ALTO Extension: Path Vector

draft-ietf-alto-path-vector-05

Presenter: Y. Richard Yang

IETF Interim
December 11, 2018

Overview

- The document does not make enough progress.

- Meeting goal: Revisit basic assumption to reach agreement and finish ASAP
  - Target: finish a complete pass in 2 weeks, before the end of this year

Recall: Current Design

- Path vector response structure consists of two maps
  - to remove redundancy; aka database normalized design should consist of two tables

```
"cost-map":{
   "PID1": { "PID2": ["ane:L001", "ane:L002"],
     ...}
}
```

```
"prop-map": {
   "ane:L001": {"delay": "10"},
   "ane:L002": {"delay": "30"} ......
}
```

- Strawman: Add "prop-map" in alto-costmap →
  - Problem: break existing alto-costmap media type
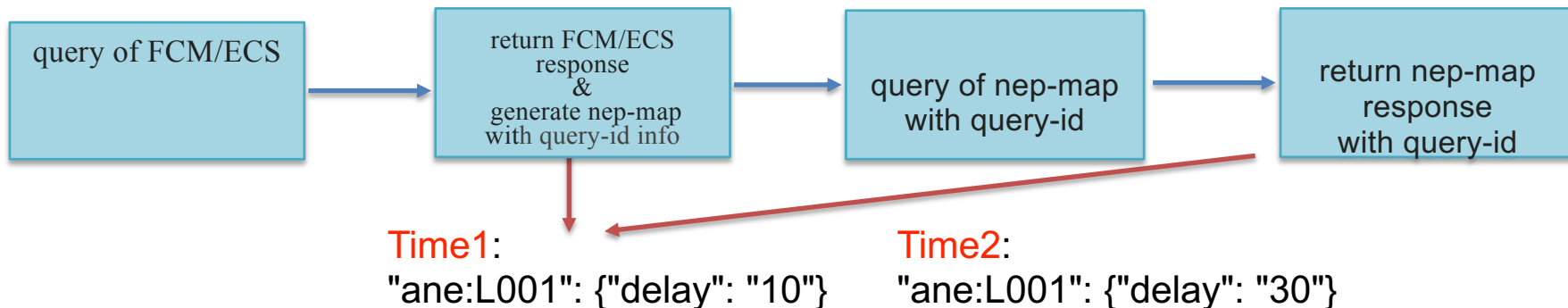
```
object {
   CostMapData cost-map;
   [PropertyMapData prop-map;]
} InfoResourceCostMap;
```

- Current design:
  - Send the two maps in two messages:
  - (1) alto-costmap message and →
  - (2) alto-propmap message for prop map

```
object {
   CostMapData cost-map;
} InfoResourceCostMap;
```

```
object {
   PropertyMapData property-map;
} InfoResourceProperties;
```

- Possible Problem: Snapshot consistency of two maps

| query of FCM/ECS | → | return FCM/ECS response & generate nep-map with query-id info | → | query of nep-map with query-id | → | return nep-map response with query-id |

