IETF Internet Draft PCE Working Group
Proposed Status: Informational
Expires: September 2005

Adrian Farrel
Old Dog Consulting
Jean-Philippe Vasseur
Cisco Systems, Inc.
Jerry Ash
AT&T
March 2005

draft-ietf-pce-architecture-00.txt

Path Computation Element (PCE) Architecture

Status of this Memo

Abstract

Constraint-based path computation is a fundamental building block for traffic engineering systems such as Multiprotocol Label Switching (MPLS) and Generalized Multiprotocol Label Switching (GMPLS) networks. Path computation in large, multi-domain, multi-region or multi-layer networks is highly complex and may require special computational components and cooperation between the different network domains.

This document specifies the architecture for a Path Computation Element (PCE)-based model to address this problem space. This document does not attempt to provide a detailed description of all the architectural

components, but rather it describes a set of building blocks for the PCE architecture from which solutions may be constructed.

Table of Contents

1. Introduction

Constraint-based path computation is a fundamental building block for traffic engineering in MPLS and GMPLS networks. Path computation in large, multi-domain networks is highly complex and may require special computational components and cooperation between the different domains. This document specifies the architecture for a Path Computation Element (PCE)-based model to address this problem space.

This document does not attempt to provide a detailed description of all the architectural components. Rather, it describes a set of building blocks for the PCE architecture from which solutions may be constructed. For example, it discusses PCE-based implementations including composite, external, and multiple PCE path computation. Furthermore, it discusses architectural considerations including centralized computation, distributed computation, synchronization, PCE discovery and load balancing, detecting PCE liveness, PCC-PCE and PCE-PCE communication, TED synchronization, stateful and stateless PCEs, monitoring, policy and confidentiality, and evaluation metrics.

2. Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

3. Terminology

CSPF: Constraint-based Shortest Path First.

LER: Label Edge Router.

LSDB: Link State Database.

LSP: Label Switched Path.

LSR: Label Switching Router.

PCC: Path Computation Client : any client application requesting a path computation to be performed by the 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 (see further description in section 3).

TED: Traffic Engineering Database which contains the topology and resource information of the domain. The TED may be fed by IGP extensions or potentially by other means.

TE LSP: Traffic Engineering MPLS Label Switched Path.

---

4. Definitions

A Path Computation Element (PCE) is an entity that is capable of computing a network path or route based on a network graph, and applying computational constraints. The PCE entity is an application that can be located within a network node or component, on an out-of-network server, etc. For example, a PCE would be able to compute the path of a TE LSP by operating on the TED and considering the bandwidth and other constraints applicable to the TE LSP service request.

A domain is any collection of network elements within a common sphere of address management or path computational responsibility. Examples of domains include IGP areas, Autonomous Systems (ASs), multiple ASs within a service provider network, or multiple ASs across multiple service provider networks. However, domains of computational responsibility may also exist as sub-domains of areas or ASs.

In order to fully characterize a PCE and clarify these definitions, the following important considerations must also be examined:

1) Path computation is applicable in both intra-domain, inter-domain, and inter-layer contexts. Inter-domain path computation may involve the correlation of topology and routing information between domains. Inter-layer path computation refers to the use of PCE where multiple layers are involved and when the objective is to perform path computation at one or multiple layers while taking into account topology and resource information at these layers. Overlapping domains are not within the scope of this document. In the inter-domain case, the domains may belong to a single or multiple Service Providers.

2) In "single PCE path computation," a single PCE is used to compute a given path in a domain. In "multiple PCE path computation," multiple PCEs are used to compute a given path in a domain.

3) "Centralized computation model" refers to a model whereby all paths in a domain are computed by a single, centralized PCE. Conversely, "Distributed computation model" refers to the computation of paths in a domain being shared among multiple PCEs. Paths that span multiple domains may be computed using the distributed model with a PCE responsible for each domain, or the centralized model by defining a domain that encompasses all of the other domains. From these definitions, a centralized computation model inherently uses single PCE path computation. However, a distributed computation model could use either single PCE path computation or multiple PCE path computations. There would be no such thing as a centralized model which uses multiple PCE path computations.

4) The PCE may or may not be located at the head-end of the path. For example, a conventional intra-domain solution is to have path computation performed by the head-end LSR of an MPLS TE LSP; in this

---

case, the head-end LSR contains a PCE. But solutions also exist where other nodes on the path must contribute to the path computation (for example, loose hops) making them PCEs in their own right. At the same time, the path computation may be made by some other PCE physically distinct from the computed path.

5) The path computed by the PCE may be an 'explicit PCE path' (that is, the full explicit path from start to destination, made of a list of strict hops) or a 'strict/loose PCE path' (that is, a mix of strict and loose hops comprising of at least one loose hop representing the destination), where a hop may be an abstract node such as an AS.

6) A PCE-based path computation model does not mean to be exclusive and can be used in conjunction with other path computation models. For instance, the path of an inter-AS TE LSP may be computed using a PCE-based path computation model in some IGP areas, whereas the set of traversed ASes may be specified by other means (not determined by any PCE).

7) This document does not make any assumptions about the nature or implementation of a PCE. A PCE could be implemented on a router, an LSR, a dedicated network server, etc. Moreover, the PCE function is orthogonal to the forwarding capability of the node on which it is implemented.

