ESnet
ENERGY SCIENCES NETWORK

# Hecate Update

**Mariam Kiran, Scott Campbell, Nick Buraglio**
Energy Sciences Network
Lawrence Berkeley National Laboratory

Internet 2 TechEX
18–22 September, 2023

U.S. DEPARTMENT OF ENERGY
Office of Science

BERKELEY LAB

# 20 slides, 20 minutes

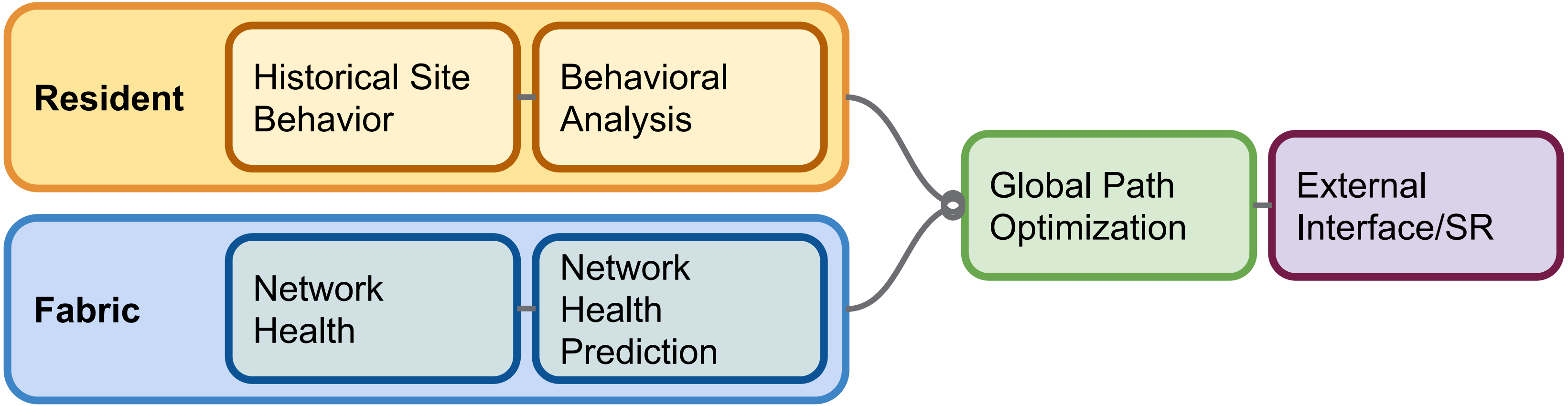
- Project Overview: What is Hecate about?
- Project Update: What have we done?
- Planned research agenda: How are we planning on reaching our objective?

I will focus on where we are in terms of providing a deliverable or more realistically *how we can best be in a position to deeply understand details around proposed vendor solutions*.

ESnet

# Project Overview

Objective: *Create efficient routing advice for traffic within ESnet based on a combination of historical site behaviors and current/projected network health.*

