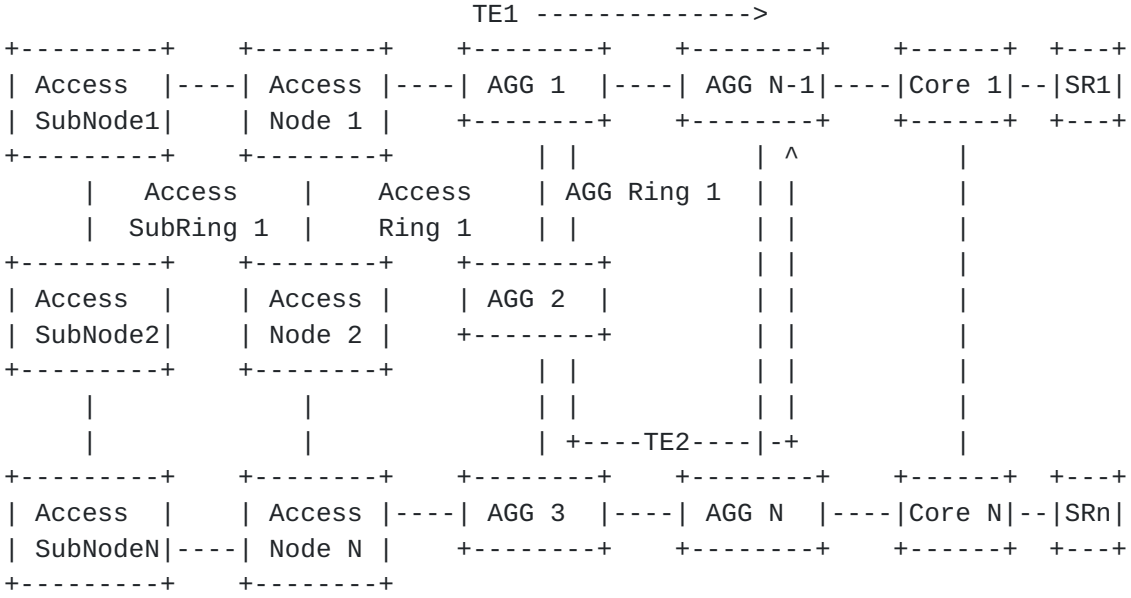
- PCECC would provision each node along the path and assign incoming and outgoing labels from R1 to R8 with the path: {R1, link1, 1001}, {1001, R2, link3, 2003], {2003, R4, link10, 4010}, {4010, R3, link8, 3008}, {3008, R8}.

- For the end to end protection, PCECC program each node along the path from R1 to R8 with the secondary path: {R1, link2, 1002}, {1002, R2, link4, 2004], {2004, R5, link7, 5007}, {5007, R3, link9, 3009}, {3009, R8}.

- It is also possible to have a bypass path for the local protection setup by the PCECC. For example, the primary path as above, then to protect the node R4 locally, PCECC can program the bypass path like this: {R2, link5, 2005}, {2005, R3}. By doing this, the node R4 is locally protected at R2.

3.3.1. PCECC Load Balancing (LB) Use Case

Very often many service providers use TE tunnels for solving issues with non-deterministic paths in their networks. One example of such applications is usage of TEs in the mobile backhaul (MBH). Consider the following topology -



This MBH architecture uses L2 access rings and sub-rings. L3 starts at the aggregation layer. For the sake of simplicity, the figure shows only one access sub-ring, access ring and aggregation ring (AGG1...AGGN), connected by Nx10GE interfaces. Aggregation domain runs its own IGP. There are two Egress routers (AGG N-1,AGG N) that are connected to the Core domain via L2 interfaces. Core also have connections to service routers, RSVP-TEs are used for MPLS transport inside the ring. There could be at least 2 tunnels (one way) from each AGG router to egress AGG routers. There are also many L2 access rings connected to AGG routers.

Service deployment made by means of either L2VPNs (VPLS) or L3VPNs. Those services use MPLS TE as transport towards egress AGG routers. TE tunnels could be also used as transport towards service routers in case of seamless MPLS based architecture in the future.

There is a need to solve the following tasks:

- Perform automatic load-balance amongst TE tunnels according to current traffic load.

- TE bandwidth (BW) management: Provide guaranteed BW for specific service: HSI, IPTV, etc., provide time-based BW reservation (BoD) for other services.

- Simplify development of TE tunnels by automation without any manual intervention.

- Provide flexibility for Service Router placement (anywhere in the network by creation of transport LSPs to them).

Since other tasks are already considered by other PCECC use cases, in this section, the focus is on load balancing (LB) task. LB task could be solved by means of PCECC in the following way:

- After application or network service or operator can ask SDN controller (PCECC) for LSP based LB between AGG X and AGG N/AGG N-1 (egress AGG routers which have connections to core) via North Bound Interface (NBI). Each of these would have associated constrains (i.e. Path Setup Type (PST), bandwidth, inclusion or exclusion specific links or nodes, number of paths, objective function (OF), need for disjoint LSP paths etc.).

- PCECC could calculate multiple (Say N) LSPs according to given constrains, calculation is based on results of Objective Function (OF) [RFC5541], constraints, endpoints, same or different bandwidth (BW) , different links (in case of disjoint paths) and other constrains.

- Depending on given LSP Path setup type (PST), PCECC would use download instructions to the PCC. At this stage it is assumed the PCECC is aware of the label space it controls and in case of SR the SID allocation and distribution is already done.

- PCECC would send PCInitiate PCEP message [RFC8281] towards ingress AGG X router(PCC) for each of N LSPs and receives PCRpt PCEP message [RFC8231] back from PCCs. If the PST is PCECC-SR, the PCECC would include the SID stack as per [I-D.ietf-pce-segment-routing]. If the PST is PCECC (basic), then

the PCECC would assigns labels along the calculated path; and set up the path by sending central controller instructions in PCEP message to each node along the path of the LSP as per [I-D.ietf-pce-pcep-extension-for-pce-controller] and then send PCUpd message to the ingress AGG X router with information about new LSP and AGG X(PCC) would respond with PCRpt with LSP status.

- o AGG X as ingress router now have N LSPs towards AGG N and AGG N-1 which are available for installing to router's forwarding and LB of traffic between them. Traffic distribution between those LSPs depends on particular realization of hash-function on that router.

- o Since PCECC is aware of TEDB (TE state) and LSP-DB, it can manage and prevent possible over-subscriptions and limit number of available LB states. Via PCECC mechanism the control can take quick actions into the network by directly provisioning the central control instructions.

3.3.2. PCECC and Inter-AS TE

There are various signaling options for establishing Inter-AS TE LSP: contiguous TE LSP [RFC5151], stitched TE LSP [RFC5150], nested TE LSP [RFC4206].

Requirements for PCE-based Inter-AS setup [RFC5376] describe the approach and PCEP functionality that are needed for establishing Inter-AS TE LSPs.

[RFC5376] also gives Inter- and Intra-AS PCE Reference Model that is provided below in shorten form for the sake of simplicity.