5. Motivation for a PCE-based Architecture

Several motivations for a PCE-based architecture (described in section 5) are listed below. This list is not meant to be exhaustive and is provided for the sake of illustration.

It should be highlighted that the aim of this section is to provide some application examples for which a PCE-based path may be suitable: this also clearly states that such a model does not aim to replace existing path computation model but would apply to specific existing situations.

5.1. CPU-intensive Path Computation/Global Optimization

There are many situations where the computation of a path may be highly CPU-intensive: examples of CPU-intensive path computations include the resolutions of NP-complete problems such as:

- Global optimization in placing a set of TE LSPs within a domain so as to optimize an objective function (for example, minimization of the maximum link utilization)

- Multi-criteria path computation (for example, delay and link utilization, inclusion of switching capabilities, adaptation features, encoding types and optical constraints within a GMPLS optical network)

- Computation of minimal cost Point to Multipoint trees (Steiner trees).

In these situations, it may not be possible or desirable for a router to perform path computation because of the constraints on its CPU, in which case the path computation may be off-loaded to some other PCE(s).

5.2. Partial Visibility

There are several scenarios where the node responsible for path computation has limited visibility of the network topology to the destination. This limitation may occur, for instance, when an ingress router attempts to establish an LSP to a destination that lies in a separate domain, since TE information is not exchanged across the domain boundaries. In such cases, it is possible to use loose routes to establish the LSP, relying on routers at the domain borders to establish the next piece of the path, however, it is not possible to guarantee that the optimal (shortest) path will be used, nor even that a viable path will be discovered except, possibly, through repeated trial and

error using crankback or other signaling extensions.

This problem of inter-domain path computation may most probably be addressed through distributed computation with cooperation among PCEs within each of the domains, or perhaps by using a central "all-seeing" PCE. In this latter case there are challenges of scalability (both the size of the TED and the responsiveness of a single PCE handling requests for many domains) and of preservation of confidentiality when the domains belong to different Service Providers.

Note that the issues described here can be further highlighted in the context of LSP re-optimization, or the establishment of multiple diverse LSPs for protection or load sharing.

5.3. Absence of the TED or use of Non-TE-Enabled IGP

The traffic engineering database (TED) may be a large drain on the resources of a network node (such as an edge router or LER) both from a memory perspective and because it may require non-negligible CPU activity to maintain. The use of a distinct PCE may be appropriate in such circumstances, and a separate node can be used to establish and maintain the TED, and to make it available for path computation.

The IGPs run within some networks are not sufficient to build a full TED. For example, a network may run OSPF/IS-IS without the OSPF-TE/ISIS-TE extensions, or some routers in the network may not support the TE extensions. In these cases, in order to successfully compute paths through the network, the TED must be constructed or supplemented through configuration action, and updated as network resources are reserved or released. Such a TED could be distributed to each router so that each router can perform path computation, or held centrally (on a distinct node that supports PCE) for centralized path

---

computation.

5.4. Node Outside the Routing Domain

An LER might not be part of the routing domain for administrative reasons (for example, a customer-edge (CE) router connected to the provider-edge (PE) router in the context of MPLS VPN [RFC2547] and for which it is desired to provide a CE to CE TE LSP path).

This scenario suggests a solution that does not involve doing computation on the ingress router, and that does not rely on static

loose hops configuration in which case optimal shortest paths could not be achieved. A distinct PCE-based solution can help here. Note that the PCE in this case may, itself, provide a path that includes loose hops.

5.5. Network Element Lacks Control Plan or Routing Capability

It is common in legacy optical networks for the network elements not to have a control plane or routing capability. On such network elements there only exists the data plane and management plane, and all cross-connections are made from the management plane. It is desirable in this case to run the path computation on the PCE, and send the cross-connection commands to each node on the computed path. This scenario is important for ASON-capable networks, and may also be used for interworking between GMPLS-capable and GMPLS-incapable networks.

5.6. Backup Path Computation for Bandwidth Protection

A PCE can be used to compute backup paths in the context of fast reroute protection of TE-LSPs. In this model all backup TE-LSPs protecting a given facility are computed in a coordinated manner by a PCE. This allows ensuring complete bandwidth sharing between bypass tunnels protection independent elements, while avoiding any extensions to LSP signaling. Both centralized and distributed computation models are applicable. In the distributed case each LSR can be a PCE to compute its own protection.

5.7. Multi-Layer Networks

A server-layer network of one switching capability may support multiple networks of another (more granular) switching capability. For example, a TDM network may provide connectivity for client-layer networks such as IP, MPLS or Layer 2 [MRN].

The server-layer network is unlikely to provide the same connectivity paradigm as the client networks so that bandwidth granularity in the server-layer network may be much coarser than in the client-layer network. Similarly, there is likely to be a management separation between the two networks providing independent address spaces. Further, where multiple client-layer networks make use of the same server-layer network, those client-layer networks may have

---

independent policies, control parameters, address spaces and routing preferences.

The different client and server layer networks may be considered as distinct path computation regions within a PCE domain, and so the PCE architecture is
useful to allow path computation from one client-layer network region, across the server-layer network to another client-layer network region.

