

ALTO Extensions to Support Application and Network Resource Information Exchange for High Bandwidth Applications for TE Networks

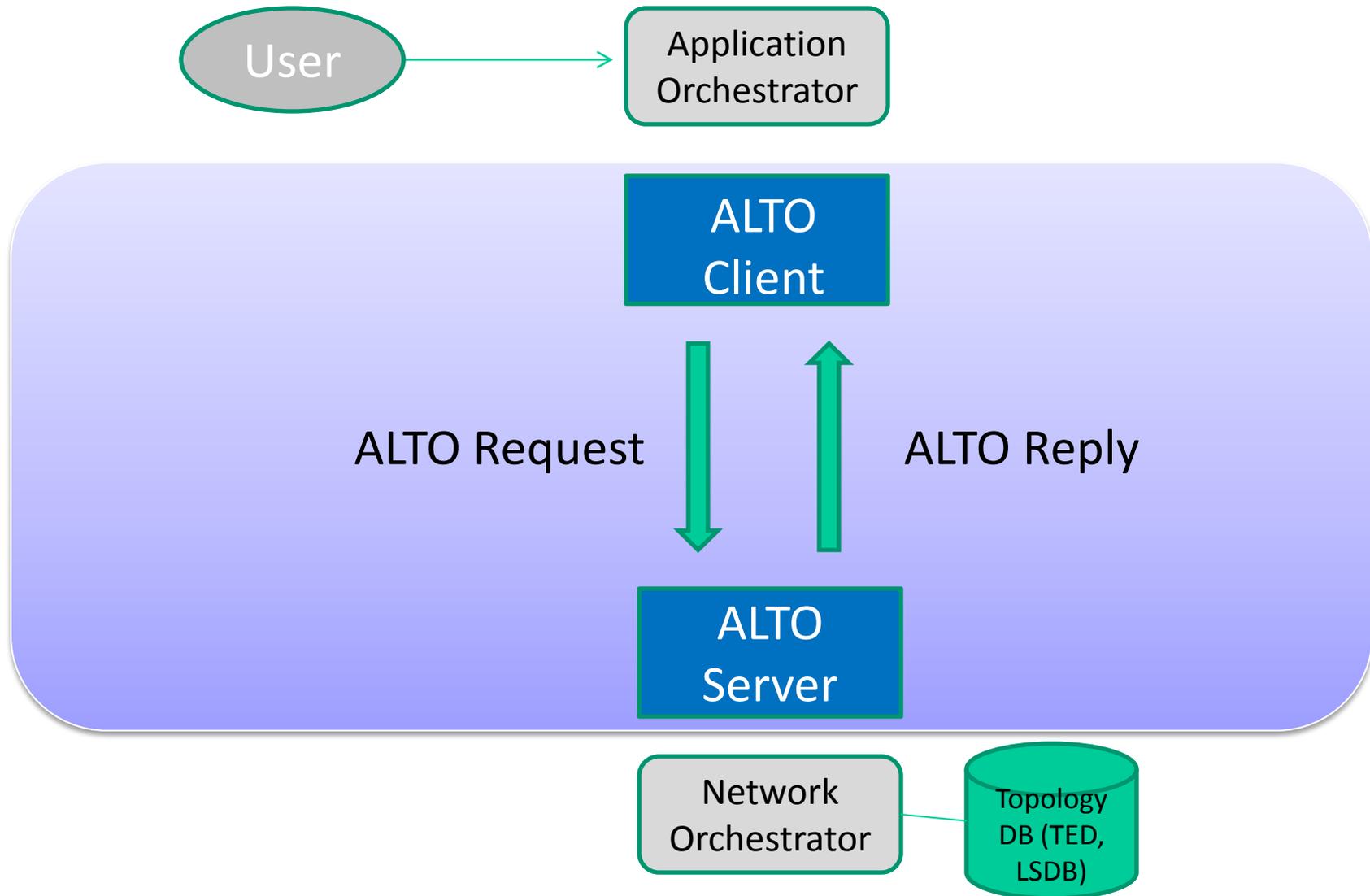
`<draft-lee-alto-app-net-info-exchange-04.txt>`

