

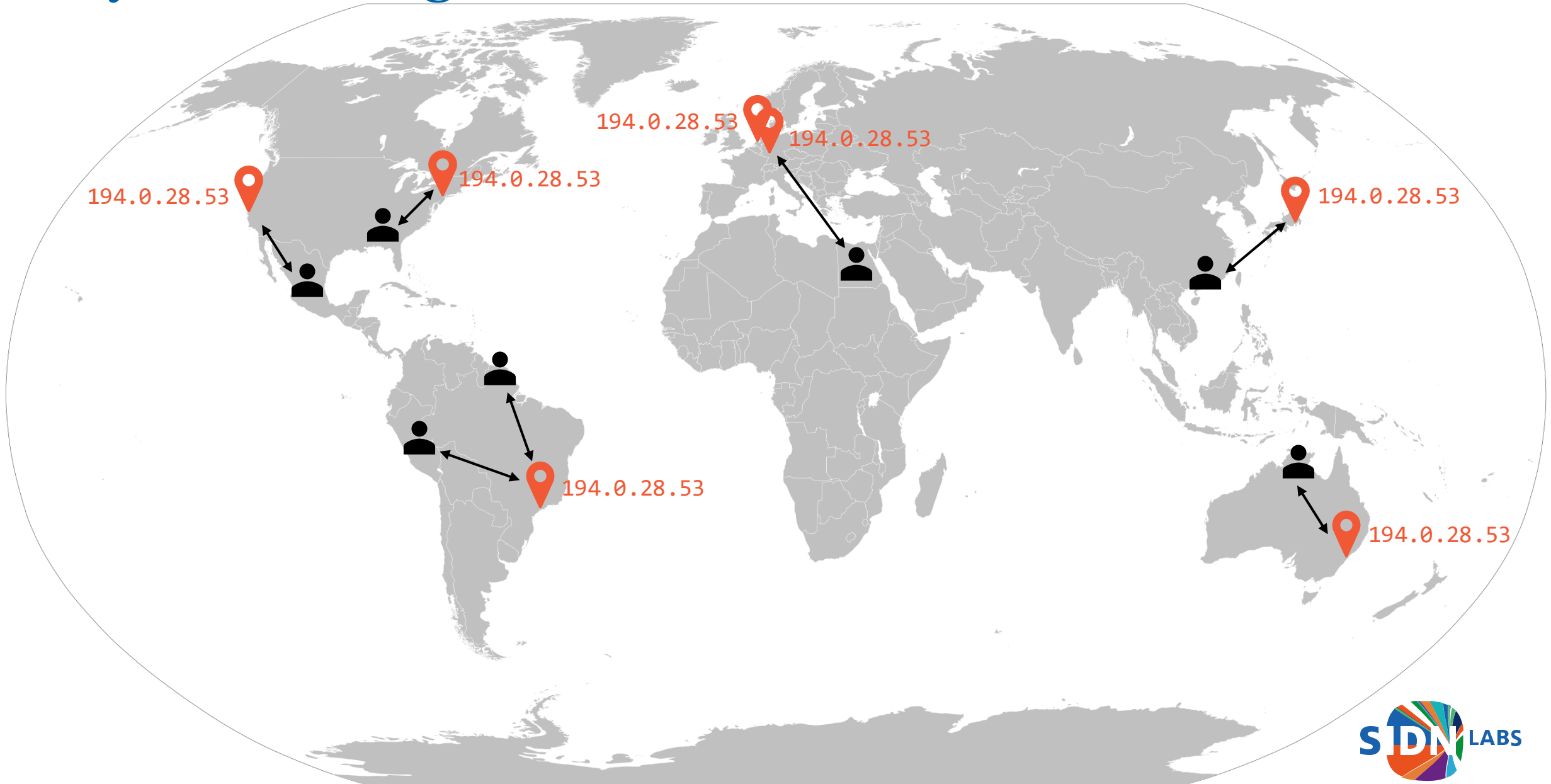
Autocast: automatic anycast site optimization