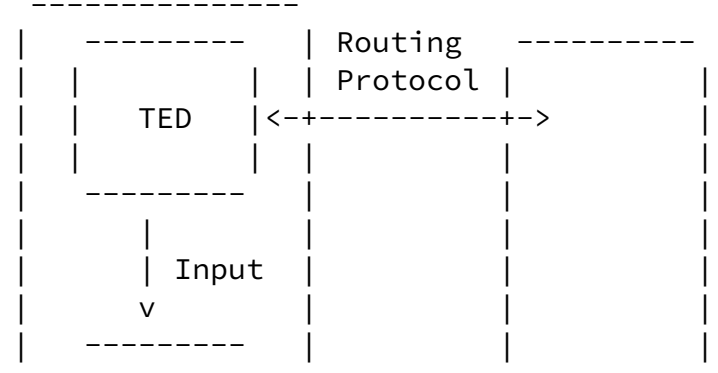
In this case, the PCE is responsible for resolving address space issues, handling differences in policy and control parameters, and coordinating resources between the networks. Note that, because of the differences in bandwidth granularity, connectivity across the server-layer network may be provided through virtual TE links or Forwarding Adjacencies: the PCE may offer a point of control responsible for the decision to provision new TE links or Forwarding Adjacencies across the server-layer network.

6. Overview of the PCE-Based Architecture

This section is intended to give an overview of the network architecture of the PCE model. It needs to be read in conjunction with the details provided in the next section to provide a full view of the flexibility of the model.

6.1. Composite PCE

Figure 1 below shows the components of a typical composite PCE node (that is, a router that also implements the PCE functionality) that utilizes path computation. The routing protocol is used to exchange TE information from which the TED is constructed. Service requests to provision TE LSPs are received by the node and converted into signaling requests, but these may first require path computation which is requested from a Path Computation Element, the PCE. The PCE operates on the TED in order to respond with the requested path.



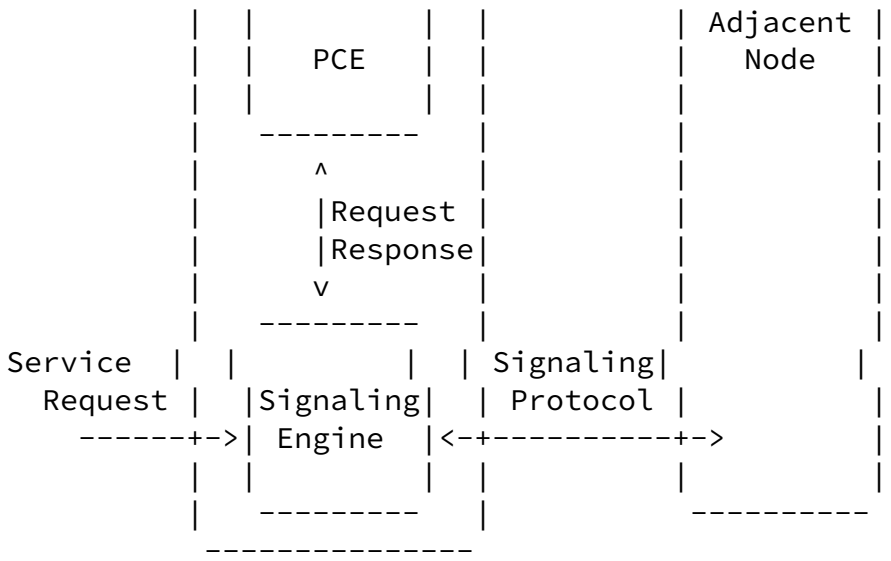
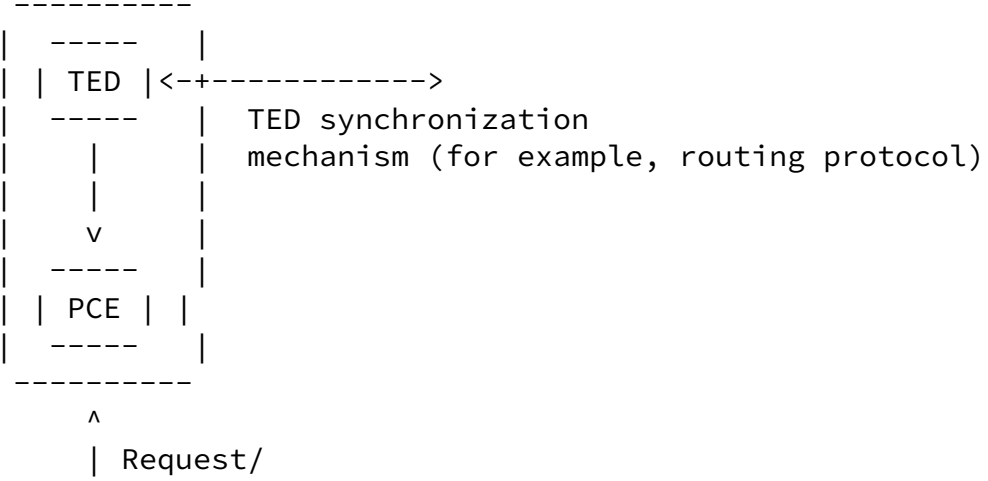


Figure 1. Composite PCE Node

Note that the routing adjacency between the composite PCE node and any other router may be performed by means of direct connectivity or any tunneling mechanism.