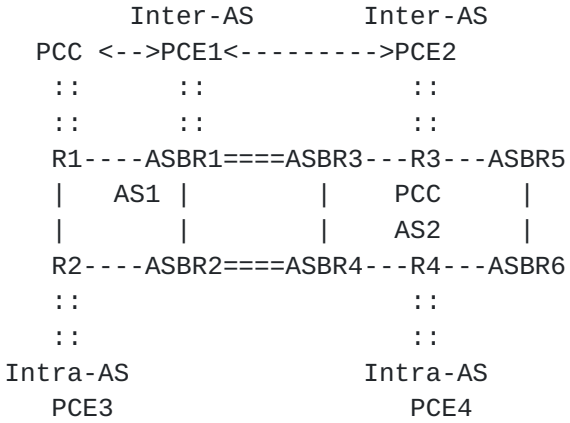


Figure 3: Shorten form of Inter- and Intra-AS PCE Reference Model [RFC5376]

The PCECC belonging to different domain can co-operate to setup inter-AS TE LSP. The stateful H-PCE [I-D.ietf-pce-stateful-hpce] mechanism could also be used to first establish a per-domain PCECC LSP. These could be stitched together to form inter-AS TE LSP as described in [I-D.dugeon-pce-stateful-interdomain].

For the sake of simplicity, here after the focus is on a simplified Inter-AS case when both AS1 and AS2 belong to the same service provider administration. In that case Inter and Intra-AS PCEs could be combined in one single PCE if such combined PCE performance is enough for handling all path computation request and setup. There is a potential to use a single PCE for both ASes if the scalability and performance are enough. The PCE would require interfaces (PCEP and BGP-LS) to both domains. PCECC redundancy mechanisms are described in [RFC8283]. Thus routers in AS1 and AS2 (PCCs) can send PCEP messages towards same PCECC.

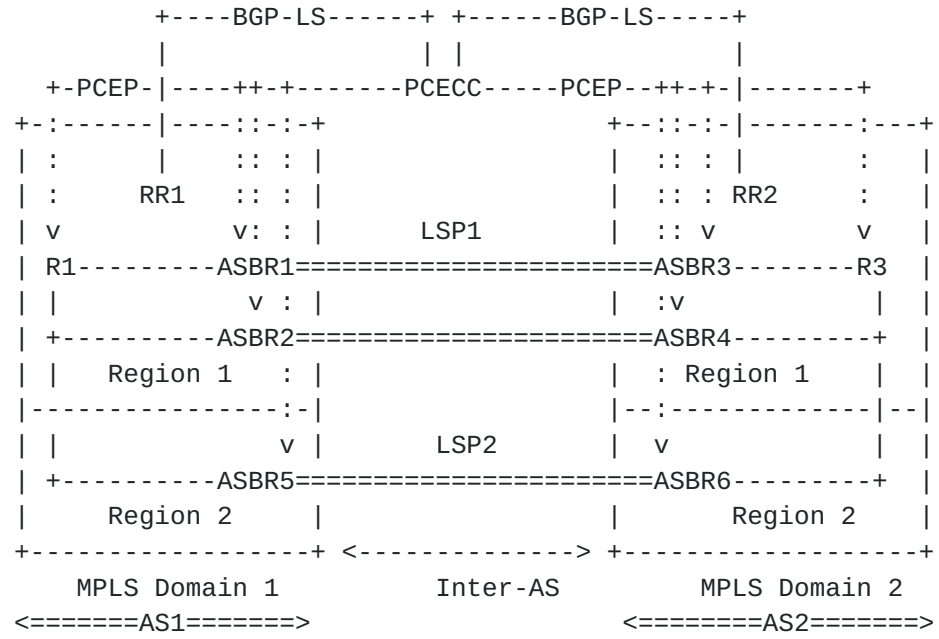


Figure 4: Particular case of Inter-AS PCE

In a case of PCECC Inter-AS TE scenario where service provider controls both domains (AS1 and AS2), each of them have own IGP and MPLS transport. There is a need is to setup Inter-AS LSPs for transporting different services on top of them (Voice, L3VPN etc.). Inter-AS links with different capacity exist in several regions. The task is not only to provision those Inter-AS LSPs with given constrains but also calculate the path and pre-setup the backup Inter-AS LSPs that will be used if primary LSP fails.

As per the Figure 4, LSP1 from R1 to R3 goes via ASBR1 and ASBR3, and it is the primary Inter-AS LSP. R1-R3 LSP2 that go via ASBR5 and ASBR6 is the backup one. In addition there could also be a bypass LSP setup to protect against ASBR or inter-AS link failure.

After the addition of PCECC functionality to PCE (SDN controller), PCECC based Inter-AS TE model SHOULD follow as PCECC usecase for TE LSP as requirements of [RFC5376] with the following details:

- Since PCECC needs to know the topology of both domains AS1 and AS2, PCECC could use BGP-LS peering with routers (or RRs) in both domains.

- PCECC needs to PCEP connectivity towards all routers in both domains (see also section 4 in [RFC5376]) in a similar manner as a SDN controller.

- After operator's application or service orchestrator will create request for tunnel creation of specific service, PCECC should receive that request via NBI (NBI type is implementation dependent, could be NETCONF/Yang, REST etc.). Then PCECC would calculate the optimal path based on Objective Function (OF) and given constraints (i.e. path setup type, bandwidth etc.), including those from [RFC5376]: priority, AS sequence, preferred ASBR, disjoint paths, protection. On this step we would have two paths: R1-ASBR1-ASBR3-R3, R1-ASBR5-ASBR6-R3

- Depending on given LSP PST (PCECC or PCECC-SR), PCECC would use central control download instructions to the PCC. At this stage it is assumed the PCECC is aware of the label space it controls and in case of SR the SID allocation and distribution is already done.

- PCECC would send PCInitiate PCEP message [RFC8281] towards ingress router R1 (PCC) in AS1 and receives PCRpt PCEP message [RFC8231] back from PCC. If the PST is PCECC-SR, the PCECC would include the SID stack as per [I-D.ietf-pce-segment-routing]. It may also include binding SID based on AS boundary. The backup SID stack could also be installed at ingress but more importantly each node along the SR path could also do local protection just based on the top segment. If the PST is PCECC (basic), then the PCECC would assigns labels along the calculated paths (R1-ASBR1-ASBR3-R3, R1-ASBR5-ASBR6-R3); and set up the path by sending central controller instructions in PCEP message to each node along the path of the LSPs as per [I-D.ietf-pce-pcep-extension-for-pce-controller] and then send PCUpd message to the ingress R1 router with information about new LSPs and R1 would respond with PCRpt with LSP(s) status.