Thijs van den Hout

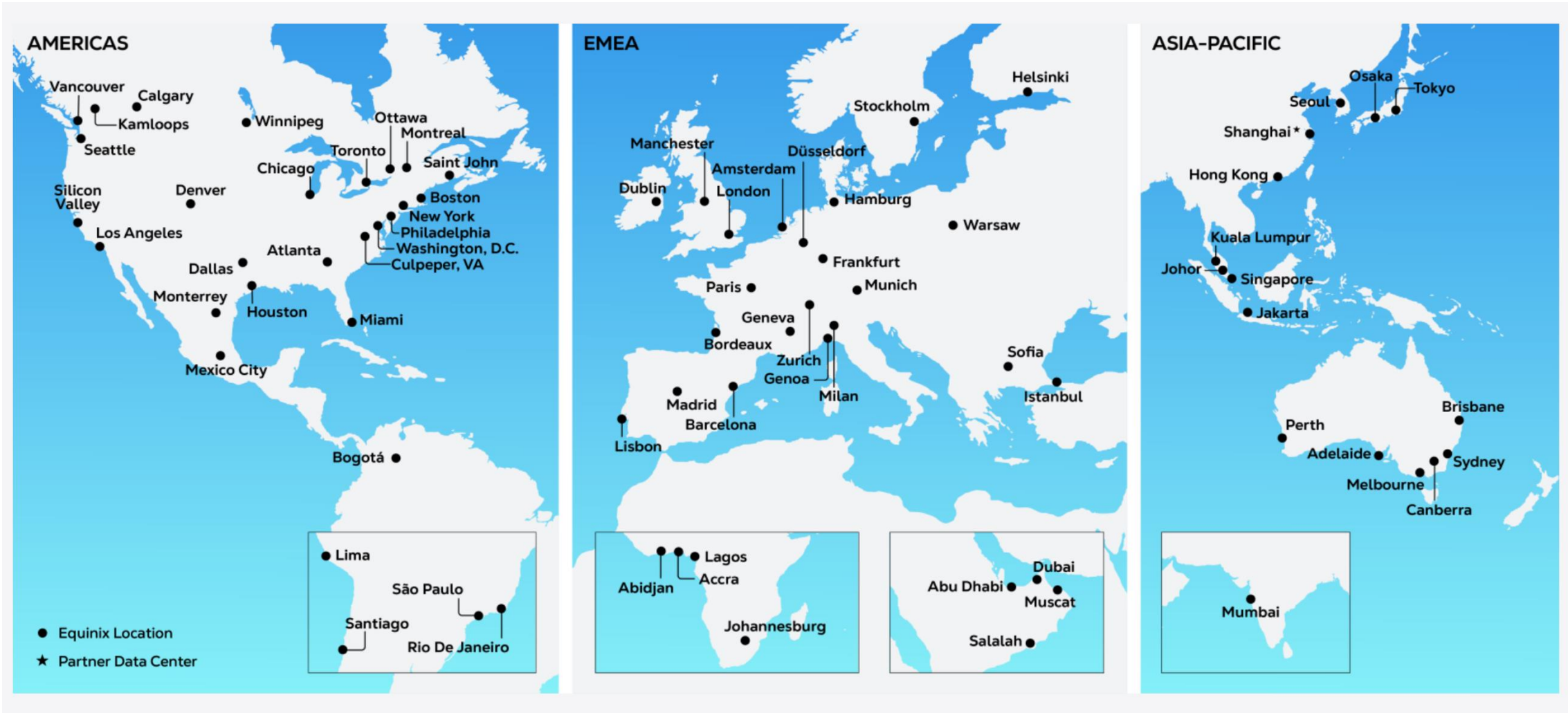
DNS Week, Stockholm | October 8, 2025



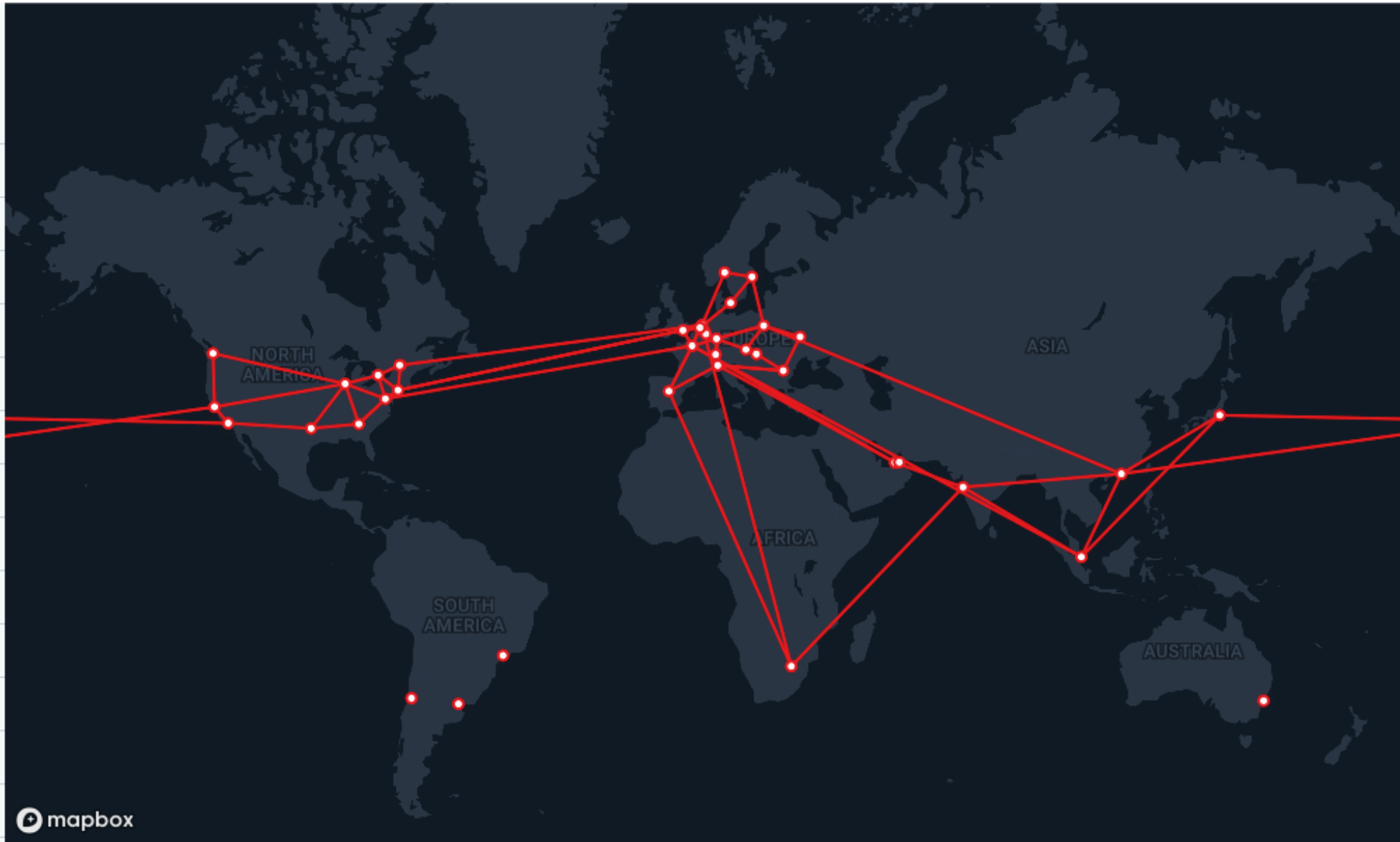
Anycast routing



Anycast, anywhere: Equinix



Anycast, anywhere: i3D.net



Anycast, anywhere: Vultr



Which anycast locations do we choose?

We want to: minimize latency, ensure resilience, save on costs

- Best practices, literature, constant tweaking
- All the locations?!
- Can we do it based on measurements? Data-driven
- Select 10 sites from 40 points of presence => 847M combinations
- Performance of all possible anycast configurations cannot be measured

Autocast (automating anycast)