Time1:
"ane:L001": {"delay": "10"}

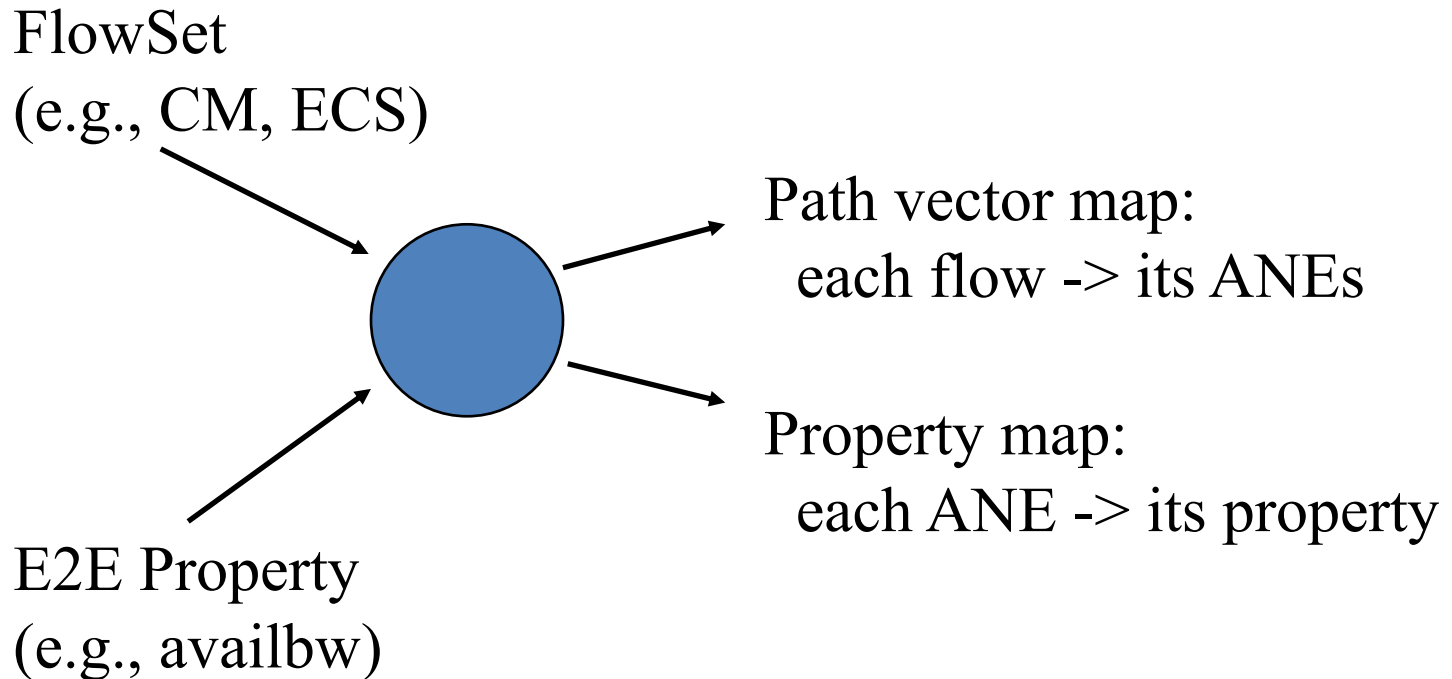Time2:
"ane:L001": {"delay": "30"}

Revisit First Principle

- Although the "single-node" network view abstraction works well in many settings, it lacks the ability to support emerging use cases, such as bandwidth or reliability scheduling of data transfer paths for multiple flows, which we refer to as co-flow scheduling; see Section 3 for a concrete example.

- Similar to BGP interdomain routing, which depends on exposing network paths to detect routing loops and support policy routing, these use cases require exposure of network paths to detect issues such as shared bottlenecks or shared risk link groups (SRLGs).

- A naive approach to exposing a network path is to reveal all network elements along the path and all properties of each element. This, however, can have both privacy and scalability issues. Similar to BGP interdomain routing, which exposes abstracted network paths (i.e., AS paths) called path vectors, the ALTO path-vector extension to be introduced in this document also uses abstracted network paths.
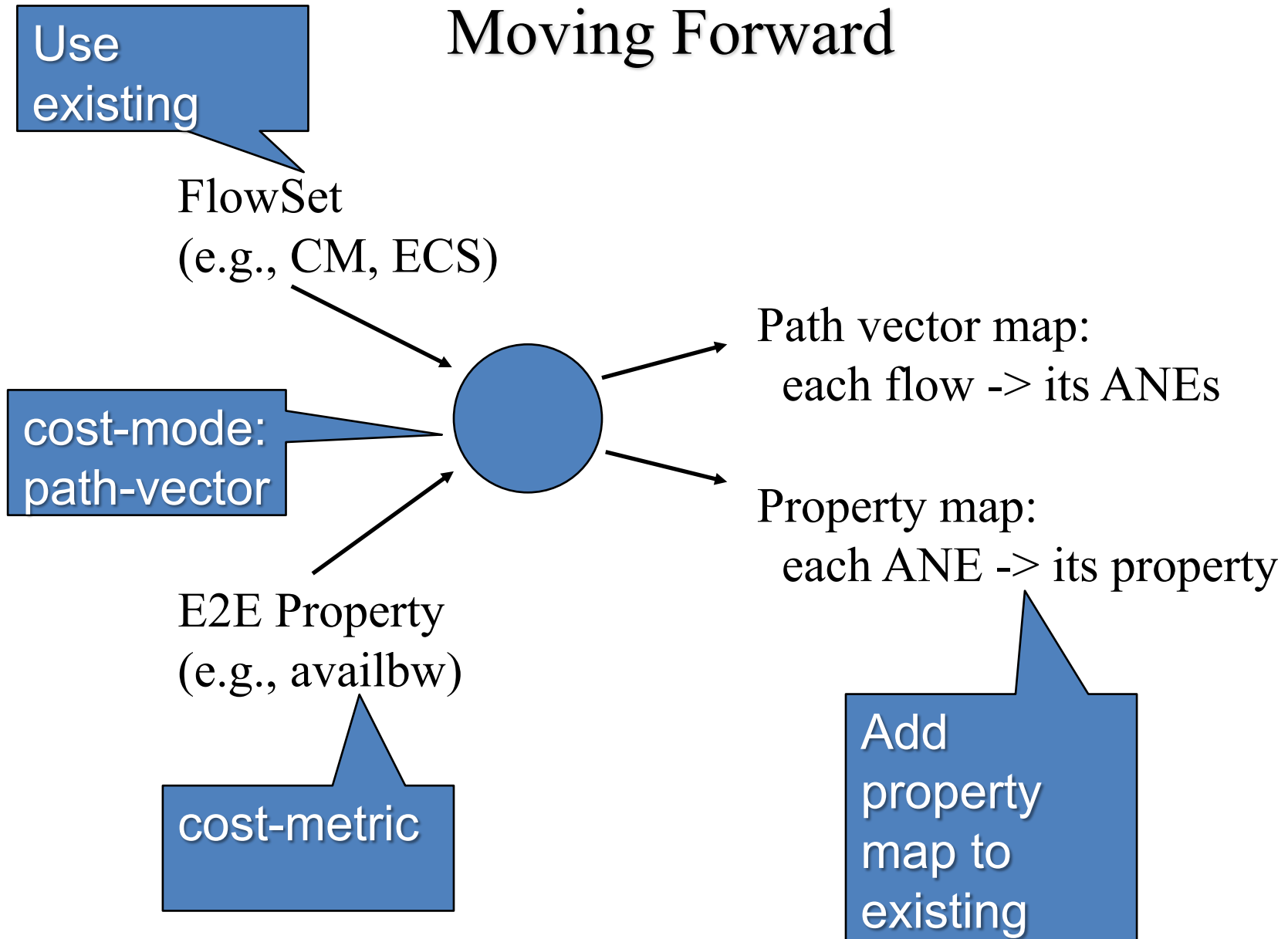
Revisit First Principle

