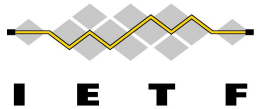
## Application Layer Traffic Optimization (ALTO) WG

# OpenALTO Implementation, Deployment Supporting LHCONE Use Cases

Presenter: Lauren Delwiche
on behalf of team (Jensen, Kai, …)

**IETF 115 Hackathon**
**06 November 2022, London**



# Context: OpenALTO, openalto.org

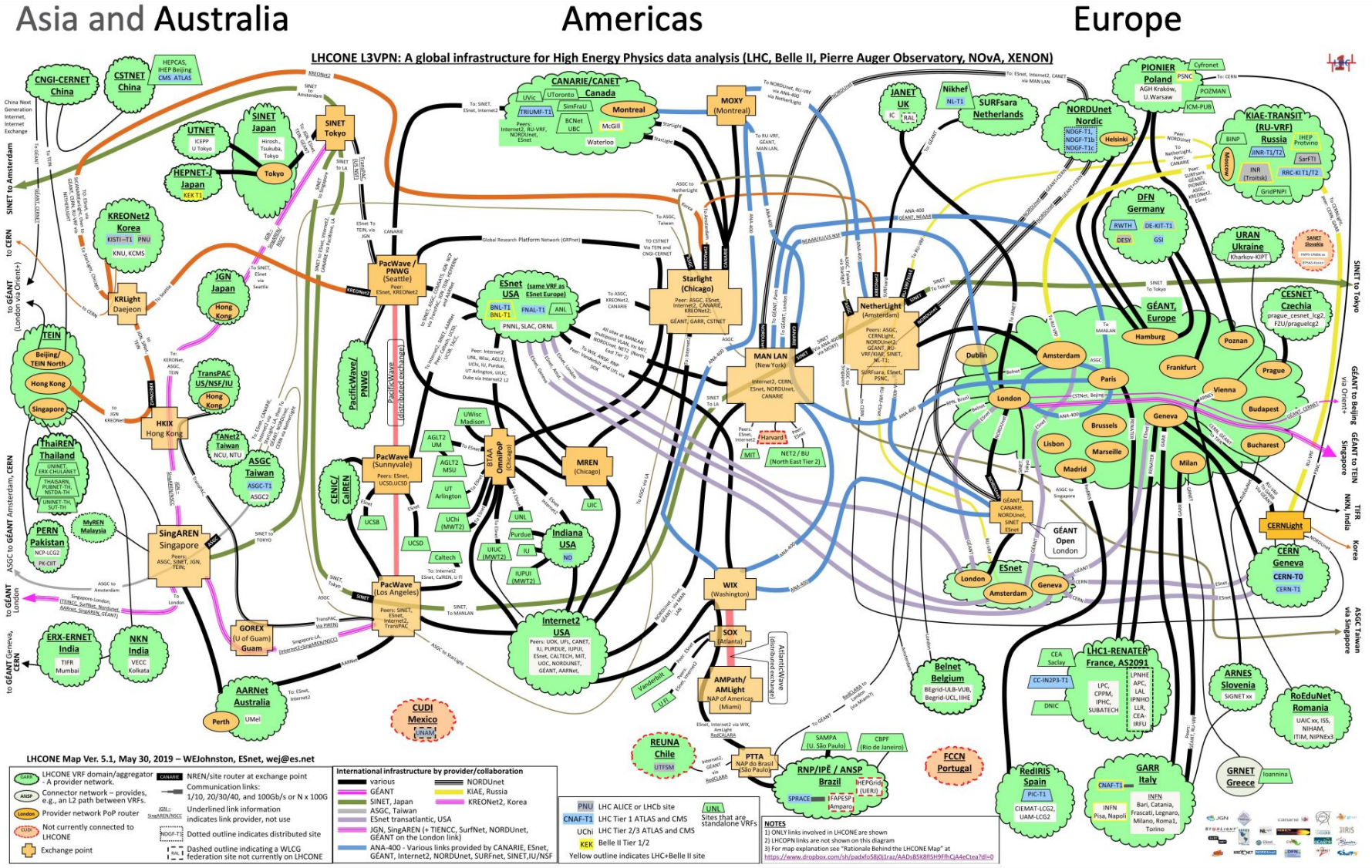


OpenALTO is an open-source **implementation** and platform of ALTO (MIT License).