A measurement-based anycast site selection method to minimize client latency

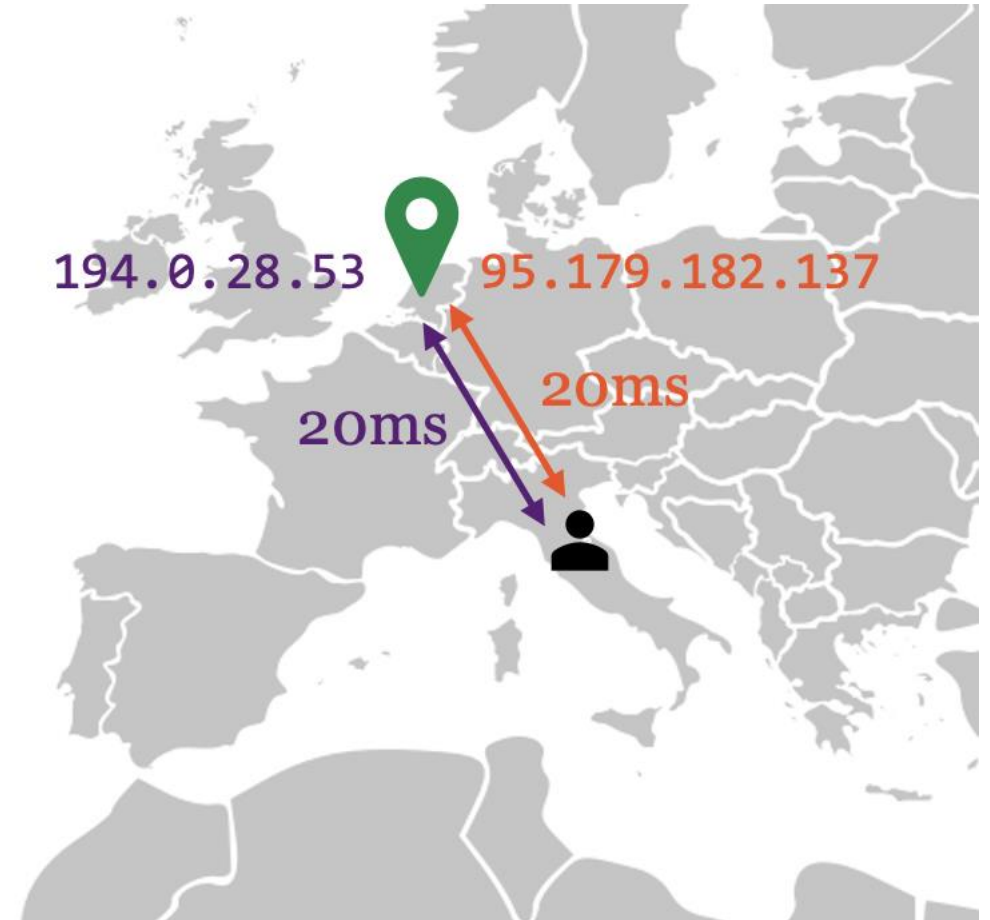
Requirements:

- Fast results
- Accurate predictor
- Rooted in operations

Autocast assumptions

Assumption 1: clients are routed to the anycast site with the lowest latency

Assumption 2: anycast and unicast latencies to the same anycast site are the same



Method: Measuring the unmeasurable

Step 1: measure the latency from all possible anycast PoPs to all clients

Ping RTT	Site A	Site B	Site C	Site D	...
Client 1	14.3ms	65.6ms	22.1ms	6.7ms	...
Client 2	21.0ms	84.1ms	9.2ms	26.1ms	...
Client 3	40.5ms	9.3ms	38.5ms	31.1ms	...
...	...	..	...	...	...

Measuring the unmeasurable

Step 1: measure the latency from all possible anycast PoPs to all clients

Step 2: simulate anycast site selections and compute expected latency.

Ping RTT	Site A	Site B	Site C	Site D	...
Client 1	14.3ms	65.6ms	22.1ms	6.7ms	...
Client 2	21.0ms	84.1ms	9.2ms	26.1ms	...
Client 3	40.5ms	9.3ms	38.5ms	31.1ms	...
...	...	..	...	...	...

Measuring the unmeasurable

Step 1: measure the latency from all possible anycast PoPs to all clients

Step 2: simulate anycast site selections and compute expected latency.

Ping RTT	Site A	Site B	Site C	Site D	...
Client 1	14.3ms	65.6ms	22.1ms	6.7ms	...
Client 2	21.0ms	84.1ms	9.2ms	26.1ms	...
Client 3	40.5ms	9.3ms	38.5ms	31.1ms	...
...	...	..	...	...	...