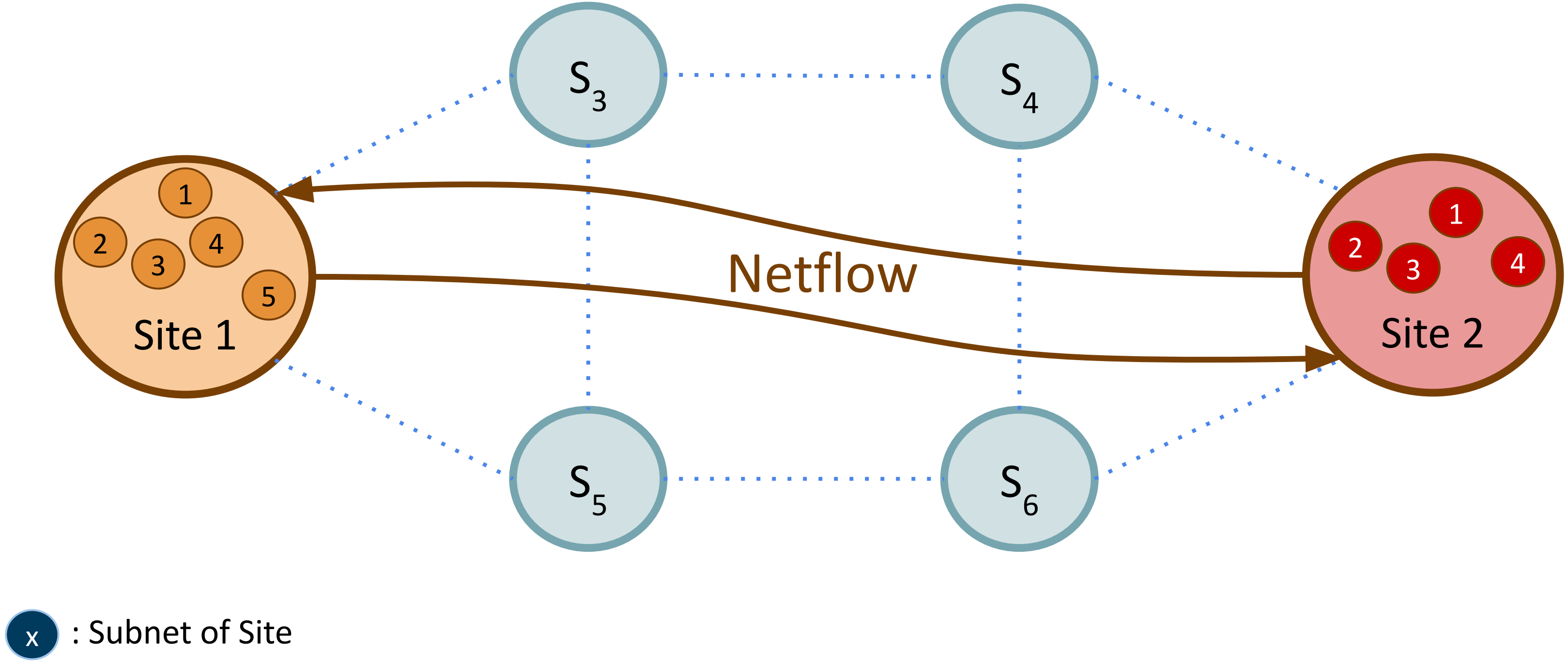



ESnet

# Data: What have we done?

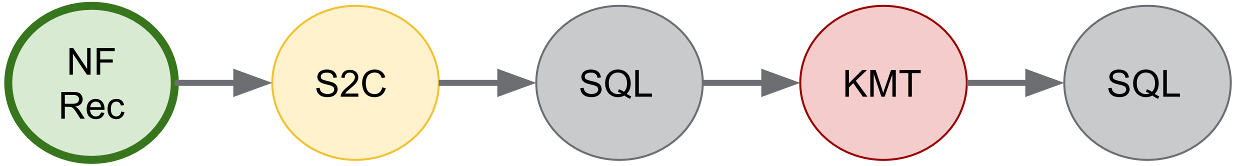

- Behavior of long term/permanent data plane residents: sites, facility, instrument
- Look backwards (time) for results
- Statistical not specific behaviors

# Data: Historical Network Activity



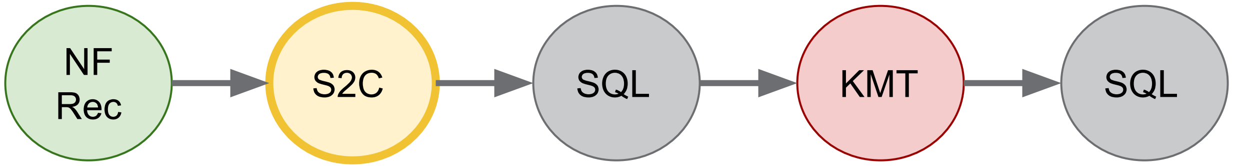

ESnet

# Data: Historical Site Activity




| | |
|---|---|
| Raw Flow Records | Raw flow records in Splunk |
| splunk_to_conn.py | Read time index flow records, process into connections w/ metadata -> SQLite |
| k-means-time.py | SQLite -> Read time window get clustered size and duration information -> SQLite |

# Data: Historical Site Activity




| | |
|---|---|
| Raw Flow Records | Raw flow records in Splunk |
| splunk_to_conn.py | Read time index flow records, process into connections w/ metadata -> SQLite |
| k-means-time.py | SQLite -> Read time window get clustered size and duration information -> SQLite |

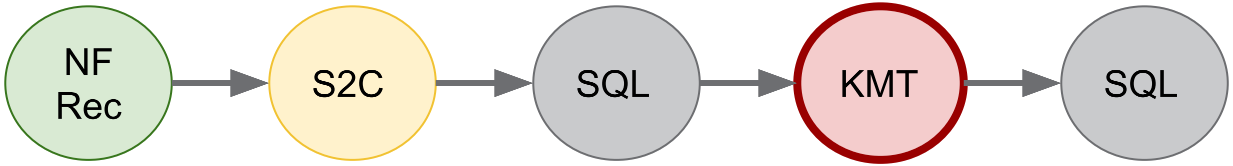
ESnet

# Connection Metadata: splunk_to_conn.py

| | |
|---|---|
| **C/S Site** | Ex: LBNL, ORNL |
| **cs_data_ratio_norm** | client/server ratio: data ratio imbalance |
| **cs_psize_ratio_norm** | client/server ratio: pkt size ratio imbalance |
| **C/S velocity** | data/duration |
| **C/S avg_size** | data/total packets |
| **C/S/T packets, bytes** | packet count |
| **duration** | |
| Data taken from 1/1000 sampled flows; conn = IP1:high_port -> IP2:low_port<br>Ratio Norm designed to separate A/B~1 vs. A/B >>1 OR A/B << 1 | |