6.2. External PCE

Figure 2 shows PCE support that is external from the requesting network element. A service request is received by the head-end node and before it can signal to establish the service it makes a request to the external PCE for a path to be computed. The PCE makes the computation using the TED and returns a response.



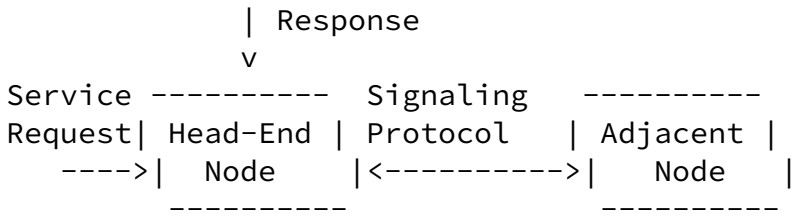


Figure 2. External PCE Node

Note that in this case, the node that supports the PCE function may also perform forwarding, but those functions are purely orthogonal.

6.3. Multiple PCE Path Computation

Figure 3 illustrates how multiple PCE path computations may be performed along the path of a signaled service. As in the previous example, the head-end PCC makes a request to an external PCE, but the path that is returned is such that the next network element finds it necessary to perform further computation. It consults another PCE to establish the next hop in the path.

Note that either or both PCEs in this case could be co-resident with the network node as in Section 5.1.

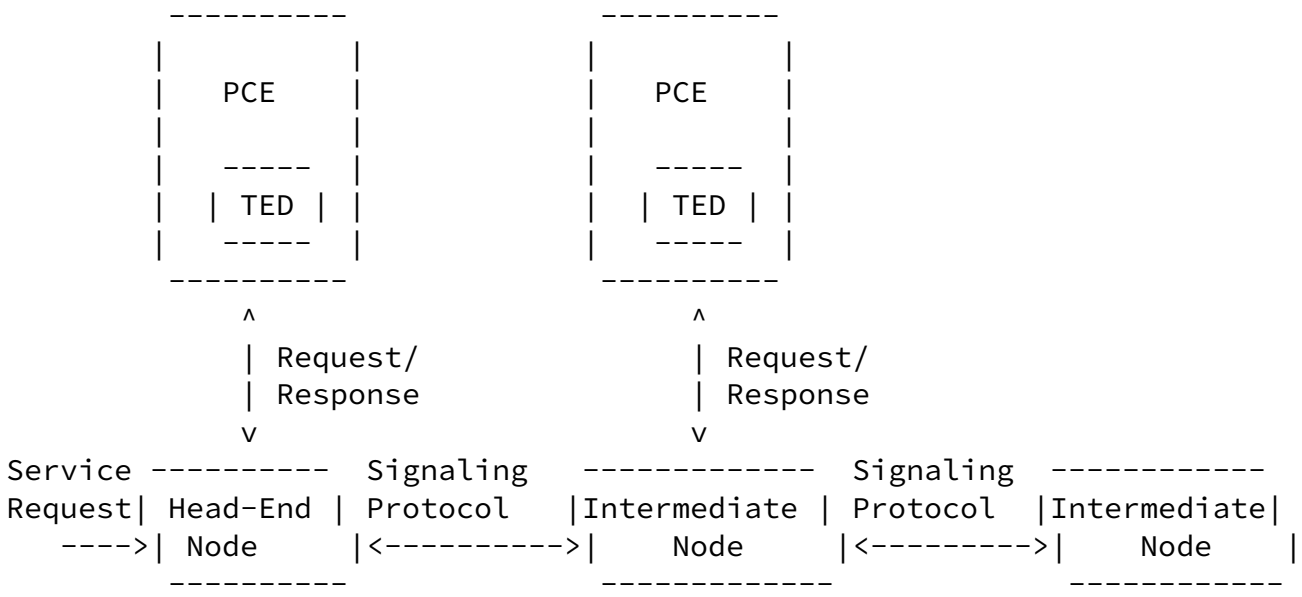


Figure 3. Multiple PCE Path Computation

6.4. Multiple PCE Path Computation with Inter-PCE Communication

The PCE in Section 5.3 was not able to supply a full path for the requested service and this resulted in the adjacent node needing to make its own computation request. As illustrated in Figure 4, the same problem is solved by introducing inter-PCE communication and cooperation between PCEs so that the PCE consulted by the head-end network node makes a request of another PCE to help with the computation.