Measuring the unmeasurable

Step 1: measure the latency from all possible anycast PoPs to all clients

Step 2: simulate anycast site selections and compute expected latency.

Ping RTT	Site A	Site B	Site C	Site D	...
Client 1	14.3ms	65.6ms	22.1ms	6.7ms	...
Client 2	21.0ms	84.1ms	9.2ms	26.1ms	...
Client 3	40.5ms	9.3ms	38.5ms	31.1ms	...
...	...	..	...	...	...



A+D latency
6.7ms
21.0ms
31.1ms
...

Optimization

- Millions/billions of possible anycast site selections
- Metric to optimize: median resolver latency weighted by nr queries

B+C latency
22.1ms
9.2ms
9.3ms
...

15.3ms

A+D latency
6.7ms
21.0ms
31.1ms
...

21.1ms

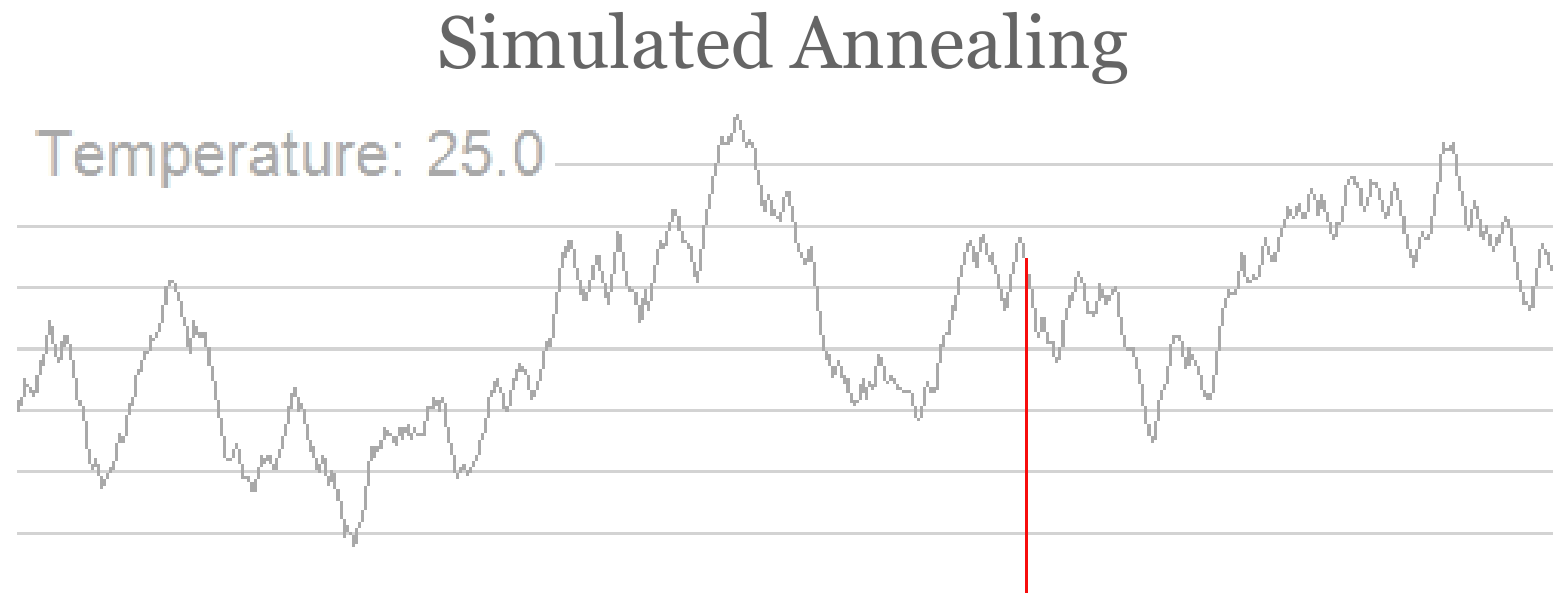


Illustration: Wikimedia

Putting it to the test on Vultr

Measurement

- Automatic infrastructure deployment on Vultr (32 PoPs)
- Client hitlist of .nl-querying resolvers (230K IPv4 addresses and 73K IPv6)
- Unicast ping measurement from all PoPs to all clients (73% response)