ESnet

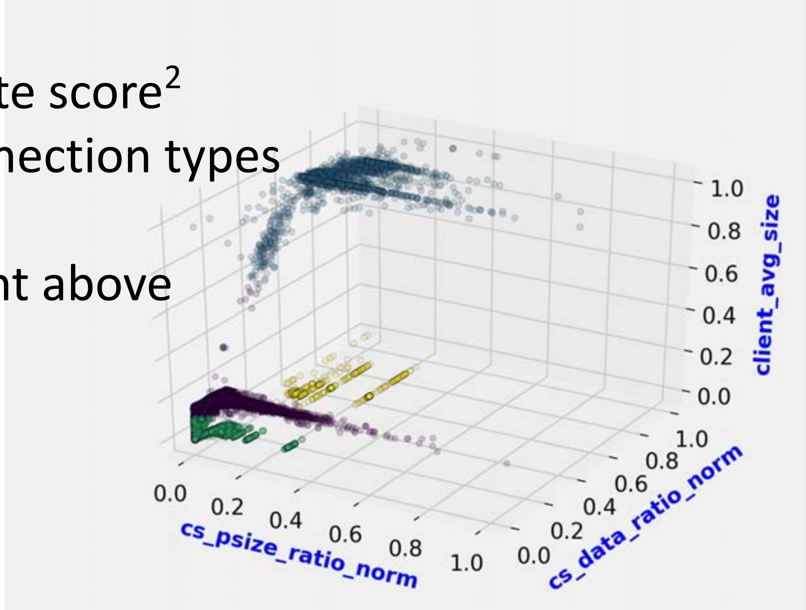
# Data: Historical Site Activity




| | |
|---|---|
| Raw Flow Records | Raw flow records in Splunk |
| splunk_to_conn.py | Read time index flow records, process into connections w/ metadata -> SQLite |
| k-means-time.py | SQLite -> Read time window get clustered size and duration information -> SQLite |

# Unsupervised Clustering

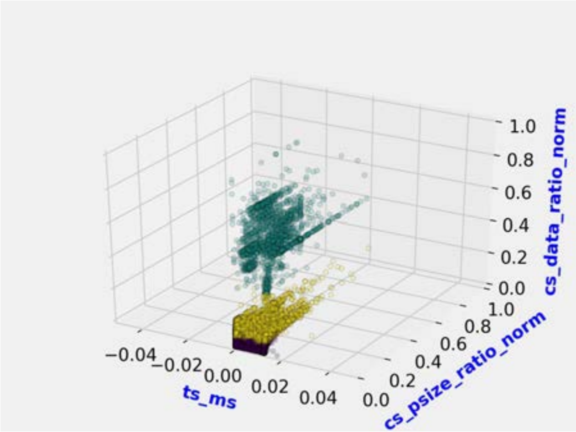
1. Embed connection in {Packet size ratio, Data size ratio, client avg size}[1]
2. Calculate number of clusters via Silhouette score[2]
3. ID cluster identities by taking known connection types and looking at each
4. Tag cluster members based on assignment above
5. Sanity check results

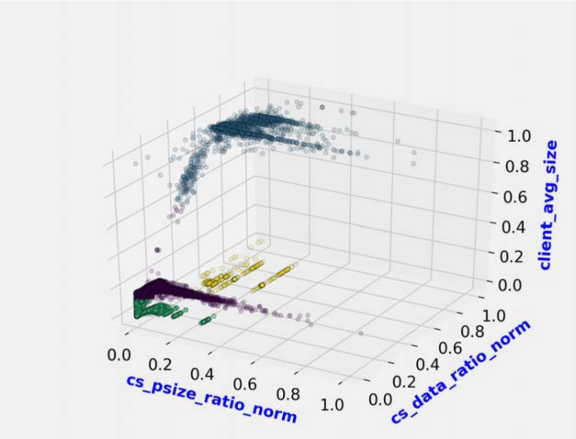



1: Test moving from mean to median to address outliers
2: if n>1000, sample set and calc SS from it

ESnet

# Data: Historical Site Activity



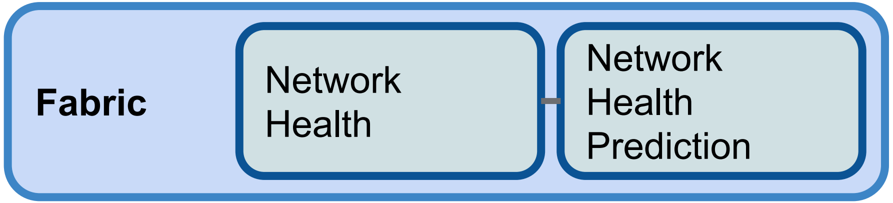




| Time Focused Tag | Data Volume Focused Tag |
|---|---|
| Short\|Med\|Long : **000111** [n] | Small\|Med\|Large: **111000** [n] |

Individual Site1[$x_1$] -> Site2[$x_2$]

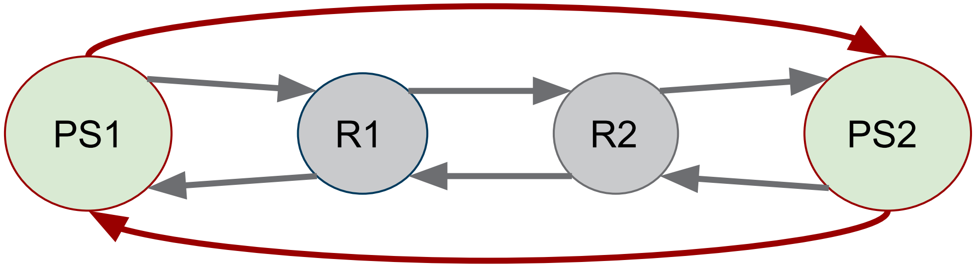
ESnet

# Data: What have we done?




- Shorter term transient effects
- Project forward (time) for results
- More specific than statistical behaviors

