Standards-Based Path Computation Element (PCE) Architectures

Wes Doonan
Control Plane R&D
July 2009
Three Primary Components of a Control Plane

- **Discovery**
  - Topology – how is transport network constructed?
  - Resources – what transport capabilities are available?

- **Path Computation**
  - Where can a given optical connection go?
  - Which resources are consumed by that optical connection?

- **Signaling**
  - How do I reserve the necessary resources?
  - How do I provision the necessary equipment?
Path Computation

- Where can a given optical connection go?
  - Describe the network as an abstract graph datastructure
  - Perform graph-theoretic computations on that datastructure
  - Generate an Explicit Route description
    - Connects endpoints, satisfies constraints

<table>
<thead>
<tr>
<th>IP Address</th>
<th>AS Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>49.1.1.1</td>
<td>1533.47</td>
</tr>
<tr>
<td>49.1.1.4</td>
<td>1533.47</td>
</tr>
<tr>
<td>11.2.2.7</td>
<td>1533.47</td>
</tr>
<tr>
<td>11.2.2.44</td>
<td>1549.32</td>
</tr>
<tr>
<td>67.77.1.14</td>
<td>1549.32</td>
</tr>
<tr>
<td>10.3.9.6</td>
<td>1549.32</td>
</tr>
<tr>
<td>10.3.9.1</td>
<td>1536.61</td>
</tr>
<tr>
<td>104.1.15.6</td>
<td>1536.61</td>
</tr>
<tr>
<td>104.1.15.7</td>
<td>1536.61</td>
</tr>
</tbody>
</table>
Path Computation Models

- **Have: Manual**
  - Person, pencil, paper ...

- **Have: Centralized**
  - Offline server(s) know entire topology
    - Topology data entered via spreadsheet, NMS/EMS extraction, etc
  - Server(s) crank out paths
    - Handed to signaling-enabled network elements as Explicit Routes

- **Have: Fully Distributed**
  - Topology discovered via intelligent agents & protocols
  - Network elements must perform their own computations
    - End-to-end or segmented via loose hops

- **Want: Flexible**
  - Topology discoverable via intelligent agents & protocols
  - Computations can be delegated to specific network elements
    - Choice to distribute or centralize or hybridize
Current Architectures

Area 0

START

= NE

Area 0

Area 0

END
Operating Constraints

- **Underpowered Elements**
  - Computation mostly needed at the edge
  - Edge devices often “computationally challenged”
    - Also “memory-challenged”, also “persistence-challenged”
  - Edge devices don’t care / shouldn’t know about core topology

- **Large Networks**
  - As network grows in size, so does the graph to process
  - To auto-discover topology via protocol, network load increases
  - Usual scaling concerns ...

- **Local Policies**
  - Policy enforcement possibly required upon every computation
    - With fully-distributed computation, policy enforcement is hard
  - Some network data may not be shareable outside local domain

Separate where computation is *needed* from where it is *done*
IETF PCE Architecture

- **Path Computation Client (PCC)**
  - Entity requesting path computation services
  - Input: <request type, ingress/egress, constraints ...>
  - Output: <explicit path description>
  - With standardized architecture, any element can be a PCC
    - Network elements
    - Network management systems
    - Diagnostic tools
    - Other PCEs (see below)

- **Path Computation Element (PCE)**
  - Entity performing path computations on behalf of clients
  - Satisfies requests from PCCs
    - Accesses network topology + capabilities data (e.g. via TEDB)
    - Performs graph-theoretic computations using that data
  - If full topology not locally known, can act as PCC to other PCEs
    - Calls other PCEs to generate full end-to-end path
PCE Architecture

Sees self

Sees nodes \(<A, B, C, D>\)

Sees nodes \(<E, F, G, H>\)

“compute A -> H”

“A->B->C->D -> E->F->G->H”

“compute E -> H”

“E->F->G->H”

= PCC  = PCE

= PCEP
PCE Protocol (PCEP)

- Standardized Protocol (RFC5440)
  - Request / Response Protocol
    - PCC issues requests to PCE, PCE replies to PCC
  - TCP transport
    - port 4189
  - Type/Length/Value (TLV) message format
  - Reuses several protocol objects from RSVP-TE
    - Session Attribute Object
    - Explicit Route Object + subobjects
    - Recorded Route Object + subobjects
  - Session context maintained through KeepAlive processing
  - Framework allows for Authentication, Policy extensions
    - Basic MD5 authentication, can operate over TLS if needed