Young Lee, Dhruv Dhody, Qin Wu, Huawei
Greg Bernstein, Grotto Networking
Tae Sang Choi, ETRI

Scope

- This draft is to fulfill Large Bandwidth Use Case: <draft-bernstein-alto-large-bandwidth-cases-01>
- The context is TE networks
- Introducing new requirements to the current ALTO:
 - Graph Representation & Multiple Cost
 - Optimization Criteria
 - Constraint Filtering on paths or graphs (e.g., bandwidth, latency, hop count, packet loss, etc.)
 - Multiple Service Class

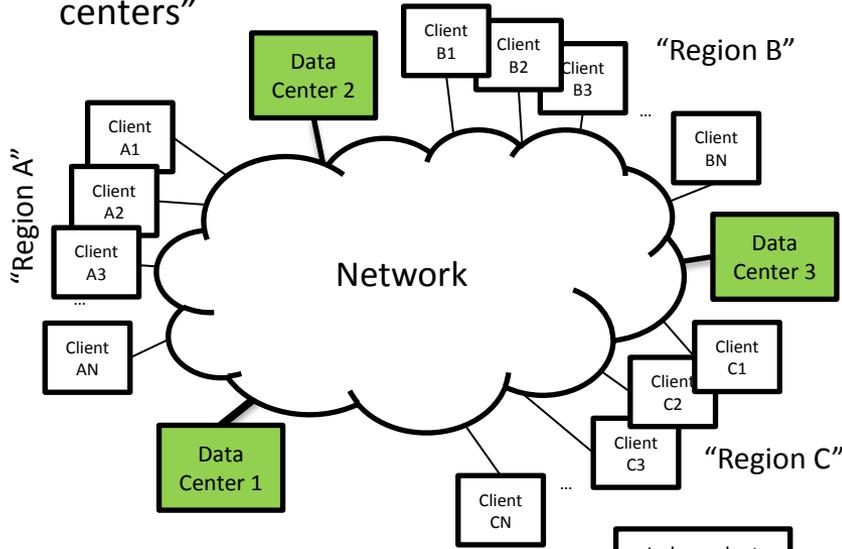
ALTO NET-APP context



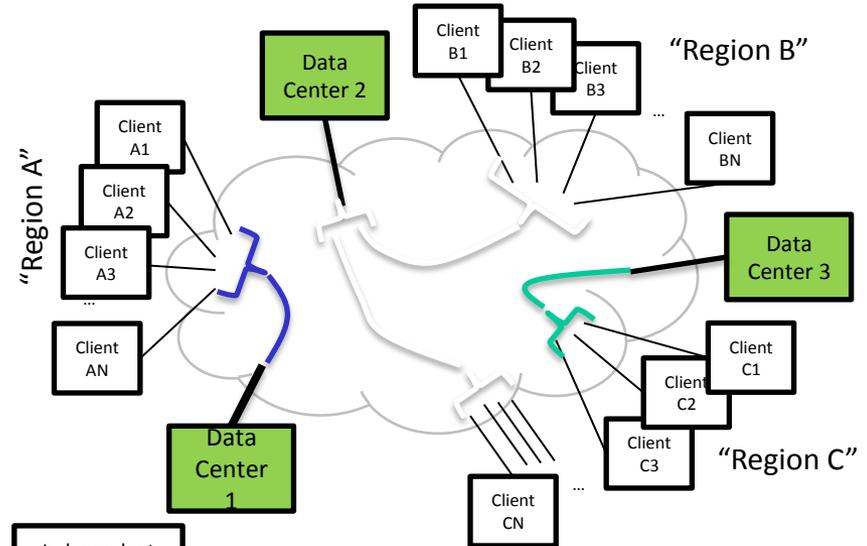
Large Bandwidth Use Case

<draft-bernstein-alto-large-bandwidth-cases-01>

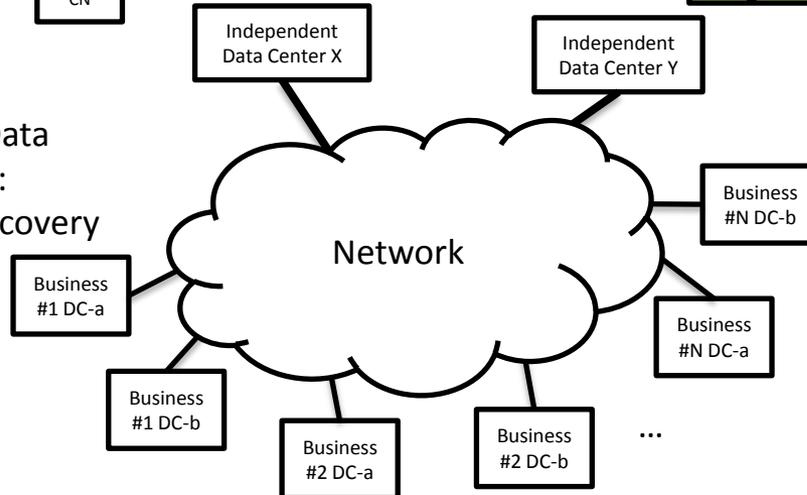
Case 1: End System Aggregation: Many clients using services offered at two or more "data centers"



Case 2: Traffic engineered "express lanes" between data centers and end user regions