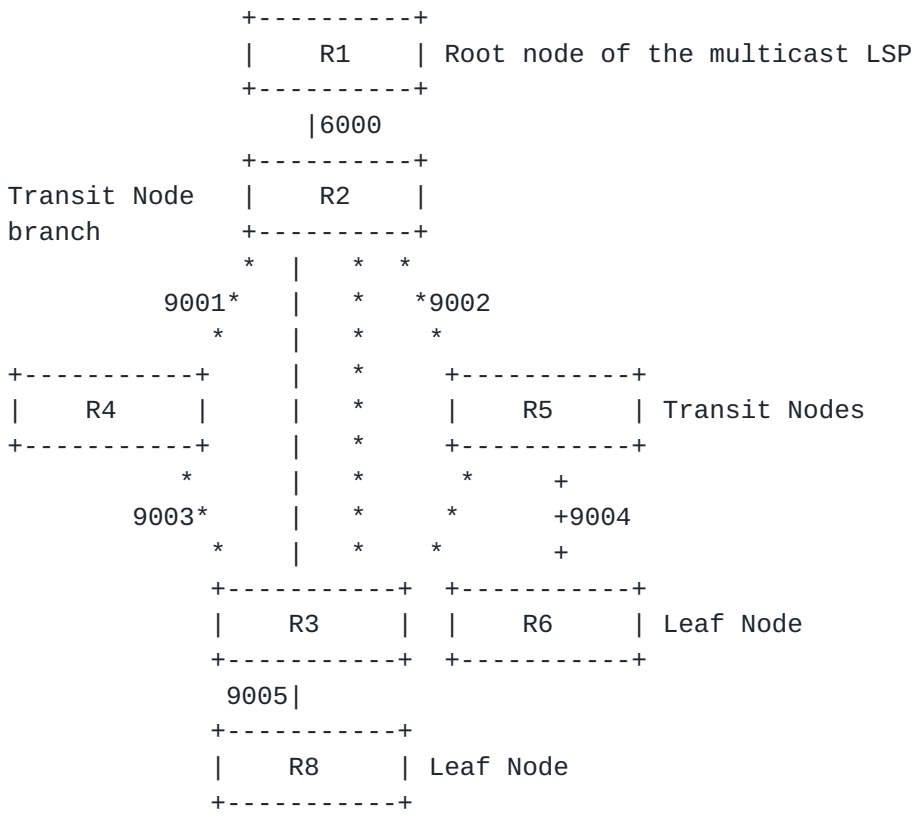
o After that step R1 now have primary and backup TEs (LSP1 and LSP2) towards R3. It is up to router implementation how to make switchover to backup LSP2 if LSP1 fails.

3.4. Use Cases of PCECC for Multicast LSPs

The current multicast LSPs are setup either using the RSVP-TE P2MP or mLDP protocols. The setup of these LSPs may require manual configurations and complex signaling when the protection is considered. By using the PCECC solution, the multicast LSP can be computed and setup through centralized controller which has the full picture of the topology and bandwidth usage for each link. It not only reduces the complex configurations comparing the distributed RSVP-TE P2MP or mLDP signaling, but also it can compute the disjoint primary path and secondary P2MP path efficiently.

3.4.1. Using PCECC for P2MP/MP2MP LSPs' Setup

It is assumed the PCECC is aware of the label space it controls for all nodes and make allocations accordingly.



The P2MP examples are explained here, where R1 is root and R8 and R6 are the leaves.

- Based on the P2MP path computation request / delegation or PCE initiation, the PCECC receives the PCECC request with constraints and optimization criteria.

- PCECC would calculate the optimal P2MP path according to given constrains (i.e.bandwidth).

- PCECC would provision each node along the path and assign incoming and outgoing labels from R1 to {R6, R8} with the path: {R1, 6000}, {6000, R2, {9001,9002}}, {9001, R4, 9003}, {9002, R5, 9004} {9003, R3, 9005}, {9004, R6}, {9005, R8}. The main difference is in the branch node instruction at R2 where two copies of packet are sent towards R4 and R5 with 9001 and 9002 labels respectively.

The packet forwarding involves -

Step1: R1 may send a packet P1 to R2 simply by pushing an label of 6000 to the packet.

Step2: After R2 receives the packet with label 6000, it will forwarding to R4 by swapping label to 9001 and by swapping label of 9002 towards R5.

Step3: After R4 receives the packet with label 9001, it will forwarding to R3 by swapping to 9003. After R5 receives the packet with label 9002, it will forwarding to R6 by swapping to 9004.

Step4: After R3 receives the packet with label 9003, it will forwarding to R8 by swapping to 9005 and when R5 receives the packet with label 9004, it will swap to 9004 and send to R6.

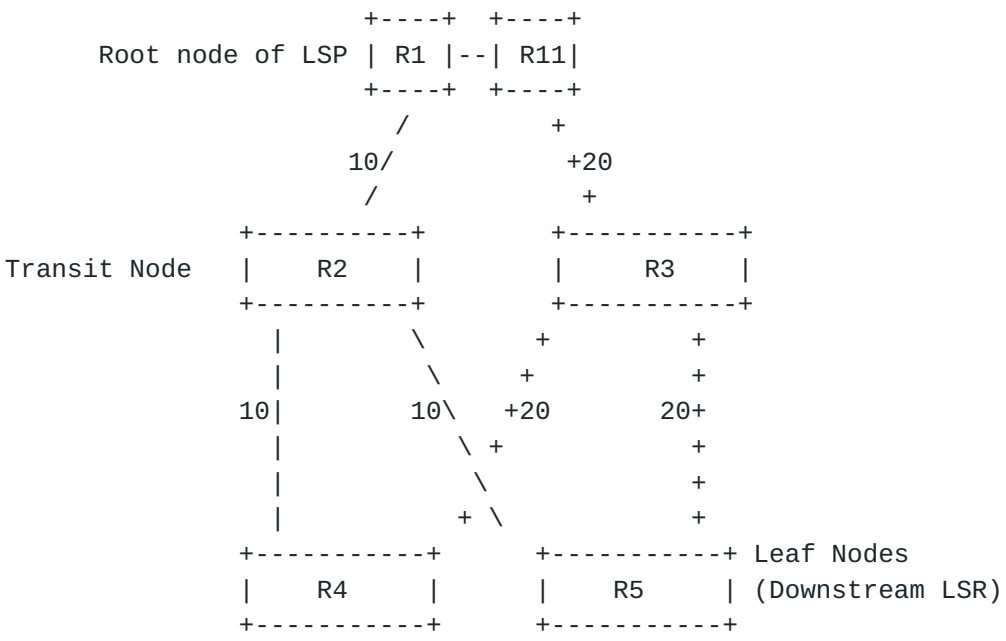
Step5: Packet received at R8 and 9005 is popped; packet receives at R6 and 9004 is popped.

3.4.2. Use Cases of PCECC for the Resiliency of P2MP/MP2MP LSPs

3.4.2.1. PCECC for the End-to-End Protection of the P2MP/MP2MP LSPs

In this section we describe the end-to-end managed path protection service as well as the local protection with the operation management in the PCECC network for the P2MP/MP2MP LSP.

An end-to-end protection principle can be applied for computing backup P2MP or MP2MP LSPs. During computation of the primary multicast trees, PCECC server may also take the computation of a secondary tree into consideration. A PCE may compute the primary and backup P2MP (or MP2MP) LSP together or sequentially.