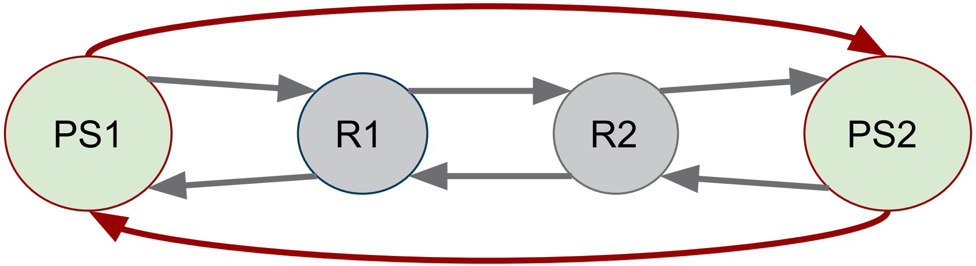
# Network Health




| Perfsonar: Node to Node | SNMP Link Data: Router Link |
|---|---|
| latency_mean<br>latency_sd | bytes_in<br>bytes_out |
| packet_lost | error_in<br>error_out |
| packet_reorder<br>packet_dupe | discard_in<br>discard_out |

ESnet

# Network Health: Time and Value Normalization




Stardust/PerfSonar

Sample Time: ~15 min

Predict window: 1 hour

Stardust/SNMP

Sample Time: 60 sec

Predict window: 1 hour

Link vs. PS to PS

End-to-end measure vs. Link measure *introduces question about where and when the lost/reorder/duped packets happened*

ESnet

# Network Health: Prediction

Initial work based on successful use of GNN in predicting Transatlantic WAN link traffic volumes over 24 hour windows.

(Based on Net Predict / DAPHNE)

# Network Health: Prediction

Initial work based on successful use of GNN in predicting Transatlantic WAN link traffic volumes over 24 hour windows.

(Based on Net Predict / DAPHNE)

Current state of the art in predictive networking takes a different approach - ML architecture for sequence modeling called *Transformer* which can take into consideration the context of the data point within the sequence.

Think of this as a LLM/Chat Gpt for network traffic volumes and congestion/errors

ESnet

# Our Agenda


So how do we get closer to our objective?

Lego 71004 Lego Movie Series 1 Lord Business #2 Minifigure CEM

# Research Agenda

**Project 1:** Prediction. Can we take historical time series data for traffic volumes and accurately predict future values on individual links?

**Project 2:** Data feature engineering. What core features in terms of historical behaviors as well as network health are used/measured/required for optimal solutions.

**Project 3:** Optimization. How can we identify an optimal solution in terms of routing solution in terms of combining historical site behavior, current topological design, and predicted values?

**Project 4:** Routing integration + testing. How do we quantify changes to the (proposed) network to ensure that nothing bad is happening from our diffs?

ESnet

# Project 1: Prediction

- Can we take historical time series data for traffic volumes and accurately predict future values on individual links?
- Can this be done for errors, loss, jitter, and retransmits as well?
- There is some work already done here on this topic, but not anything in terms of code, data, and reproducibility.

Output if successful:

- (Transformer) model capable of predicting bandwidth and (possibly) related error values 1-24 hours in advance
- Paper describing method and related code
- Explore releasing model if possible

ESnet

# Project 2: Data feature engineering

- Look at current workflow wrt unsupervised clustering of historical network behavior from sites
- Measure continuity across time to ensure that <net1> <-> <net2> generalizations are stable enough to be used for optimization purposes
- Are there additional data features beyond the usual candidates? Think about this in terms of higher degree moments to better express stability and efficiency in prediction/optimization
- Explore **HT** as a data source: what (if any) new or additional metrics can we look at for health prediction and route optimization?

ESnet

# Project 2: Data feature engineering

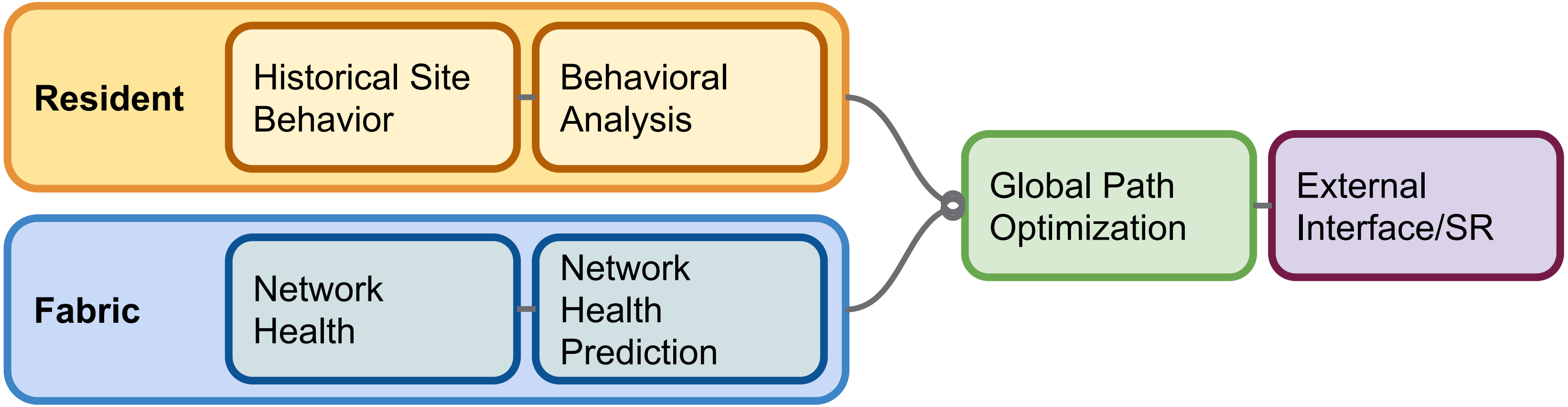
Output of a successful project:

- Paper describing in detail how unsupervised clustering works across time and at scale for site related analysis
- Confirmation that the health values being looked at are relevant - are there others that could be used with greater utility?
- What would very high time resolution bring to a statistical measurement?

ESnet

# Questions?

FIN




ESnet

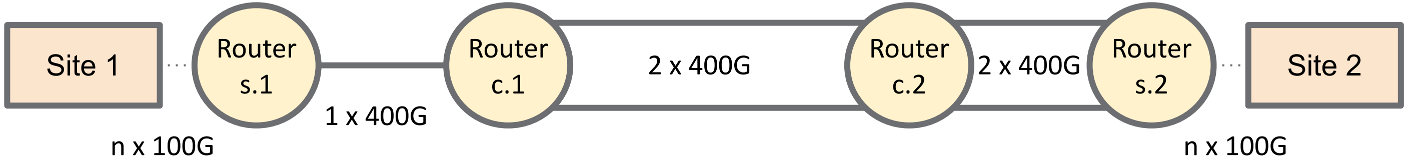
This slide intentionally left blank

# [1] Required Abstractions

$Site_1$ to $Site_2$: **Flow records** via ASN
dt = A

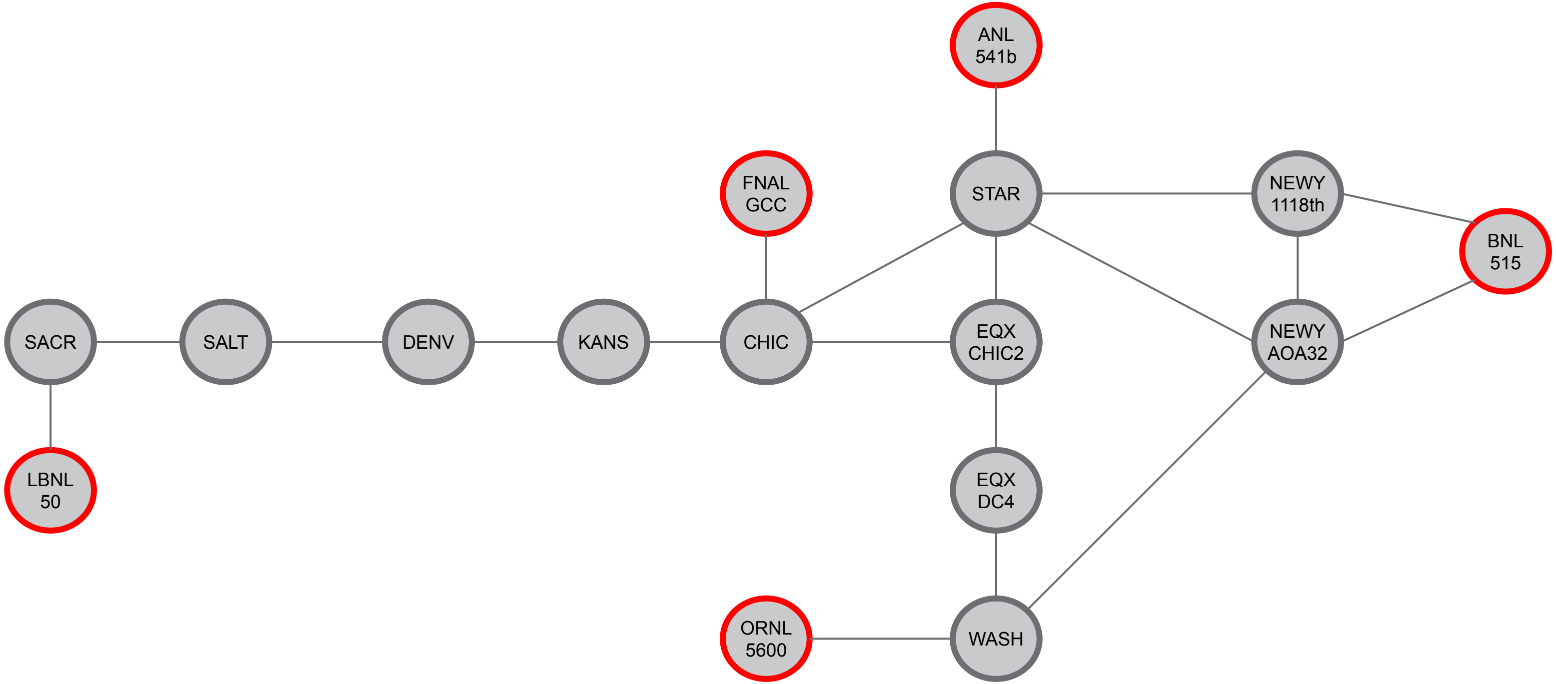

$PS_1$ to $PS_2$:
**Latency,Jitter,Drops,Rxmits**
**dt = B**
**Active test data**
**Unidirectional**


