Multi-Cost ALTO

Updates in

draft-randriamasy-alto-multi-cost-06

S. Randriamasy(ed.), N. Schwan

Diffs & use cases

- Main part of 05 - Multi-Cost (MC) Services
  - Extensions of ALTO protocol and ALTO services to include several cost types in 1 ALTO transaction
  - Introduces new Multi-Cost specific ALTO services

- New use cases
  - Endsystems needing to spare time by optimizing their ALTO transactions
- Discussions related to i2aex
  - CDN use case
  - Data center use case
  - Need information on more than topology
    - Resources on path and at endpoints

Objectives of Multi-Cost

- Gain time and resources by
  - Transport information on N Cost Types in 1 ALTO transaction rather than in N transactions
- 1 Multi-Cost Map instead of N Cost Maps
  - Less bulky to store than N Cost Maps
  - Represents a smaller data volume to transport
  - 1 MC transaction is faster than N single cost
  - Same for Filtered MC Map
- Suitable ALTO Services for multi-cost
  - Endpoint Multi-Cost service
  - Filtered multi-cost map

Specified Multi-Cost Services

- Multi-Cost Map Service
- Filtered Multi-Cost Map Service
- Endpoint Multi-Cost Service
  - Need to synchonize with current protocol updates
- Example of MC ALTO requests and responses
  - For each of the 3 services

Example request – Filtered MC Map

- •Suppose Cost Type « routingcost » = monetary cost.
- •Client wants to figure out delay, so it requests Type « hopcount »

```
POST multi/multicostmap/filtered HTTP/1.1
  Host: alto.example.com
  Content-Type: application/alto-nnncostmapfilter+json
  Accept: application/alto-nnncostmap+json,application/alto-error+json

  {
    "cost-mode" : "numerical", "numerical"],
    "cost-type" : "routingcost", "hopcount"],
    "pids" : {
      "srcs" : [ "PID1" ],
      "dsts" : [ "PID1", "PID2", "PID3" ]
    }
  }
```

Example response – Filtered MC Map

```
HTTP/1.1 200 OK
  Content-Length: [TODO]
  Content-Type: application/alto-NNNcostmap+json

  {
    "meta" : {},
    "data" : {
      "cost-mode" : ["numerical", "numerical"],
      "cost-type" : ["routingcost", "hopcount"],
      "map-vtag" : "1266506139",
      "map" : {
        "PID1": { "PID1": [1,6],  "PID2": [5,23],  "PID3": [10,5] }
      }
    }
  }
```

Thank you

back-up slides follow

ALTO Multi-Cost rules

- Term EP covers
  - Peer, CDN storage location, party in grid computing or on-line gaming or other resources sharing applications.
- Properties have constant values, costs can vary
- **Rule1**
  - when multiple cost types are requested then the requested Cost Mode MUST be numerical for those Costs Types encoded in JSONNumber
    - Reason: avoid mixing ordinal and numerical costs, requests too complex to handle and ordinal is easy to retrieve from numerical
    - Does not apply to Costs encode with JSONBool, JSONString
- **Rule2 – value order specification**
  - The ALTO response, MUST include an array of cost-types, arranged the same way as the values
  - The cost values for Source/Destination pairs MUST be provided in the same order as in the array of cost types

UC3: data transfer scheduling with « dynamic » costs

- CDNs need to regularly transfer their data for dissemination purposes
  - Need to avoid interfering with user peak activity
- Particular groups of users have limited access
  - to network and/or resources in time
- In both cases
  - Fixed/limited choice on target locations
  - Need for bandwidth
  - ➔Need to schedule their transfers
  - ➔Need information at various time periods on e.g.
    - ❖ Path occupation
    - ❖ Routing cost