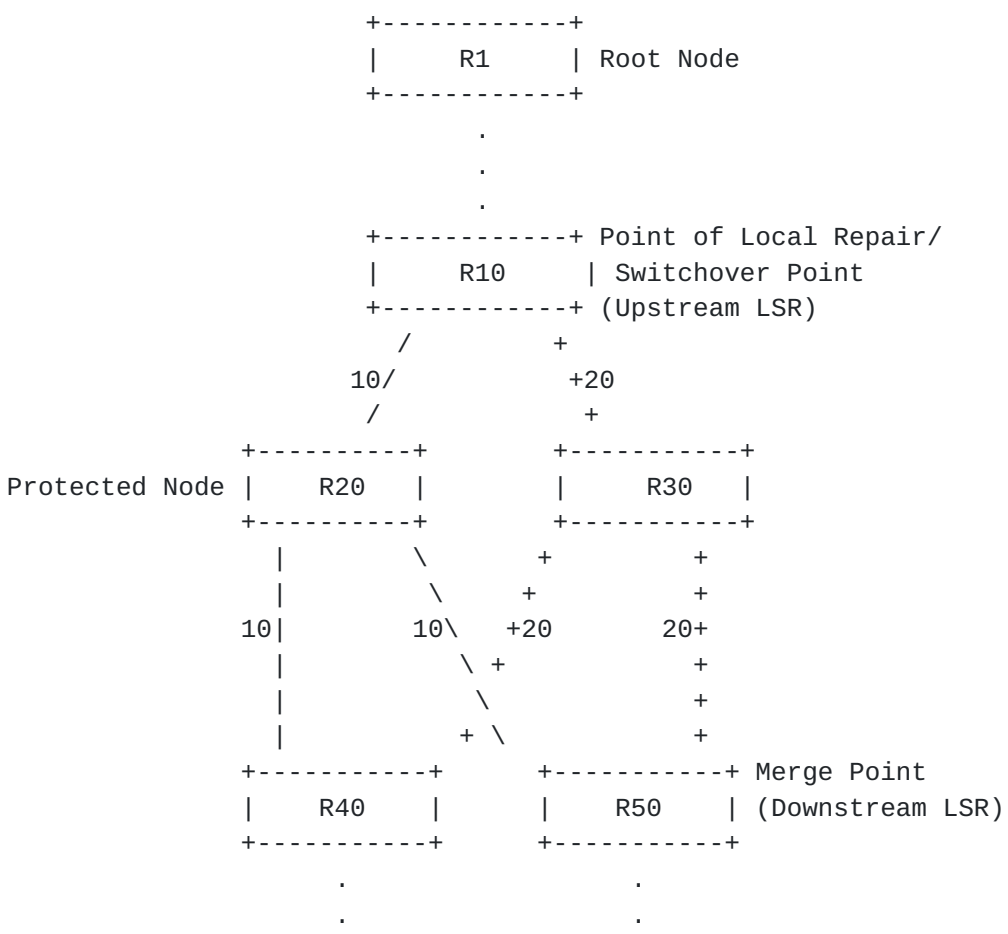


In the example above, when the PCECC setup the primary multicast tree from the root node R1 to the leaves, which is R1->R2->{R4, R5}, at same time, it can setup the backup tree, which is R1->R11->R3->{R4, R5}. Both the these two primary forwarding tree and secondary forwarding tree will be downloaded to each routers along the primary path and the secondary path. The traffic will be forwarded through the R1->R2->{R4, R5} path normally, and when there is a node in the primary tree fails (say R2), then the root node R1 will switch the flow to the backup tree, which is R1->R11->R3->{R4, R5}. By using the PCECC, the path computation and forwarding path downloading can all be done without the complex signaling used in the P2MP RSVP-TE or mLDP.

3.4.2.2. PCECC for the Local Protection of the P2MP/MP2MP LSPs

In this section we describe the local protection service in the PCECC network for the P2MP/MP2MP LSP.

While the PCECC sets up the primary multicast tree, it can also build the back LSP among PLR, the protected node, and MPs (the downstream nodes of the protected node). In the cases where the amount of downstream nodes are huge, this mechanism can avoid unnecessary packet duplication on PLR and protect the network from traffic congestion risk.



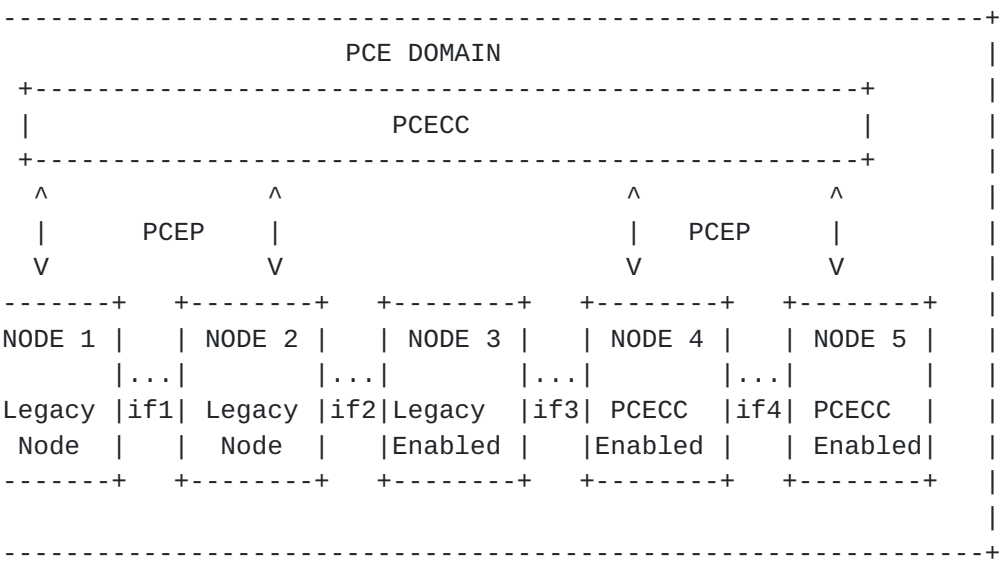
In the example above, when the PCECC setup the primary multicast path around the PLR node R10 to protect node R20, which is R10->R20->{R40, R50}, at same time, it can setup the backup path R10->R30->{R40, R50}. Both the these two primary forwarding path and secondary bypass forwarding path will be downloaded to each routers along the primary path and the secondary bypass path. The traffic will be forwarded through the R10->R20->{R40, R50} path normally, and when there is a node failure for node R20, then the PLR node R10 will switch the flow to the backup path, which is R10->R30->{R40, R50}. By using the PCECC, the path computation and forwarding path downloading can all be done without the complex signaling used in the P2MP RSVP-TE or mLDP.

3.5. Use Cases of PCECC for LSP in the Network Migration

One of the main advantages for PCECC solution is that it has backward compatibility naturally since the PCE server itself can function as a proxy node of MPLS network for all the new nodes which may no longer support the signaling protocols.

As it is illustrated in the following example, the current network could migrate to a total PCECC controlled network gradually by replacing the legacy nodes. During the migration, the legacy nodes still need to signal using the existing MPLS protocol such as LDP and RSVP-TE, and the new nodes setup their portion of the forwarding path through PCECC directly. With the PCECC function as the proxy of these new nodes, MPLS signaling can populate through network as normal.

Example described in this section is based on network configurations illustrated using the following figure:



Example: PCECC Initiated LSP Setup In the Network Migration

In this example, there are five nodes for the TE LSP from head end (Node1) to the tail end (Node5). Where the Node4 and Node5 are centrally controlled and other nodes are legacy nodes.