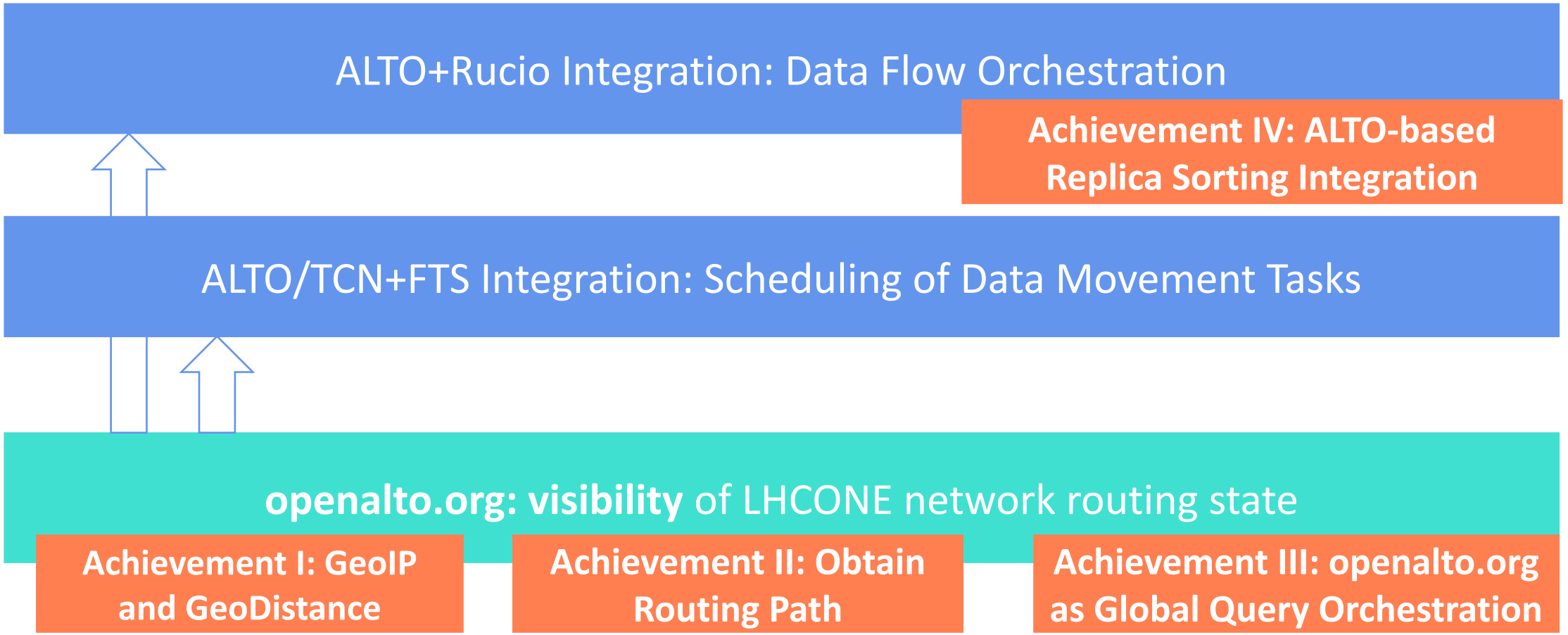
Available at *https://github.com/openalto/alto*

openalto.org is a running instance of deployment of OpenALTO, providing network information, in particular, in the context of data-intensive sciences, such as LHCONE.

# Context: LHCONE



# Context: LHCONE, openalto.org Use Cases



# Hackathon Achievement I: GeoIP and GeoDistance

- Providing geoip property using the standard ALTO endpoint property service **[RFC 9240]**
- Providing geo distance between endpoints using the standard ALTO Endpoint Cost Service (ECS) **[RFC 7285]**

```
➜curl -s -k -u cern:lhcone -H 'Content-Type: application/alto
-propmapparams+json' -d '{"entities":["ipv4:198.17.101.70"]}'
https://localhost:8443/entityprop/geoip | jq .
{
  "property-map": {
    "ipv4:198.17.101.70": {
      "geolocation": {
        "lat": 32.8515,
        "lng": -117.2798
      }
    }
  }
}
```