SNMP <u>**Interface Link**</u> **Bandwidth** Actual, drops,discards
dt = C
Passive data
Bidirectional




ESnet

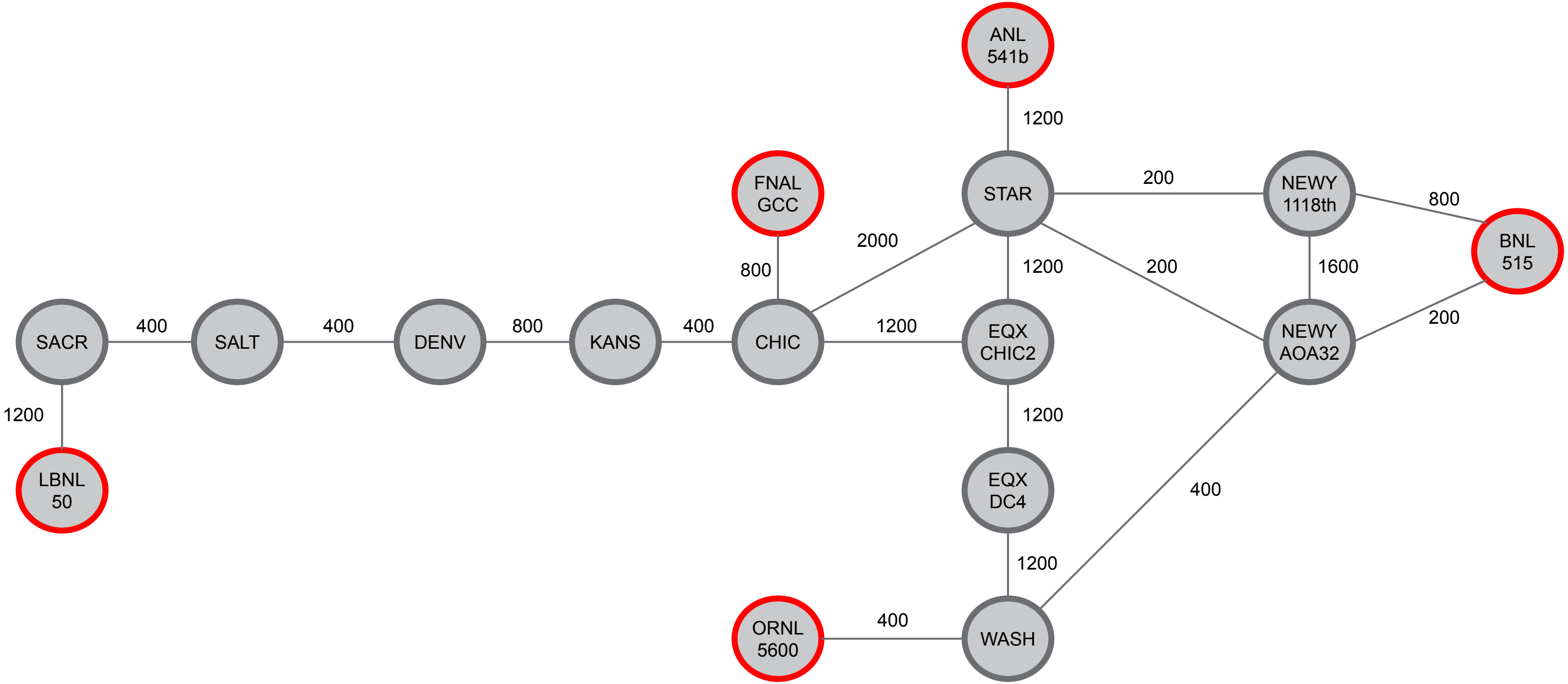