- Node1 sends a path request message for the setup of LSP destinating to Node5.

- PCECC sends to node1 a reply message for LSP setup with the path: (Node1, if1),(Node2, if2), (Node3, if3), (Node4, if4), Node5.

- Node1, Node2, Node3 will setup the LSP to Node5 using the local labels as usual. Node 3 with help of PCECC could proxy the signaling.

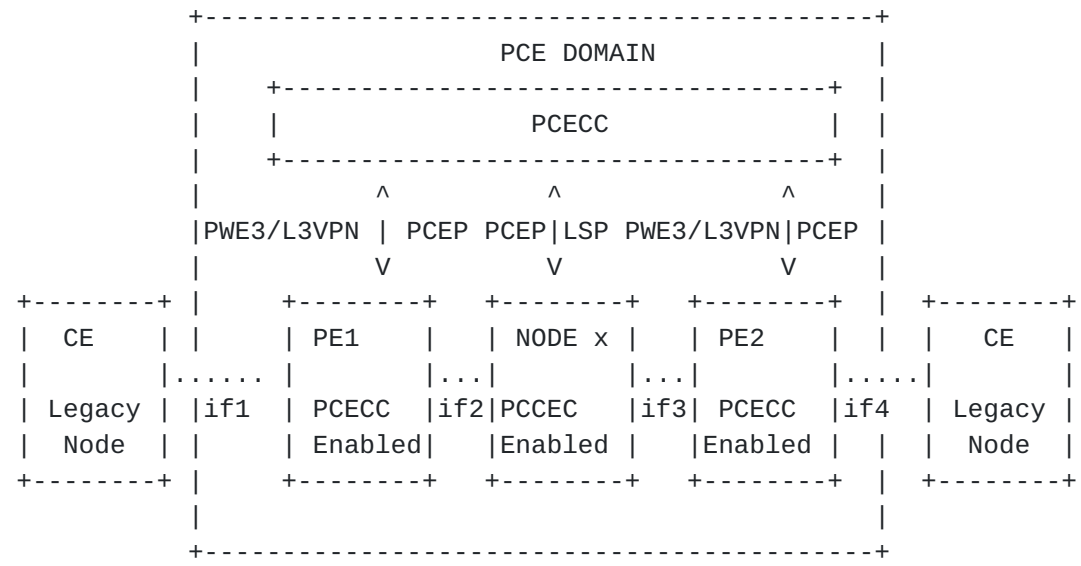
- Then the PCECC will program the out-segment of Node3, the in-segment/ out-segment of Node4, and the in-segment for Node5.

3.6. Use Cases of PCECC for L3VPN and PWE3

As described in [RFC8283], various network services may be offered over a network. These include protection services (including Virtual Private Network (VPN) services (such as Layer 3 VPNs [RFC4364] or Ethernet VPNs [RFC7432]); or Pseudowires [RFC3985]. Delivering services over a network in an optimal way requires coordination in the way that network resources are allocated to support the services. A PCE-based central controller can consider the whole network and all components of a service at once when planning how to deliver the service. It can then use PCEP to manage the network resources and to install the necessary associations between those resources.

In the case of L3VPN, VPN labels can be assigned and distributed through the PCECC PCEP among the PE router instead of using the BGP protocols.

Example described in this section is based on network configurations illustrated using the following figure:



Example: Using PCECC for L3VPN and PWE3

In the case PWE3, instead of using the LDP signaling protocols, the label and port pairs assigned to each pseudowire can be assigned through PCECC among the PE routers and the corresponding forwarding entries will be distributed into each PE routers through the extended PCEP protocols and PCECC mechanism.

3.7. Using PCECC for Traffic Classification Information

As described in [RFC8283], traffic classification is an important part of traffic engineering. It is the process of looking at a packet to determine how it should be treated as it is forwarded through the network. It applies in many scenarios including MPLS traffic engineering (where it determines what traffic is forwarded onto which LSPs); segment routing (where it is used to select which set of forwarding instructions to add to a packet); and SFC (where it indicates along which service function path a packet should be forwarded). In conjunction with traffic engineering, traffic classification is an important enabler for load balancing. Traffic classification is closely linked to the computational elements of planning for the network functions just listed because it determines how traffic load is balanced and distributed through the network. Therefore, selecting what traffic classification should be performed by a router is an important part of the work done by a PCECC.

Instructions can be passed from the controller to the routers using PCEP. These instructions tell the routers how to map traffic to paths or connections. Refer [I-D.ietf-pce-pcep-flowspec].

Along with traffic classification, there are few more question that needs to be considered once the path is setup -

- how to use it
- Whether it is a virtual link
- Whether to advertise it in the IGP as a virtual link
- What bits of this information to signal to the tail end

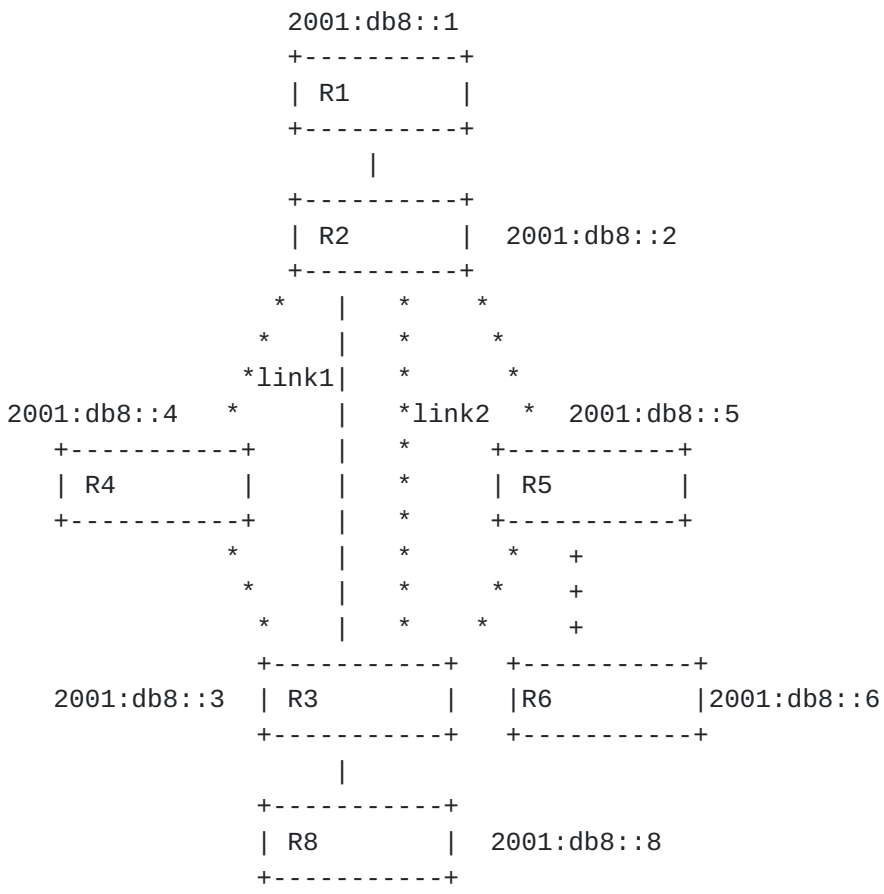
These are out of scope of this document.