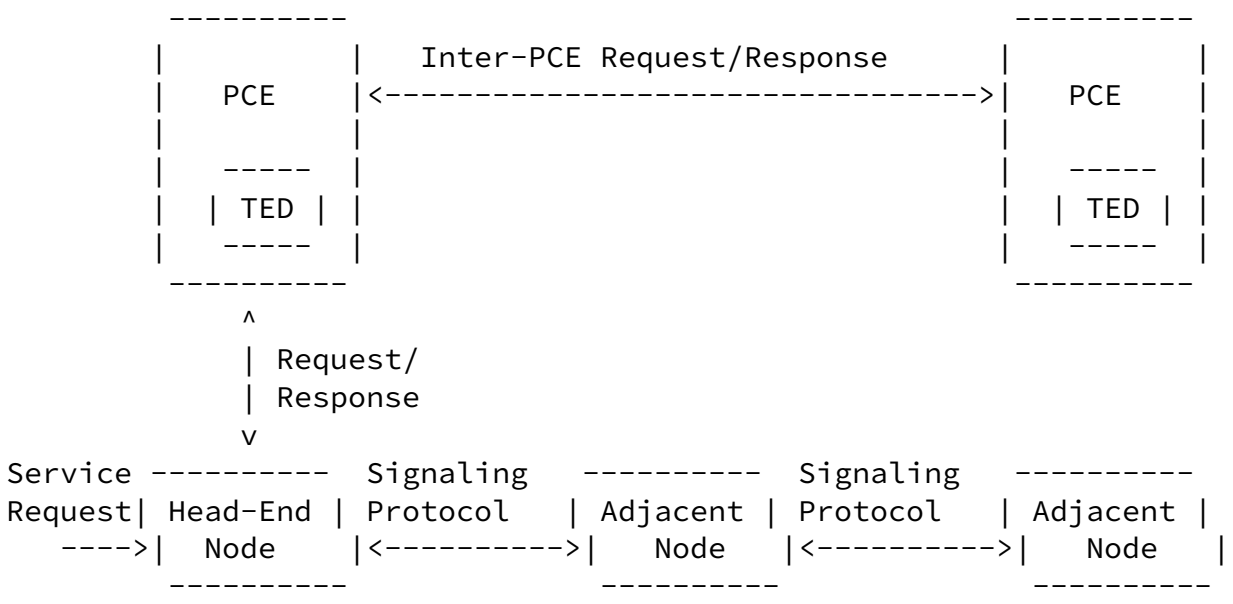


Figure 4. Multiple PCE Path Computation with Inter-PCE Communication

Multiple PCE path computation with inter-PCE communication involves coordination between distributed PCEs such that the result of the computation performed by one PCE depends on information supplied by other PCEs. PCE-PCE communication is discussed further in section 6.6.

Note that a PCC cannot see the difference between centralized computation, and multiple PCE path computation with inter-PCE communication. That is, the PCC network node or component that requests the computation makes a single request and receives a full or partial path in response, but the response is actually achieved through the coordinated, cooperative efforts of more than one PCE.

6.5 Areas for Standardization

According to the PCE charter, the following are the standardization

areas that the PCE working group will address:

- communication between PCCs and PCEs, and between cooperating PCEs
- requirements for extensions to existing routing and signaling protocols in support of PCE discovery and signaling of inter-domain paths
- definition of metrics to evaluate path quality, scalability, responsiveness and robustness of path computation models.

7. PCE Architectural Considerations

The aim of this section is to provide a list of the PCE architectural components. Specific realizations and implementation details (state machines or algorithms, etc.) of PCE-based solutions are out of the scope of this document.

Note also that PCE-based path computation does not affect in any way the use of the computed paths. For example, the use of PCE does not change the way in which Traffic Engineering LSPs are signaled, maintained and torn down, but strictly relates to the path computation aspects of such TE LSPs.

7.1. Centralized Computation Model

A "centralized computation model" considers that all path computations for a given domain will be performed by a single, centralized PCE. This may be a dedicated server (for example, an external PCE node), or a nominated router (for example, a composite PCE node) in the network. In this model, all PCCs in the domain would send their path computation requests to the central PCE. While a domain in this context might be an IGP area or AS, it might also be a sub-group of network nodes that is defined by its dependence on the PCE.

7.2. Distributed Computation Model

A "distributed computation model" refers to a domain or network that may include multiple PCEs, and where computation of paths is shared among the PCEs. A given path may in turn be computed by a single PCE ("single PCE path computation") or multiple PCEs ("multiple PCE path computation"). A PCC may be linked to a particular PCE, or may be able

to choose freely among several PCEs - the method of choice between PCEs is out of scope of this document, but see section 6.4. It will often be the case that the computation of an individual path is performed

entirely by a single PCE. For example, this is usually the case in MPLS TE within a single IGP where the ingress LSR/composite node is responsible for computing the path or for contacting an external PCE. Conversely, multiple PCE path computation implies the involvement of more than one PCE in the computation of a single path. An example of this is where loose hop expansion is performed by transit LSRs/composite nodes on an MPLS TE LSP. Another example is the use of multiple cooperative PCE involved in the computation of a single LSP path.

7.3. Synchronization

It is often the case that multiple paths need to be computed to support a single service (for example, for protection or load sharing).
A PCC that determines that it requires more than one path to be computed may send a series of individual requests to the PCE. In this case, the PCE may make multiple individual path computations to generate the set of paths - the resultant paths are non-synchronized and are exactly those that would have been generated had the PCC made multiple requests. In this case of non-synchronized path computation, the path computation of a set of TE LSPs can be shared among a set of PCEs (that is, one path computed by each PCE). Furthermore, each PCE can be backed up by one or more PCEs, should it fail.

Conversely, the PCC may issue a single request to the PCE asking for all of the paths. The PCE will then in turn perform the simultaneous computation of the set of requested path. Such synchronized computation usually provides more optimal results.