# [2] Link Mesh: Logical




Data from Dec 2022

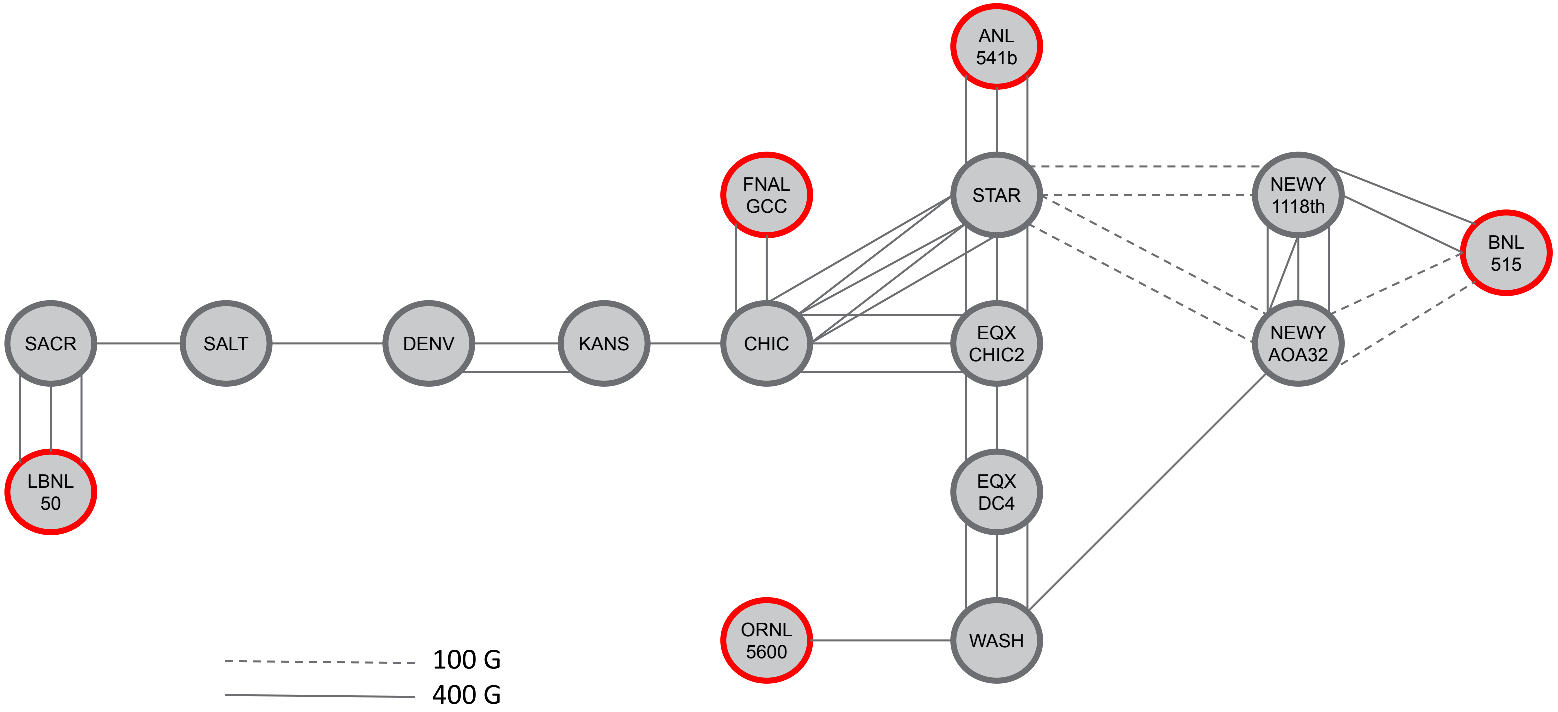
ESnet

# [3] Link Mesh: Logical, Weighted



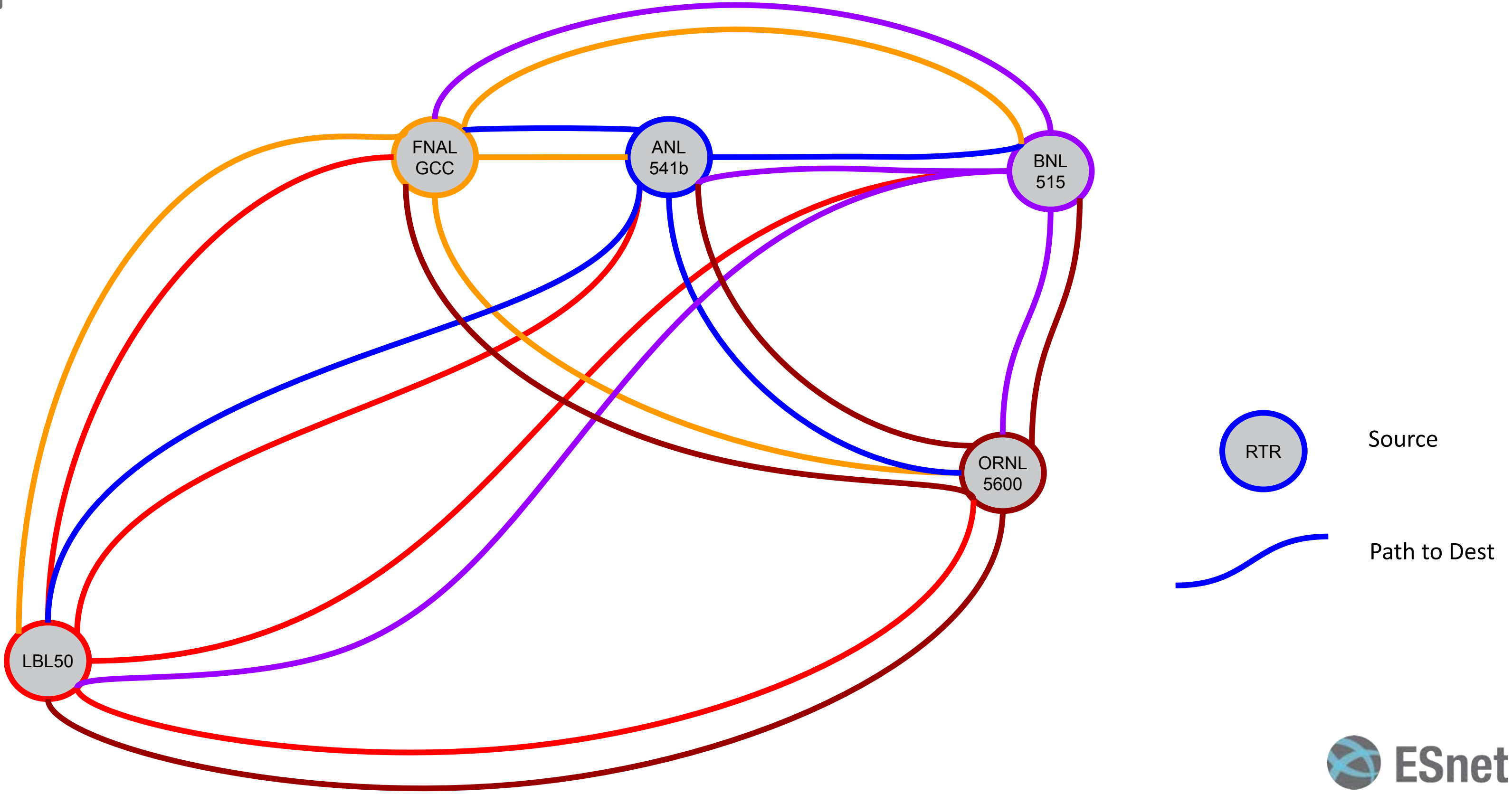

Data from Dec 2022