3.8. Use Cases of PCECC for SRv6

As per [RFC8402], with Segment Routing (SR), a node steers a packet through an ordered list of instructions, called segments. Segment Routing can be applied to the IPv6 architecture with the Segment Routing Header (SRH) [I-D.ietf-6man-segment-routing-header]. A segment is encoded as an IPv6 address. An ordered list of segments is encoded as an ordered list of IPv6 addresses in the routing header. The active segment is indicated by the Destination Address of the packet. Upon completion of a segment, a pointer in the new routing header is incremented and indicates the next segment.

As per [I-D.ietf-6man-segment-routing-header], an SRv6 Segment is a 128-bit value. "SRv6 SID" or simply "SID" are often used as a shorter reference for "SRv6 Segment". Further details are in An illustration is provided in [I-D.filsfils-spring-srv6-network-programming] where SRv6 SID is represented as LOC:FUNCT.

[I-D.ietf-pce-segment-routing-ipv6] extends [I-D.ietf-pce-segment-routing] to support SR for IPv6 data plane. Further a PCECC could be extended to support SRv6 SID allocation and distribution.



In this case, PCECC could assign the SRv6 SID (in form of a IPv6 address) to be used for node and adjacency. Later SRv6 path in form of list of SRv6 SID could be used at the ingress. Some examples -

- SRv6 SID-List={2001:db8::8} - The best path towards R8

- SRv6 SID-List={2001:db8::5, 2001:db8::8} - The path towards R8 via R5

3.9. Use Cases of PCECC for SFC

Service Function Chaining (SFC) is described in [RFC7665]. It is the process of directing traffic in a network such that it passes through specific hardware devices or virtual machines (known as service function nodes) that can perform particular desired functions on the traffic. The set of functions to be performed and the order in which they are to be performed is known as a service function chain. The chain is enhanced with the locations at which the service functions are to be performed to derive a Service Function Path (SFP). Each packet is marked as belonging to a specific SFP, and that marking lets each successive service function node know which functions to perform and to which service function node to send the packet next. To operate an SFC network, the service function nodes must be configured to understand the packet markings, and the edge nodes must be told how to mark packets entering the network. Additionally, it may be necessary to establish tunnels between service function nodes to carry the traffic. Planning an SFC network requires load balancing between service function nodes and traffic engineering across the network that connects them. As per [RFC8283], these are operations that can be performed by a PCE-based controller, and that controller can use PCEP to program the network and install the service function chains and any required tunnels.

PCECC can play the role for setting the traffic classification rules at the classifier as well as downloading the forwarding instructions to the SFFs so that they could process the NSH and forward accordingly.

[Editor's Note - more details to be added]

3.10. Use Cases of PCECC for Native IP

[I-D.ietf-teas-native-ip-scenarios] describes the scenarios, and suggestions for the "Centrally Control Dynamic Routing (CCDR)" architecture, which integrates the merit of traditional distributed protocols (IGP/BGP), and the power of centrally control technologies (PCE/SDN) to provide one feasible traffic engineering solution in various complex scenarios for the service provider. [I-D.ietf-teas-pce-native-ip] defines the framework for CCDR traffic engineering within Native IP network, using Dual/Multi-BGP session strategy and CCDR architecture. PCEP protocol can be used to transfer the key parameters between PCE and the underlying network devices (PCC) using PCECC technique. The central control instructions from PCECC to identify which prefix should be advertised on which BGP session.

3.11. Use Cases of PCECC for Local Protection (RSVP-TE)

[I-D.cbrt-pce-stateful-local-protection] describes the need for the PCE to maintain and associate the local protection paths for the RSVP-TE LSP. Local protection requires the setup of a bypass at the PLR. This bypass can be PCC-initiated and delegated, or PCE-initiated. In either case, the PLR MUST maintain a PCEP session to the PCE. The Bypass LSPs need to mapped to the primary LSP. This could be done locally at the PLR based on a local policy but there is a need for a PCE to do the mapping as well to exert greater control.

This mapping can be done via PCECC procedures where the PCE could instruct the PLR to the mapping and identify the primary LSP for which bypass should be used.

3.12. Use Cases of PCECC for BIER

Bit Index Explicit Replication (BIER) [RFC8279] defines an architecture where all intended multicast receivers are encoded as a bitmask in the multicast packet header within different encapsulations. A router that receives such a packet will forward the packet based on the bit position in the packet header towards the receiver(s) following a precomputed tree for each of the bits in the packet. Each receiver is represented by a unique bit in the bitmask.

BIER-TE [I-D.ietf-bier-te-arch] shares architecture and packet formats with BIER. BIER-TE forwards and replicates packets based on a BitString in the packet header, but every BitPosition of the BitString of a BIER-TE packet indicates one or more adjacencies. BIER-TE Path can be derived from a PCE and used at the ingress as described in [I-D.chen-pce-bier].

Further, PCECC mechanims could be used for the allocation of bits for the BIER router for BIER as well as for the adjacencies for BIER-TE. PCECC based controller can use PCEP to instruct the BIER capable routers the meaning of the bits as well as other fields needed for BIER encapsulation.

[Editor's Note - more details to be added]

4. IANA Considerations

This document does not require any action from IANA.

5. Security Considerations

TBD.

6. Acknowledgments

We would like to thank Adrain Farrel, Aijun Wang, Robert Tao, Changjiang Yan, Tieying Huang, Sergio Belotti, Dieter Beller, Andrey Elperin and Evgeniy Brodskiy for their useful comments and suggestions.

7. References

7.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

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

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8283] Farrel, A., Ed., Zhao, Q., Ed., Li, Z., and C. Zhou, "An Architecture for Use of PCE and the PCE Communication Protocol (PCEP) in a Network with Central Control", RFC 8283, DOI 10.17487/RFC8283, December 2017, <https://www.rfc-editor.org/info/rfc8283>.

7.2. Informative References

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

[RFC4206] Kompella, K. and Y. Rekhter, "Label Switched Paths (LSP) Hierarchy with Generalized Multi-Protocol Label Switching (GMPLS) Traffic Engineering (TE)", RFC 4206, DOI 10.17487/RFC4206, October 2005, <https://www.rfc-editor.org/info/rfc4206>.

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

[RFC5150] Ayyangar, A., Kompella, K., Vasseur, JP., and A. Farrel, "Label Switched Path Stitching with Generalized Multiprotocol Label Switching Traffic Engineering (GMPLS TE)", RFC 5150, DOI 10.17487/RFC5150, February 2008, <https://www.rfc-editor.org/info/rfc5150>.

[RFC5151] Farrel, A., Ed., Ayyangar, A., and JP. Vasseur, "Inter-Domain MPLS and GMPLS Traffic Engineering -- Resource Reservation Protocol-Traffic Engineering (RSVP-TE) Extensions", RFC 5151, DOI 10.17487/RFC5151, February 2008, <https://www.rfc-editor.org/info/rfc5151>.