The involvement of more than one PCE in the computation of a series of paths is by its nature non-synchronized. However, a set of cooperating PCEs may be synchronized under the control of a single PCE. For example, a PCC may send a request to a PCE which invokes domain specific computations by other PCEs before supplying a result to the PCC.

It is desirable to add a parameter to the PCC-PCE protocol to request alternate paths should the primary path fail to complete. While alternate paths may not always be successful if the primary fails, including alternate paths in a PCE response could perhaps have less overhead than having the PCC make separate requests for a second path, third path, etc. This technique is used in some existing CSPF implementations.

7.4. PCE Discovery and Load Balancing

The PCE architecture requires that the PCC knows the location of one or more PCEs that it can use for the computation of a path. Such knowledge may come through a discovery mechanism that simply relies on local configuration, or can imply dynamic PCE discovery along with various

static (for example, Boolean capability) or dynamically computed variables (for example, computing resources). Proxy PCE advertisement whereby the existence of a PCE is advertised via a proxy PCE is a viable alternative, should the PCE be incapable of such advertisement itself. In this later case, it is a requirement for the proxy to adequately advertise the PCE status and capability in a timely and synchronized fashion.

In the event that multiple PCEs are available to serve a particular path computation request, the PCC must select a PCE to satisfy the request. The details of such a selection, in order for instance to efficiently share the computation load across multiple PCEs, is local to the PCC and out of the scope of this document.

A PCE SHOULD advertise its capabilities, such as:

- set of constraints that it can account for (diversity, SRLGs, Optical impairments, wavelength continuity, etc.)
- number of switching capability layers (and which ones)
- number of path selection criteria (and which ones)
- whether it is a stateless PCE or it can send updates about better paths that might be available in the future
- whether it can compute P2MP trees (and which types)
- whether it can ensure resource sharing between backup tunnels

This information would help a PCC that dynamically learns about PCEs available on the network to decide which of them to use. Alternatively, a PCC might ask a PCE to perform a particular type of service and receive a response that says it is unable to perform the service, specify the things it can do. Note that the parameters mentioned above are not meant to be exhaustive and are listed for the sake of illustration.

7.5. Detecting PCE Liveness

The ability to detect a PCE's liveness is a mandatory piece of the overall architecture and could be achieved by several means. If some form of regular advertisement (such as through IGP extensions) is used for PCE discovery, it is expected that the PCE liveness will be determined by means of status advertisement (for example, IGP LSA/LSPs).

The failure of a PCE while processing a request, or the inability of a PCE to service a request (perhaps due to excessive load) may be determined by the PCC through the use of timers. This is particularly true in the case of inter-domain path computation where the PCE liveness may not be detected by means of the IGP. The detection of a PCE failure can be achieved by using the PCC-PCE protocol, much like the mechanisms

involving timers used in RSVP and LDP.

7.6. PCC-PCE & PCE-PCE Communication

Once the PCC has selected a PCE, and provided that the PCE is not local to the PCC, a request/response protocol is required for the PCC to communicate the path computation requests to the PCE and for the PCE to return the path computation response.

The path computation request may include a significant set of requirements including

- the source and destination of the path
- the bandwidth and other QoS parameters desired
- resources, resource affinities and shared risk link groups (SRLGs) to use/avoid
- the number of disjoint paths required and if near-disjoint paths are acceptable
- the level of robustness of the path resources
- and so on.

The level of robustness of the path resources covers a qualitative assessment of the vulnerability of the resources that may be used. For example, one might grade resources based on empirical evidence (mean time between failures), on known risks (there is major building work going on near this conduit), or on prejudice (vendor X's software is always crashing). A PCC could request that only robust resources be used, or allow any resource. Of course, this information does not comprise part of the TE information advertise by IGPs. It must come from somewhere else.

In case of a positive response from the PCE, one or more paths would be returned to the requesting node. In the event of a failure to compute the desired path(s), an error is returned together with as much information as possible about the reasons for the failure, and potentially advice about which constraints might be relaxed to be more likely to achieve a positive result.

Note that the resultant path(s) may be made up of a set of strict or loose hops, or any combination of strict and loose hops. Moreover, a hop may have the form of a non-explicit abstract node.

A request/response protocol is also required for a PCE to communicate

path computation requests to another PCE and for the PCE to return the path computation response. The path computation request may include a significant set of requirements including those defined above. In case of a positive response from the PCE, one or more paths would be returned to the requesting PCE. In the event of a failure to compute the desired path(s), an error is returned together with as much information as possible about the reasons for the failure, and potentially advice about which constraints might be relaxed to be more likely to achieve a positive result. Note that the resultant path(s) may be made up of a set of strict or loose hops, or any combination of strict and loose

hops. Moreover, a hop may have the form of a non-explicit abstract node.

No assumption is made at this stage about whether the PCC-PCE and PCE-PCE communication protocols are identical.

7.7. PCE TED Synchronization

As previously described, the PCE operates on a TED. Information on network status to build the TED may be provided in the domain by various means:

1) Participation in IGP distribution of TE information. The standard method of distribution of TE information within an IGP area is using extensions to the IGP. This mechanism allows participating nodes to build a TED, and this is the standard technique, for example, within a single area MPLS network. A node that hosts the PCE function may collect TE information in this way by maintaining at least one routing adjacency with a router in the domain. The PCE node may be adjacent or non-adjacent (via some tunneling techniques) to the router. Such a technique provides a mechanism for ensuring that the TED is efficiently synchronized with the network state and is the normal case, for example, when the PCE is co-resident with the LSRs in an MPLS network.