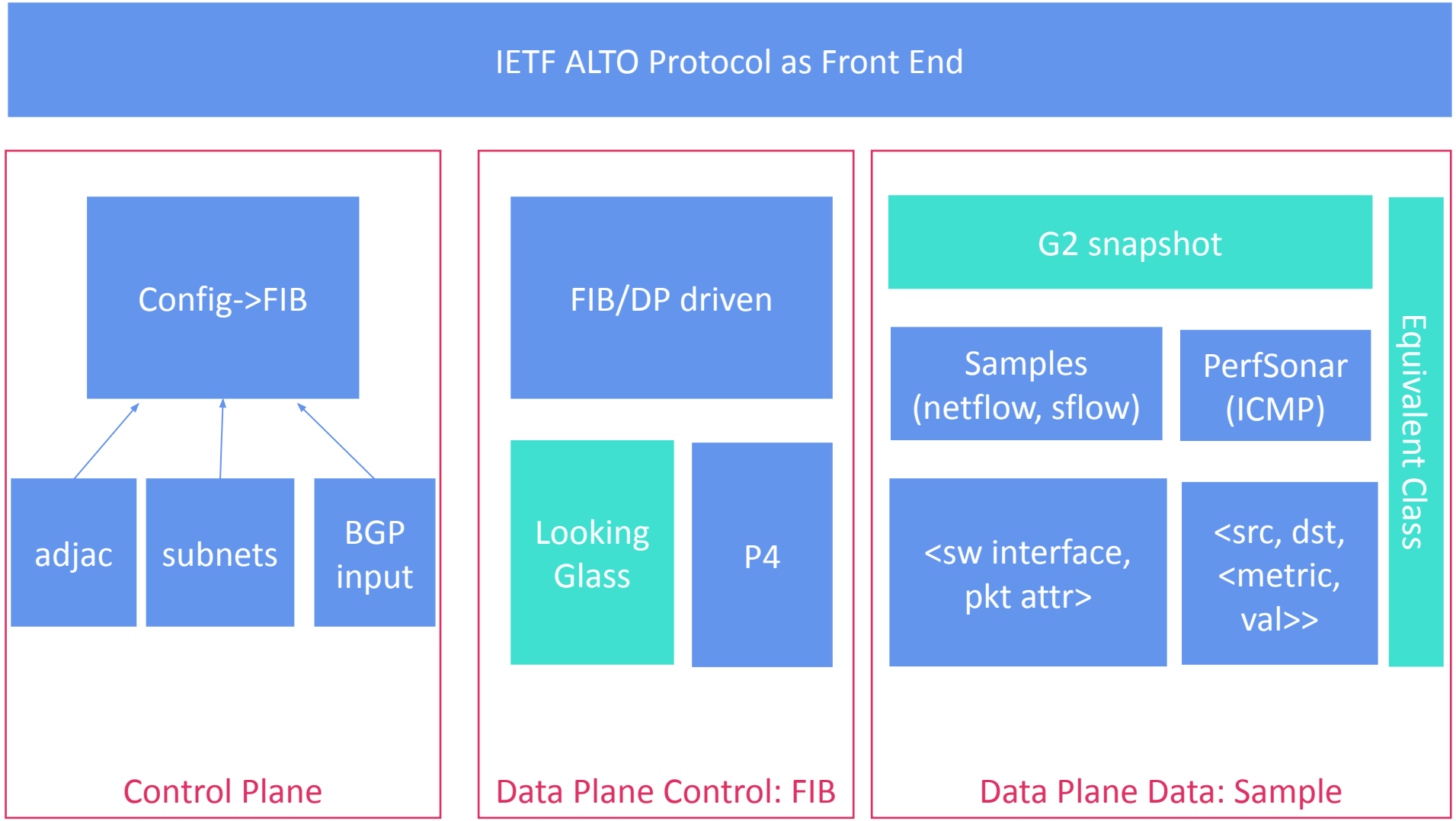
Query interface

etc > {} geoip-delegate-agent.json > ...

```
{
    "namespace": "default",
    "agent_class": "alto.agent.delegate.DelegateAgent",
    "data_source_name": "geoip",
    "data_source_config": {
        "data_source_cls": "alto.agent.geoip.GeoipAgent",
        "db_path": "/opt/geoip2/GeoLite2-City.mmdb"
    },
    "refresh_interval": 300
}
```

Implementation Config

# Hackathon Achievement II: Obtain Routing Paths



# FIB/LG Driven Implementation

Query/Response

Query Example (ECS with path vector extension)

```
→ cat request-cern.json
{
  "cost-type": {
    "cost-metric": "ane-path",
    "cost-mode": "array"
  },
  "endpoint-flows": [
     {
        "srcs": [ "ipv4:137.138.0.101" ],
        "dsts": [ "ipv4:134.158.84.23", "ipv4:144.16.112.112" ]
     },
     {
        "srcs": [ "ipv4:192.16.166.254" ],
        "dsts": [ "ipv4:140.115.32.101" ]
     }
  ],
  "ane-property-names": [ "next_hop", "as_path" ]
}
```