Case 3: Data Center to Data Center Communications: Application Overlays, Recovery



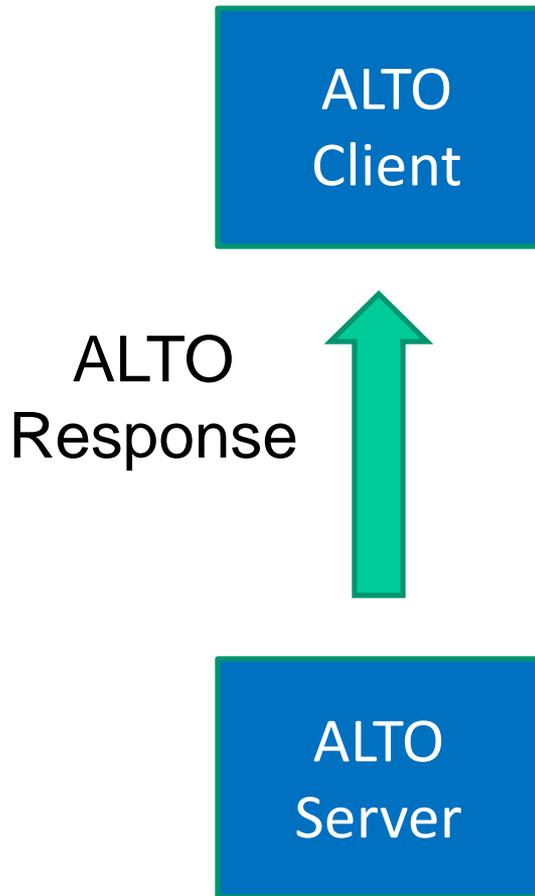
ALTO Query Mechanism that filters the request

constraints

- Cost Type:= 'routingcost' as defined by base specification. Additional cost (ex. latency, hopcount) are defined in [MultiCost] and [TE-cost].
- Cost Mode :={summary, graph} /* the cost map can be either a summary form or a graph form */
- Constraints /* a set of constraints that apply to the requested path summary or graph for filtering. (E.g., constraints can be like bandwidth greater than 'x', latency less than 'y', hopcount less than 'z', packetloss less than 'a' etc .) */
- Objective-function (or Optimization Criteria): The summary or the graph should be computed based on optimizing which parameter – IGP cost, latency, residual bandwidth, etc.
- Service-Class:= {gold, silver, bronze} /*the service class as described in this document*/