2) Out-of-band TED synchronization. It may not be convenient or possible for a PCE node to participate in the IGPs of one or more domains (for example, when there are very many domains, when IGP participation is not desired, or when some domains are not running TE-aware IGPs). In this case some mechanism may need to be defined to allow the PCE node to retrieve the TED from each domain. Such a mechanism could be incremental (like the IGP in the previous case), or could involve a bulk transfer of the complete TED. The latter might significantly limit the capability to ensure TED synchronization which might result in an increase in the failure rate of computed paths.

Consideration should also be given to the impact of the TED distribution on the network and on the network node within the domain that is asked to distribute the database. This is particularly relevant in the case of frequent network state changes.

3) Information in the TED can include LSP information obtained from sources other than the IGP. For example, this information can include LSP routes, reserved bandwidth, and measured traffic volume passing through the LSP. Such LSP information is required to perform LSP re-optimization, as described in Sections 4.4 and 7, which can be take into account the traffic fluctuations. Also, such LSP information is needed to reconfigure virtual network topology (VNT), in which lower layer LSPs such as optical paths form the VNT. The VNT is used for routing of higher-region traffic such as IP traffic.

Note that synchronization techniques apply to both intra- and inter-domain TEDs. Further, the techniques can be mixed for use with

different domains. The degree of synchronization between the PCE and the network is subject to implementation and/or policy. However, better synchronization leads to paths that are more likely to succeed.

It must also be highlighted that the PCE may have access to only a partial TED: for instance in the case of inter-domain path computation where each such domain may be managed by different entities. In such cases, each PCE may have a access to a partial TED and cooperative techniques between PCEs may be used to achieve end-to-end path computation without any requirement for any PCE to handle the complete TED related to the set of traversed domains by the LSP path in question.

7.8. Stateful Versus Stateless PCEs

A PCE can be either stateful or stateless. In the former case, there is a strict synchronization between the PCE and not only the network states (in term of topology and resource information), but also the set of computed paths and reserved resources in use in the network. In other words, the PCE utilizes information from the TED as well as information about existing paths (for example, TE LSPs) in the network when processing new requests. Note that although this allows for optimal path computation and increased path computation success, stateful PCEs require reliable state synchronization mechanisms, with potentially significant control plane overhead and the maintenance of a large amount of data/states (for example, full mesh of TE LSPs).

For example, if there is only one PCE in the domain, all LSP computation is done by this PCE, which can then track all the existing LSPs and stay synchronized. However, this could require substantial control plane resources to accomplish. If there are multiple PCEs in the network, LSP computation and information is distributed among PCEs and the resources required is also distributed. However, synchronization issues discussed in Section 6.7 also come into play.

The maintenance of a stateful database can be non-trivial. However, in a single centralized PCE environment, a stateful PCE is almost a simple matter of remembering all of the LSPs the PCE has computed, if it can also be known that the LSPs were actually set up, and when they were torn down. Out-of-band TED synchronization can also be complex with multiple PCE setup in a distributed PCE computation model, and could be prone to race conditions, scalability concerns, etc. Even if the PCE has detailed information on all paths, priorities, and layers, taking such information into account for path computation could be highly complex. PCEs might synchronize state by communicating with each other, but when LSPs are set up using distributed computation performed among several PCEs, the problem of synchronization becomes larger and more complex.

There is benefit in knowing which LSPs exist, and their routing, to support such applications as placing a high priority LSP in a crowded network such that it preempts as few other LSPs as possible. Note that

preempting based on the minimum number of links might not result in the smallest number of LSPs being disrupted. Another application concerns the construction and maintenance of a Virtual Network Topology [MRN]. It is also helpful to understand which other LSPs exist in the network in order to decide how to manage the forward adjacencies that exist or need to be set up. The cost-benefit of stateful PCE computation would be helpful to determine if the benefit in path computation is sufficient to offset the additional drain on the network computational resources.

Conversely, stateless PCEs do not have to remember any computed path and each set of request(s) is processed independently of each other. For example, stateless PCEs may compute paths based on current TED information, which could be out of sync with actual network state given other recent PCE-computed paths changes. Note that a PCC may include a set of previously computed paths in its request, in order to take them into account, for instance to avoid double bandwidth accounting, or to try to minimize changes (minimum perturbation problem).

7.9. Monitoring

PCE Monitoring is undoubtedly of the utmost importance in any PCE architecture. This must include the collection of variables related to the PCE status and operation. For example, it will be necessary to understand the way in which the TED is being kept synchronized, the rate of arrival of new requests and the computation times, the range of PCCs that are using the PCE, and the operation of any PCC-PCE protocol.

7.10. Policy and Confidentiality

As stated in [INTER-AS], the case of inter-provider TE LSP path computation requires the ability to compute a path while preserving confidentiality across multiple Service Providers cores. Thus any PCE architecture solution must support the ability to return partial paths by means of loose hops (for example, where each loose hops would for instance identify a boundary LSR). Confidentiality and security of PCC-PCE and PCE-PCE messages must also be ensured.