Response Example (ECS with path vector extension)

```
→curl -s -H 'Content-Type: application/alto-endpointcostparams+json' --data-ascii @
request-cern.json https://science.jensen-zhang.site/pathvector/cern-pv | ./pprint
--d41d8cd98f00b204e9800998ecf8427e
Content-Type: application/alto-endpointcost+json
Content-ID: <ecs@science.jensen-zhang.site>

{'endpoint-cost-map': {'137.138.0.101': {'134.158.84.23': ['autolink_1',
                                                           'autopath_2'],
                                         '144.16.112.112': ['autolink_1',
                                                            'autopath_3']},
                       '192.16.166.254': {'140.115.32.101': ['autolink_1',
                                                             'autopath_1']}},
 'meta': {'cost-type': {'cost-metric': 'ane-path', 'cost-mode': 'array'},
          'vtag': {'resource-id': 'cern-pv.ecs',
                   'tag': 'e615bf984f7249949f8903c5cf56f02d'}}}
--d41d8cd98f00b204e9800998ecf8427e
Content-Type: application/alto-propmap+json
Content-ID: <propmap@science.jensen-zhang.site>

{'meta': {'dependent-vtags': [{'resource-id': 'cern-pv.ecs',
                               'tag': 'e615bf984f7249949f8903c5cf56f02d'}]},
 'property-map': {'.ane:autolink_1': {'next_hop': '192.65.184.145'},
                  '.ane:autopath_1': {'as_path': '20965 24167 7539 1659'},
                  '.ane:autopath_2': {'as_path': '20965 2091 789'},
                  '.ane:autopath_3': {'as_path': '20965 9885 55824'}}}
--d41d8cd98f00b204e9800998ecf8427e--
```

Implementation

FIB Retrieval (LG; deployment at CERN and GEANT)

etc > {} lg-agent.json > ...

```
{
    "namespace": "default",
    "agent_class": "alto.agent.cernlg.LookingGlassAgent",
    "uri": "http://lhcone-lg.cern.ch/lg.cgi",
    "default_router": "ex2j.cern.ch:juniper",
    "refresh_interval": 300
}
```

# Data Path Sampling Driven Implementation

DP Sampling + Equivalent Class (Deployment at NRP)

etc > {} nrp-agent.json > ...

```
{
    "namespace": "default",
    "agent_class": "alto.agent.g2.G2Agent",
    "base_uri": "https://g2.nrp-nautilus.io/api/",
    "username": "admin",
    "password": "admin",
    "ec_rule": "/etc/ec-rule.json",
    "refresh_interval": 300
}
```

{} ec-rule.json > {} 3

```
[
    {
        "src_prefix": "128.114.109.70/24",
        "dst_prefix": "163.253.70.0/24"
    },
    {
        "src_prefix": "128.114.109.70/24",
        "dst_prefix": "163.253.71.0/24"
    },
    {
        "src_prefix": "128.114.109.70/24",
        "dst_prefix": "163.253.72.0/24",
        "dst_port": 80
    },
```

# Hackathon Achievement III: openalto.org as Global Query Orchestration Platform

- Good progress, not complete implementation by end of Hackathon
- High-level protocol query process: openalto.org orchestrates the multi-domain query process for LHCONE:
  - Lookup srcIP in Internet Routing Registry (IRR) to obtain source AS
  - Query ALTO server of source AS to obtain AS path
  - openalto.org refines the AS path to obtain general path representation
    - General path representation description given in WG email ( https://mailarchive.ietf.org/arch/msg/alto/2RMZgqSl2-wQ-eHKcnPyslPnzvs/)

# Hackathon Achievement IV: Integration

- Specify new ALTO resources to guide Rucio source selection
- Ongoing: specify general path to modify FTS scheduling

# Sorting Replicas with ALTO Sorting Expression

## Step 1: Configuration

Configure ALTO resources to fetch distances between replicas and the destination, and/or properties of replicas

alto.conf

```
#####################################################
# Configure an ALTO client
#####################################################
[client]
# ALTO server
default_ird = "https://science.jensen-zhang.site/pathvector/cern-pv"
metrics = {
  "as_hopcount": {
    "resource_type": "path-vector",
    "resource_id": "cern-pv",
    "prop_names": ["as_path"]},
  "geo_dist": {
    "resource_type": "endpoint-cost",
    "resource_id": "geo-distance"}}
auth_type = userpass
username = cern
password = lhcone
```