[RFC5541] Le Roux, JL., Vasseur, JP., and Y. Lee, "Encoding of Objective Functions in the Path Computation Element Communication Protocol (PCEP)", RFC 5541, DOI 10.17487/RFC5541, June 2009, <https://www.rfc-editor.org/info/rfc5541>.

[RFC5376] Bitar, N., Zhang, R., and K. Kumaki, "Inter-AS Requirements for the Path Computation Element Communication Protocol (PCECP)", RFC 5376, DOI 10.17487/RFC5376, November 2008, <https://www.rfc-editor.org/info/rfc5376>.

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

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

[RFC8231] Crabbe, E., Minei, I., Medved, J., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE", RFC 8231, DOI 10.17487/RFC8231, September 2017, <https://www.rfc-editor.org/info/rfc8231>.

[RFC8279] Wijnands, IJ., Ed., Rosen, E., Ed., Dolganow, A., Przygienda, T., and S. Aldrin, "Multicast Using Bit Index Explicit Replication (BIER)", RFC 8279, DOI 10.17487/RFC8279, November 2017, <https://www.rfc-editor.org/info/rfc8279>.

[RFC8281] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for PCE-Initiated LSP Setup in a Stateful PCE Model", RFC 8281, DOI 10.17487/RFC8281, December 2017, <https://www.rfc-editor.org/info/rfc8281>.

[RFC8355] Filsfils, C., Ed., Previdi, S., Ed., Decraene, B., and R. Shakir, "Resiliency Use Cases in Source Packet Routing in Networking (SPRING) Networks", RFC 8355, DOI 10.17487/RFC8355, March 2018, <https://www.rfc-editor.org/info/rfc8355>.

[RFC8402] Filsfils, C., Ed., Previdi, S., Ed., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", RFC 8402, DOI 10.17487/RFC8402, July 2018, <https://www.rfc-editor.org/info/rfc8402>.

[I-D.ietf-pce-segment-routing] Sivabalan, S., Filsfils, C., Tantsura, J., Henderickx, W., and J. Hardwick, "PCEP Extensions for Segment Routing", draft-ietf-pce-segment-routing-16 (work in progress), March 2019.

[I-D.ietf-pce-stateful-hpce] Dhody, D., Lee, Y., Ceccarelli, D., Shin, J., and D. King, "Hierarchical Stateful Path Computation Element (PCE).", draft-ietf-pce-stateful-hpce-10 (work in progress), June 2019.

[I-D.ietf-pce-pcep-flowspec] Dhody, D., Farrel, A., and Z. Li, "PCEP Extension for Flow Specification", draft-ietf-pce-pcep-flowspec-03 (work in progress), February 2019.

[I-D.ietf-pce-pcep-extension-for-pce-controller] Zhao, Q., Li, Z., Negi, M., and C. Zhou, "PCEP Procedures and Protocol Extensions for Using PCE as a Central Controller (PCECC) of LSPs", draft-ietf-pce-pcep-extension-for-pce-controller-01 (work in progress), February 2019.

[I-D.zhao-pce-pcep-extension-pce-controller-sr]
Zhao, Q., Li, Z., Negi, M., and C. Zhou, "PCEP Procedures and Protocol Extensions for Using PCE as a Central Controller (PCECC) of SR-LSPs", draft-zhao-pce-pcep-extension-pce-controller-sr-04 (work in progress), February 2019.

[I-D.li-pce-controlled-id-space]
Li, C., Chen, M., Dong, J., Li, Z., Wang, A., Cheng, W., and C. Zhou, "PCE Controlled ID Space", draft-li-pce-controlled-id-space-03 (work in progress), June 2019.

[I-D.dugeon-pce-stateful-interdomain]
Dugeon, O., Meuric, J., Lee, Y., and D. Ceccarelli, "PCEP Extension for Stateful Inter-Domain Tunnels", draft-dugeon-pce-stateful-interdomain-02 (work in progress), March 2019.

[I-D.cbrt-pce-stateful-local-protection]
Barth, C. and R. Torvi, "PCEP Extensions for RSVP-TE Local-Protection with PCE-Stateful", draft-cbrt-pce-stateful-local-protection-01 (work in progress), June 2018.

[I-D.filsfils-spring-srv6-network-programming]
Filsfils, C., Camarillo, P., Leddy, J., daniel.voyer@bell.ca, d., Matsushima, S., and Z. Li, "SRv6 Network Programming", draft-filsfils-spring-srv6-network-programming-07 (work in progress), February 2019.

[I-D.ietf-pce-segment-routing-ipv6]
Negi, M., Li, C., Sivabalan, S., Kaladharan, P., and Y. Zhu, "PCEP Extensions for Segment Routing leveraging the IPv6 data plane", draft-ietf-pce-segment-routing-ipv6-02 (work in progress), April 2019.

[I-D.ietf-6man-segment-routing-header]
Filsfils, C., Dukes, D., Previdi, S., Leddy, J., Matsushima, S., and d. daniel.voyer@bell.ca, "IPv6 Segment Routing Header (SRH)", draft-ietf-6man-segment-routing-header-21 (work in progress), June 2019.

[I-D.ietf-teas-pce-native-ip]
Wang, A., Zhao, Q., Khasanov, B., Chen, H., and R. Mallya, "PCE in Native IP Network", draft-ietf-teas-pce-native-ip-03 (work in progress), April 2019.

[I-D.ietf-teas-native-ip-scenarios]
Wang, A., Huang, X., Qou, C., Li, Z., and P. Mi, "Scenarios and Simulation Results of PCE in Native IP Network", draft-ietf-teas-native-ip-scenarios-06 (work in progress), June 2019.

[I-D.ietf-bier-te-arch]
Eckert, T., Cauchie, G., Braun, W., and M. Menth, "Traffic Engineering for Bit Index Explicit Replication (BIER-TE)", draft-ietf-bier-te-arch-02 (work in progress), May 2019.

[I-D.chen-pce-bier]
Chen, R. and Z. Zhang, "PCEP Extensions for BIER", draft-chen-pce-bier-05 (work in progress), March 2019.

[MAP-REDUCE]
Lee, K., Choi, T., Ganguly, A., Wolinsky, D., Boykin, P., and R. Figueiredo, "Parallel Processing Framework on a P2P System Using Map and Reduce Primitives", , may 2011, <http://leeky.me/publications/mapreduce_p2p.pdf>.

[MPLS-DC] Afanasiev, D. and D. Ginsburg, "MPLS in DC and inter-DC networks: the unified forwarding mechanism for network programmability at scale", , march 2014, <https://www.slideshare.net/DmitryAfanasiev1/yandex-nag201320131031>.

7.3. URIs

[1] https://hadoop.apache.org/

Appendix A. Using reliable P2MP TE based multicast delivery for distributed computations (MapReduce-Hadoop)

MapReduce model of distributed computations in computing clusters is widely deployed. In Hadoop [1] 1.0 architecture MapReduce operations on big data performs by means of Master-Slave architecture in the Hadoop Distributed File System (HDFS), where NameNode has the knowledge about resources of the cluster and where actual data (chunks) for particular task are located (which DataNode). Each chunk of data (64MB or more) should have 3 saved copies in different DataNodes based on their proximity.