As mentioned in section 6.9, the ability to compute a path at the request of the head end PCC, but to supply the path in segments to the domain boundary PCCs may also be desirable.

8. PCE Evaluation Metrics

PCE evaluation metrics that may be used to evaluate the efficiency and applicability of any PCE-based solution are listed below.

- Optimality: The ability to maximize network utilization and minimize cost, considering QoS objectives, multiple regions and network layers.

- Scalability: The implications of routing and signaling overhead (includes LSAs, crankbacks, queries, distribution mechanisms, etc.).

- Load sharing: The ability to allow multiple PCEs to spread the path computation load.

- Multi-path computation: The ability to compute multiple and potentially diverse paths to satisfy load-sharing of traffic and protection/restoration needs including end-to-end diversity and protection within individual domains.

- Reoptimization: The ability to perform TE LSP path reoptimization. This also includes the ability to perform inter-layer correlation when considering the reoptimization at any specific layer.

- Path computation time. The time to compute individual paths, multiple diverse paths, and to satisfy bulk path computation requests.

- Network stability: The ability to minimize any perturbation on existing TE state resulting from the computation and establishment of new TE paths.

- Ability to maintain accurate synchronization between TED and network topology and resource states.

- Speed with which TED synchronization is achieved.

- Impact of the synchronization process on the data flows in the network.

Note that other metrics may also be considered. Such metrics should be used when evaluating a particular PCE-based architecture. It must also be highlighted that the potential tradeoffs of the optimization of such metrics should be evaluated (for instance, increasing the path optimality is likely to have consequences on the computation time).

9. Security Considerations

The impact of the use of a PCE-based architecture MUST be considered in the light of the impact that it has on the security of the existing routing and signaling protocols and techniques in use within the network. There is unlikely to be any impact on intra-domain security, but an increase in inter-domain information flows and the facilitation of inter-domain path establishment may increase the vulnerability to security attacks.

Of particular relevance are the implications for confidentiality inherent in a PCE-based architecture for multi-domain networks. It is not necessarily the case that a multi-domain PCE solution will compromise security, but solutions MUST examine their effects in this area.

Applicability statements for particular combinations of signaling,

routing and path computation techniques are expected to contain detailed

security sections.

It should be observed that the use of a non-local PCE (that is, not co-resident with the PCC) does introduce additional security issues. Most notable amongst these are:

- Interception of PCE requests or responses

- Impersonation of PCE

- Falsification of TE information

- Denial of service attacks on PCE or PCE communication mechanisms.

It is expected that PCE solutions will address these issues in detail using authentication and security techniques.

10. IANA Considerations

This document makes no requests for IANA action.

11. Acknowledgements

The authors would like to extend their warmest thanks to (in alphabetical order) Zafar Ali, Dean Cheng, Kireeti Kompella, Jean-Louis Le Roux, Eiji Oki, Dimitri Papadimitriou, and Raymond Zhang for their Review and suggestions.

12. Intellectual Property Considerations

ietf-ipr@ietf.org.

13. Normative References

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

[RFC3667] Bradner, S., "IETF Rights in Contributions", BCP 78, RFC 3667, February 2004.

[RFC3668] Bradner, S., "Intellectual Property Rights in IETF Technology", BCP 79, RFC 3668, February 2004.

14. Informational References

[RFC2702] Awduche, D., Malcolm, J., Agogbua, J., O'Dell and J. McManus, "Requirements for Traffic Engineering over MPLS", RFC 2702, September 1999.

[RFC2547] Rosen, E. and Rekhter, Y. "BGP/MPLS VPNs", RFC2547, March 1999.

[RFC3209] Awduche, D., et. all, "Extensions to RSVP for LSP Tunnels", RFC 3209, December 2001.

[RFC3473] Berger, L., et. al., "Generalized Multi-Protocol Label Switching (GMPLS) Signaling - Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Extensions", RFC 3473, January 2003.

[INTER-AREA] Le Roux, J., Vasseur, JP, Boyle, J., "Requirements for Support of Inter-Area and Inter-AS MPLS Traffic Engineering", draft-ietf-tewg- interarea-mpls-te-req-03.txt, November 2004 (work in progress).

[INTER-AS] Zhang, R., Vasseur, JP., et. al., "MPLS Inter-AS Traffic Engineering requirements", draft-ietf-tewg-interas-mpls-te-req-09.txt, September 2004 (work in progress).

[MRN] Papadimitriou, D., et. al., "Generalized MPLS Architecture for Multi-Region Networks,"draft-vigoureux-shiomoto-ccamp-gmpls-mrn-04.txt, February 2004 (work in progress).

15. Authors' Addresses

Adrian Farrel
Old Dog Consulting
Phone: +44 (0) 1978 860944
Fax: +44 (0) 870-130-5411
EMail: adrian@olddog.co.uk

Jean-Philippe Vasseur
Cisco Systems, Inc.
300 Beaver Brook Road
Boxborough , MA - 01719
USA
Email: jpv@cisco.com

Jerry Ash
AT&T
Room MT D5-2A01
200 Laurel Avenue
Middletown, NJ 07748, USA
Phone: +1-(732)-420-4578
Fax: +1-(732)-368-8659
Email: gash@att.com

16. Full Copyright Statement