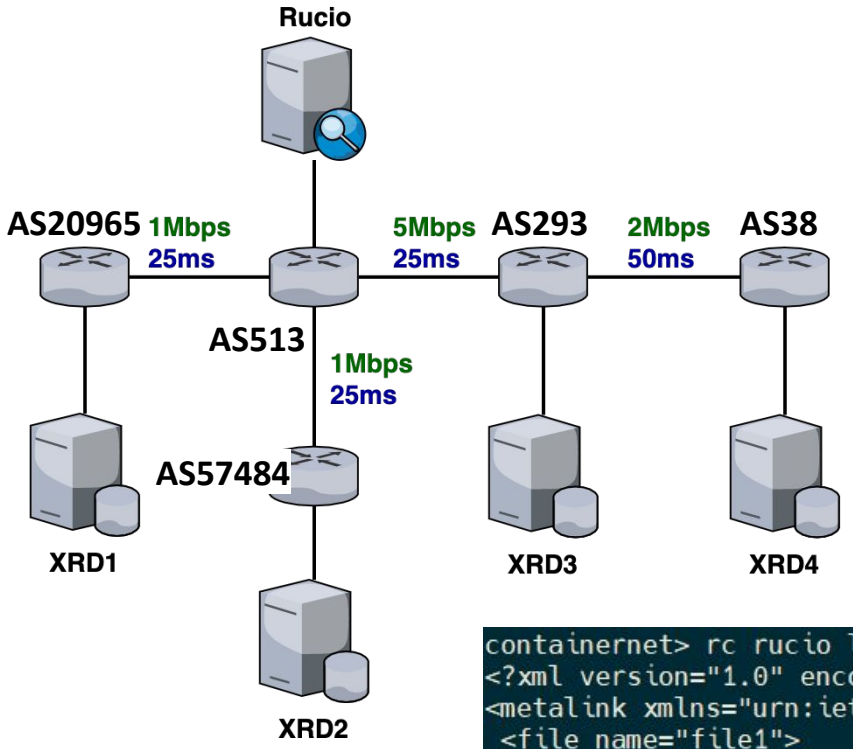
## Step 2: Express Sorting Demands

ALTO sorting expression enables Rucio download command to sort replicas based on a combination of distances and properties

BY=as_hopcount, geo_dist

WHERE continent=”EU”

# Hackathon Achievement IV: Integration




- Use mininet to partially simulate the LHCONE network.
- Host IPs are configured to the real public IPs in the real LHCONE network.

**Example of rucio replica sorting using ALTO**

```
containernet> rc rucio list-file-replicas --sort=alto,by=as-hopcount --metalink test:file1
<?xml version="1.0" encoding="UTF-8"?>
<metalink xmlns="urn:ietf:params:xml:ns:metalink">
 <file name="file1">
  <identity>test:file1</identity>
  <hash type="adler32">69fe2b13</hash>
  <hash type="md5">12969016e761864f30f97dd5fb259e30</hash>
  <size>1048576</size>
  <glfn name="/atlas/rucio/test:file1"></glfn>
  <url location="XRD1" domain="wan" priority="1" client_extract="false">root://xrd1:1094//rucio/test/80/25/file1</url>
  <url location="XRD3" domain="wan" priority="2" client_extract="false">root://xrd3:1096//rucio/test/80/25/file1</url>
  <url location="XRD4" domain="wan" priority="3" client_extract="false">root://xrd4:1097//rucio/test/80/25/file1</url>
 </file>
</metalink>
```

# Backup

# Using the Visibility Information: Rucio

- N sources of a data item
- M potential destinations of the data item
- Pick K out of M destinations to replicate

- Basic Rucio algorithm
  - Foreach s in N (sources), d in M
    - Compute the distance Distance_sd (current Rucio, Distance_sd == geo distance of s and d)
    - For the case of Distance is general path representation, we need an algebra for comparing two paths
  - Sort the array Distance_sd (N x M) distances
  - dstChose = {}
  - while ( |dstChose | < K )
    - Pick the next lowest D_sd, if the dest d has not already been chosen, dstChose += {d} // delete all Distance_sd where d appears