# [4] Link Mesh: Physical



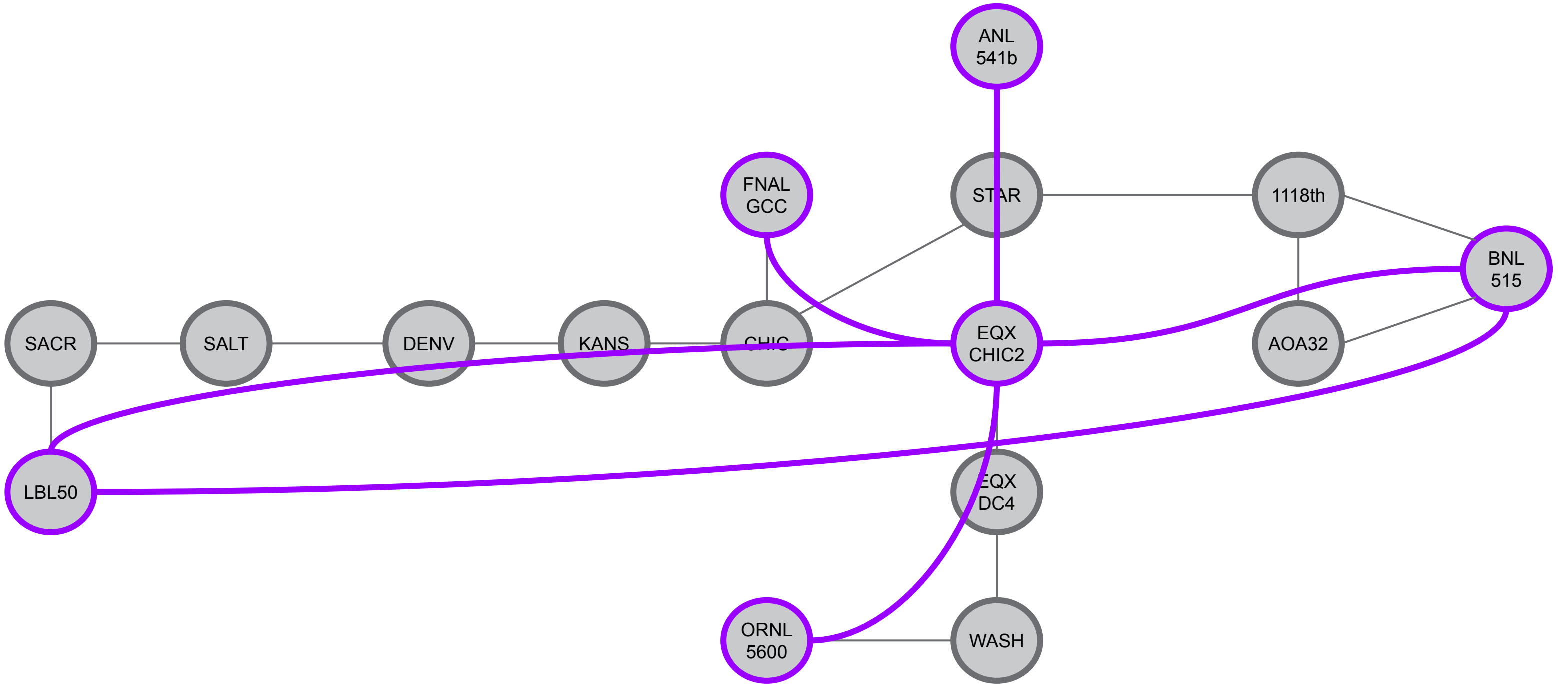

Data from Dec 2022

# [5] PS Mesh: Abstract




ESnet

# [6] PS Mesh: Real




All Data **Directional**

ESnet