ALTO Response Mechanism with Reduction of Data Sets



- The list of feasible Source-Destination pair and its Cost Type : For each feasible S-D pair, indicate the following:
 - Service Class;
 - Cost Mode;
 - Cost Type;
 - Endpoint Cost Map Data
 - Parameter Values /* indicate the actual values of each constraint requested */
- Note that in case of Graph, each S-D pair is the source of the abstract link and the destination of the abstract link. Change from constraints to parameters

Info Model

Alto request:

```
Object{
  TypedEndpointAddr Src<1...*>; /*atleast one
    source*/
  TypedEndpointAddr Dsts<2...*>; /*atleast
    two destinations*/
  }EndpointList;
```

```
  Object{
    ServiceClass    service-class;
    CostMode        cost-mode;
    CostType        cost-type;
    [JSONString     constraints<0...*>; ]
    [JSONString     ObjectiveFunction]
    EndpointList    endpoints;
  }EndpointCostMapReq;
```

Alto response:

```
Object-map{
  JSONString    costparam;
} EndpointCostParam ;
```

```
Object-map{
  TypedEndpointAddr ->
    EndpointCostParam<1...*>;
} EndpointCosts ;
```

```
Object-map{
  TypedEndpointAddr -> EndpointCosts;
} EndpointCostMapData ;
```

```
Object{
  ServiceClass    service-class;
  CostMode        cost-mode;
  CostType        cost-type;
  [EndpointCostMapData    map;]
}EndpointCostMapRsp;
```

Encoding Example

Alto Request:

POST /endpointcost/lookup HTTP/1.1

Host: alto.example.com

Content-Length: [TODO]

Content-Type: application/alto-csoendpointcostparams+json

Accept: application/alto-csoendpointsummary+json,application/alto-error+json

```
{
  "service-class" : "silver",
  "cost-mode" : "summary",
  "cost-type" : "routingcost",
  "constraints": ["availbw gt 20", "delay lt 10", "pktloss lt 0.03", "jitter lt 10", "hopcount lt 5"
  ],
  "endpoints" : {
    "srcs": ["ipv4:192.0.2.2", "ipv4:192.0.2.10" ],
    "dsts": ["ipv4:192.0.2.89", "ipv4:198.51.100.34", "ipv4:203.0.113.45" ]
  }
}
```

Alto Response:

HTTP/1.1 200 OK

Content-Length: [TODO]

Content-Type: application/alto-csoendpointssummary+json

```
{
  "meta" : {},
  "data" : {
    "service-class" : "silver",
    "cost-mode" : "summary",
    "cost-type" : "routingcost",
    "map" : {
      "ipv4:192.0.2.2": {
        "ipv4:192.0.2.89" : [ "delay eq 5", "jitter eq 5", "pktloss eq 0.01", "hopcount eq 8", "cost eq 100" ],
        "ipv4:198.51.100.34" : [ "delay eq 9", "jitter eq 3", "pktloss eq 0.02", "hopcount eq 10", "cost eq 500" ],
        "ipv4:203.0.113.45" : [ "delay eq 4", "jitter eq 4", "pktloss eq 0.02", "hopcount eq 12", "cost eq 200" ]
      }
      "ipv4:192.0.2.10": {
        "ipv4:192.0.2.89" : [ "delay eq 4", "jitter eq 4", "pktloss eq 0.03", "hopcount eq 6", "cost eq 300" ],
        "ipv4:203.0.113.45" : [ "delay eq 6", "jitter eq 6", "pktloss eq 0.04", "hopcount eq 8", "cost eq 400" ]
      }
    }
  }
}
```

Summary & Next Steps

- This draft fulfills high-bandwidth TE network exchange using ALTO mechanism introducing:
 - Graph Representation & Multiple Cost
 - Optimization Criteria
 - Constraint Filtering on paths or graphs (e.g., bandwidth, latency, hop count, packet loss, etc.)
 - Multiple Service Class