Proximity level currently has semi-manual allocation and based on Rack IDs (Assumption is that closer data are better because of access speed/smaller latency).

JobTracker node is responsible for computation tasks, scheduling across DataNodes and also have Rack-awareness. Currently transport protocols between NameNode/JobTracker and DataNodes are based on IP unicast. It has simplicity as pros but has numerous drawbacks related with its flat approach.

It is clear that we should go beyond of one DC for Hadoop cluster creation and move towards distributed clusters. In that case we need to handle performance and latency issues. Latency depends on speed of light in fiber links and also latency introduced by intermediate devices in between. The last one is closely correlated with network device architecture and performance. Current performance of NPU based routers should be enough for creating distribute Hadoop clusters with predicted latency. Performance of SW based routers (mainly as VNF) together with additional HW features such as DPDK are promising but require additional research and testing.

Main question is how can we create simple but effective architecture for distributed Hadoop cluster?

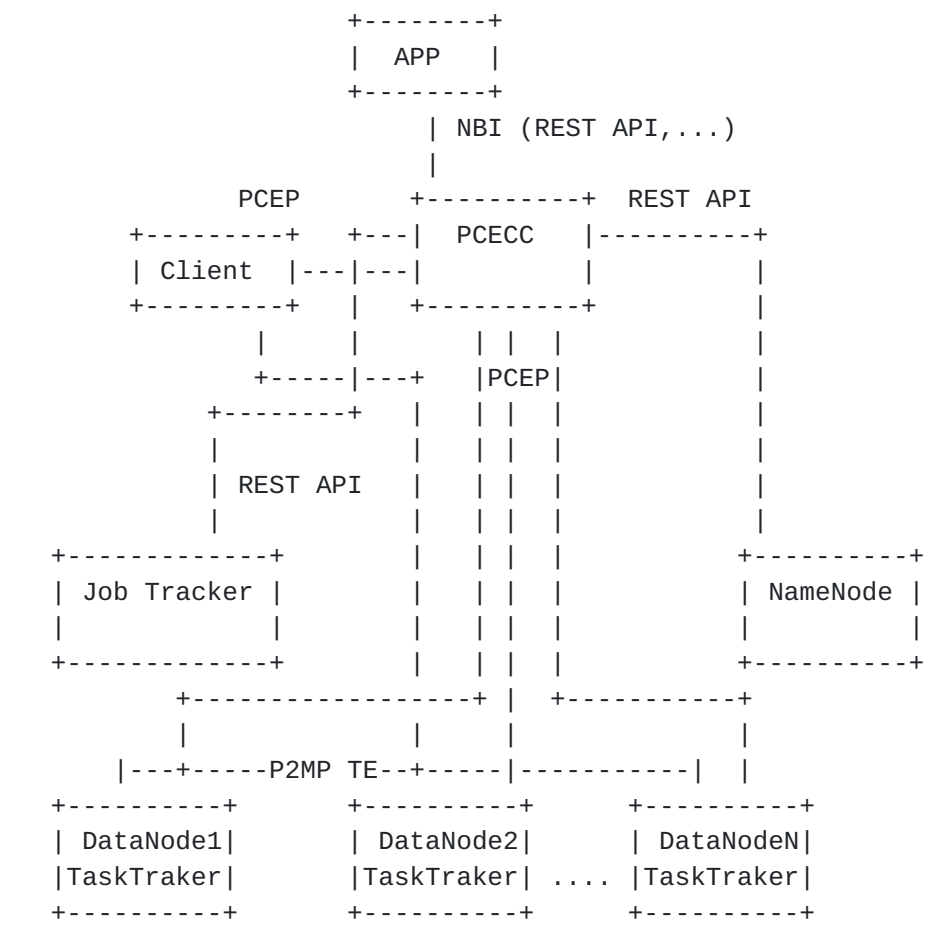
There is research [MAP-REDUCE] which show how usage of multicast tree could improve speed of resource or cluster members discovery inside the cluster as well as increase redundancy in communications between cluster nodes.

Is traditional IP based multicast enough for that? We doubt it because it requires additional control plane (IGMP, PIM) and a lot of signaling, that is not suitable for high performance computations, that are very sensitive to latency.

P2MP TE tunnels looks much more suitable as potential solution for creation of multicast based communications between Master and Slave nodes inside cluster. Obviously these P2MP tunnels should be dynamically created and turned down (no manual intervention). Here, the PCECC comes to play with main objective to create optimal topology of each particular request for MapReduce computation and also create P2MP tunnels with needed parameters such as bandwidth and delay.

This solution would require to use MPLS label based forwarding inside the cluster. Usage of label based forwarding inside DC was proposed by Yandex [MPLS-DC]. Technically it is already possible because MPLS on switches is already supported by some vendors, MPLS also exists on Linux and OVS.

The following framework can make this task:



Communication between Master nodes (JobTracker and NameNode) and PCECC via REST API MAY be either done directly or via cluster manager such as Mesos.

Phase 1: Distributed cluster resources discovery During this phase Master Nodes SHOULD identify and find available Slave nodes according to computing request from application (APP). NameNode SHOULD query PCECC about available DataNodes, NameNode MAY provide additional constrains to PCECC such as topological proximity, redundancy level.

PCECC SHOULD analyze the topology of distributed cluster and perform constrain based path calculation from client towards most suitable NameNodes. PCECC SHOULD reply to NameNode the list of most suitable DataNodes and their resource capabilities. Topology discovery mechanism for PCECC will be added later to that framework.

Phase 2: PCECC SHOULD create P2MP LSP from client towards those DataNodes by means of PCEP messages following previously calculated path.

Phase 3. NameNode SHOULD send this information to client, PCECC informs client about optimal P2MP path towards DataNodes via PCEP message.

Phase 4. Client sends data blocks to those DataNodes for writing via created P2MP tunnel.

When this task will be finished, P2MP tunnel could be turned down.

Authors' Addresses

Quintin Zhao
Huawei Technologies
125 Nagog Technology Park
Acton, MA 01719
US

Email: quintinzhao@gmail.com

Zhenbin (Robin) Li
Huawei Technologies
Huawei Bld., No.156 Beiqing Rd.
Beijing 100095
China

Email: lizhenbin@huawei.com

Boris Khasanov
Huawei Technologies
Moskovskiy Prospekt 97A
St.Petersburg 196084
Russia

Email: khasanov.boris@huawei.com

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

Email: dhruv.ietf@gmail.com

King Ke
Tencent Holdings Ltd.
Shenzhen
China

Email: kinghe@tencent.com

Luyuan Fang
Expedia, Inc.
USA

Email: luyuanf@gmail.com

Chao Zhou
Cisco Systems

Email: chao.zhou@cisco.com

Boris Zhang
Telus Communications

Email: Boris.zhang@telus.com

Artem Rachitskiy
Mobile TeleSystems JLLC
Nezavisimosti ave., 95
Minsk 220043
Belarus

Email: arachitskiy@mts.by

Anton Gulida
LLC "Lifetech"
Krasnoarmeyskaya str., 24
Minsk 220030
Belarus

Email: anton.gulida@life.com.by