- Specifically, the abstraction model used by the ALTO path-vector extension is based on the observation that given the set F of flows and the end-to-end property p to be exposed, abstracted network paths can be computed to both protect network privacy and reduce the amount of information to be exposed to network applications. We refer to the abstracted network paths computed by ALTO path vector extension as abstract path vectors, where an abstract path vector consists of a sequence of abstract network elements (ANEs); each ANE has a set of properties. By aggregating the path vectors consisting of ANEs and the properties of individual ANEs, network applications can compute required end-to-end property. Note that the ANEs and their properties computed depend on both the set of flows and the end-to-end property to be exposed.

Bigger Picture

FlowSet
(e.g., CM, ECS)

Path vector map:
   each flow -> its ANEs

Property map:
   each ANE -> its property

E2E Property
(e.g., availbw)

Take-away: A simple, coherent design. No separation of FlowSet, E2E Property, Path-vector map, Property map.

Moving Forward

Use existing

FlowSet
(e.g., CM, ECS)

cost-mode:
path-vector

E2E Property
(e.g., availbw)

cost-metric

Path vector map:
    each flow -> its ANEs

Property map:
    each ANE -> its property

Add property map to existing

Request

POST /endpointcostmap/pv HTTP/1.1

Host: alto.example.com

Accept: multipart/related, application/alto-costmap+json, application/alto-propmap+json, application/alto-error+json

Content-Length: [TBD]

Content-Type: application/alto-costmapfilter+json

```
{
    "cost-type": {  "cost-mode": "path-vector",
          "cost-metric": "availbw" },
   "endpoints": {
      "srcs": [ "ipv4:192.0.2.2" ],
      "dsts": [ "ipv4:192.0.2.89",
                "ipv4:203.0.113.45",
                "ipv6:2001:db8::10" ]
    }
}
```

Response

HTTP/1.1 200 OK
Content-Length: [TBD]
Content-Type: multipart/related; boundary=example-2

--example-2
Content-Type: application/alto-endpointcost+json

```
{
    "meta": {
       "cost-type": {...}
       "vtag": {
          …
       }
    },
    "endpoint-cost-map" : …
```

--example-2
Content-Type: application/alto-propmap+json

```
{
    "property-map" : { …}
}
```

--example-2--

Next Steps

- Quick update to move forward
  - First version: 1 week
  - Post to WG: 1 week

Q & A

Thanks