- References
  - RFC4655 – PCE Architecture
  - RFC4657 – PCEP Generic Requirements
Feeding the PCE

- PCE must have view to some or all of transport topology

  **Source: OSPF-TE**
  - PCE participates in TE-enabled IGP
  - Builds local TEDB from received TE-Router/TE-Link LSAs
  - Most common method for feeding the beast

  **Source: Manual Configuration**
  - Element provides method for inserting nodes/links into TEDB
  - Accomodates legacy elements without OSPF-TE capability
  - Special circumstances, in practice quite rare

  **Source: PCEP**
  - Potential future direction for PCEP evolution
  - PCC may request insertion of node/link into PCE’s TEDB
    - Under policy/access control, clearly
  - Eases distribution of user-configured entries
Example: One PCE per Area

- PCC
- PCE
- PCEP
- LSP

START

END
One PCE per Area

- Fully Distributed Path Computation
  - Aligns with administrative regions/boundaries
    - Information scoping, administrative control
  - Relatively easy to setup, but complex to run
    - PCEs talking to other PCEs – many possible failure paths
    - How do PCEs know which PCEs handle which domains?

- PCEs only have visibility to their local domains
  - End-to-end path computed as sequence of segments
  - Segments may be optimal within their domain, but end-to-end is not
    - No opportunity for network-wide optimization
    - May not care, but then again may care a lot!

- Struggles mightily with certain problems
  - Parallel links, redundant paths (meshed regions)
Multiple Links

(Assumes all link metrics are equal)

START

BETTER PATH!

END

© 2009 ADVA Optical Networking. All rights reserved. ADVA confidential.
Hybrid Model

- Distributed + Centralized
  - Retain distributed autodiscovery via TE-enabled IGP
  - Retain separation between PCC and PCE functions
  - Select one or more elements as PCEs for entire network
    - Administrative choice, per deployment requirements

- How feed the PCEs?
  - Per standard, TE information flooded only at OSPF Area scope
    - Selected PCE element may not be a member of all Areas

- Tunnelling
  - Create GRE/IPIP tunnels between PCEs and elements in remote Areas
    - PCE tunnel interface configured with remote Area Id
    - Forms OSPF adjacency, TE information flooded to PCE

- PCEs have full view of topology, can do network-wide optimization
Characteristics

- Distributed discovery, centralized control
  - Ease of administration, error/fault processing

- Can globally optimize
  - Full view of topology, all paths/constraints known
  - Particularly relevant to optical networks
    - Optical nets full of hidden/subtle constraints
    - Piecemeal local computation sometimes insufficient
    - Optical paths are expensive, optimization a growing necessity

- Centralizes data
  - Good and bad; full view, but administrators must allow
    - TE data sharing only, however

- Single point of failure?
  - Well ...
Backup PCE

- PCC
- Primary PCE
- Backup PCE

Area 0

Area 1

Area 2

START

END

= GRE Tunnel
Redundant, Centralized approach implies failover
  PCC implementations become more intelligent
  Become aware of a set of possible PCEs
  Implement timeout/retry/failover semantics
  Needed anyway, really ...

PCE Autodiscovery a useful adjunct feature
  Eases PCC administration
  OSPF RI extension, LDAP, special-purpose protocol, etc

TE vs IGP LSA scoping
  Standard adjacencies over standard tunnels carry all LSA types
  Must get TE LSAs to feed PCEs, but also get standard IGP LSAs
  IGP visibility unnecessary if reachability achieved some other way
  Filtered by the routing agent, or just leave ‘em alone
  IGP LSAs often small compared to TE LSAs for optical nets!
  Alternatively, do multi-instance OSPF
  One for TE, one for IGP; very ASON-like ...
Summary

- IETF PCE Architecture provides basic toolbox
  - Separates PCC from PCE, defines standard protocol

- Considerable technology reuse, relatively easy implementation
  - Standard TLVs, single standard transport, etc

- Toolbox allows many deployment options
  - Choose method best suited to application and environment

- Standards are Done, RFCs issued
  - RFC4655 – PCE Architecture
  - RFC4657 – PCE Protocol Requirements
  - RFC4927 – PCE Protocol Requirements for Multi-Area
  - ... several others ...

- Available in shipping products from multiple vendors