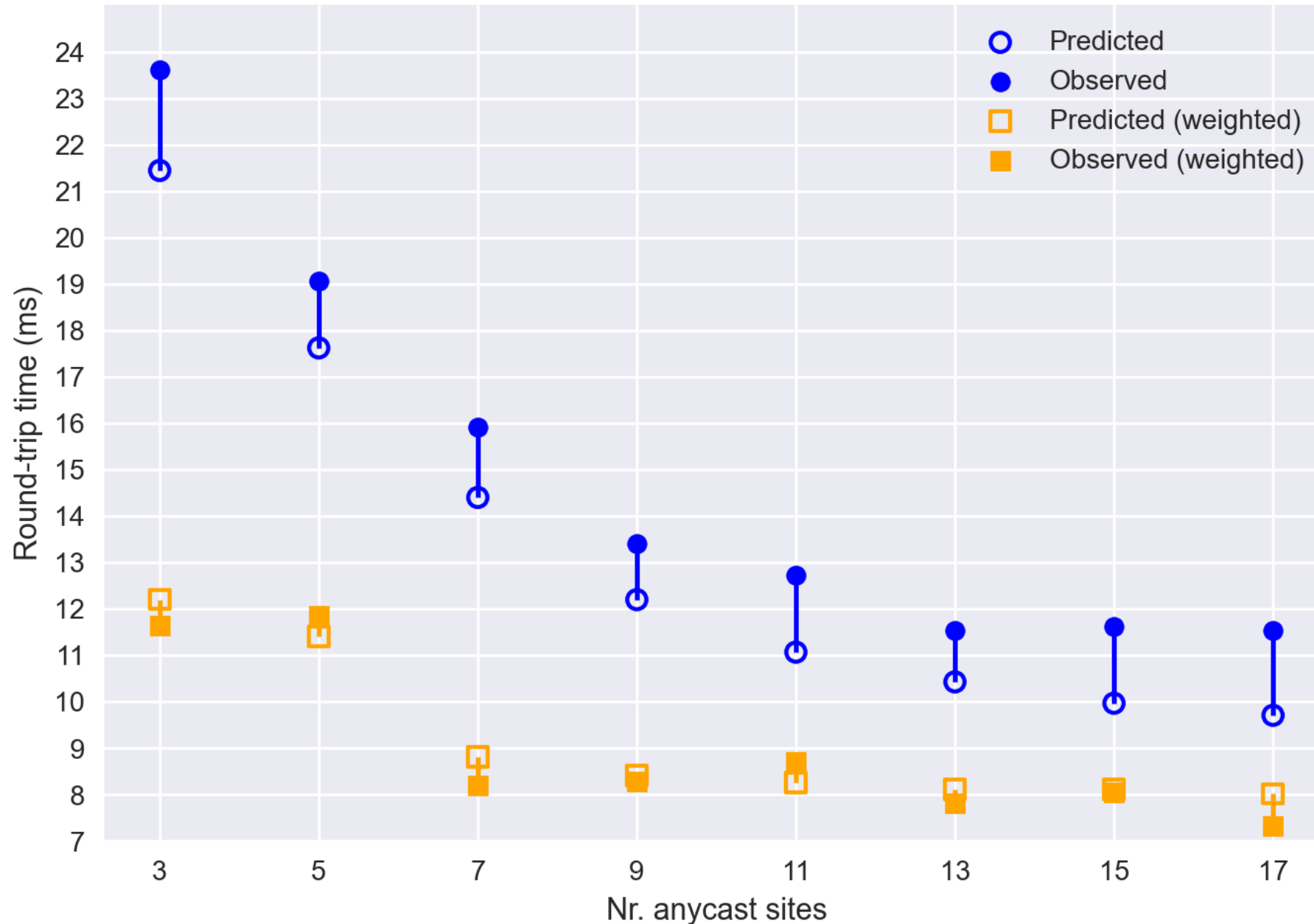
Putting it to the test

Optimization

- Generate anycast deployments for 3, 5, 7, 9, 11, 13, 15, 17 sites
- Run optimization algorithm for each number of sites => 8 deployments
- Generate 5 random configuration for each number => 40 deployments
- Deploy them with anycast and evaluate!

Results: Autocast-suggested deployments

Predicted vs. observed RTTs

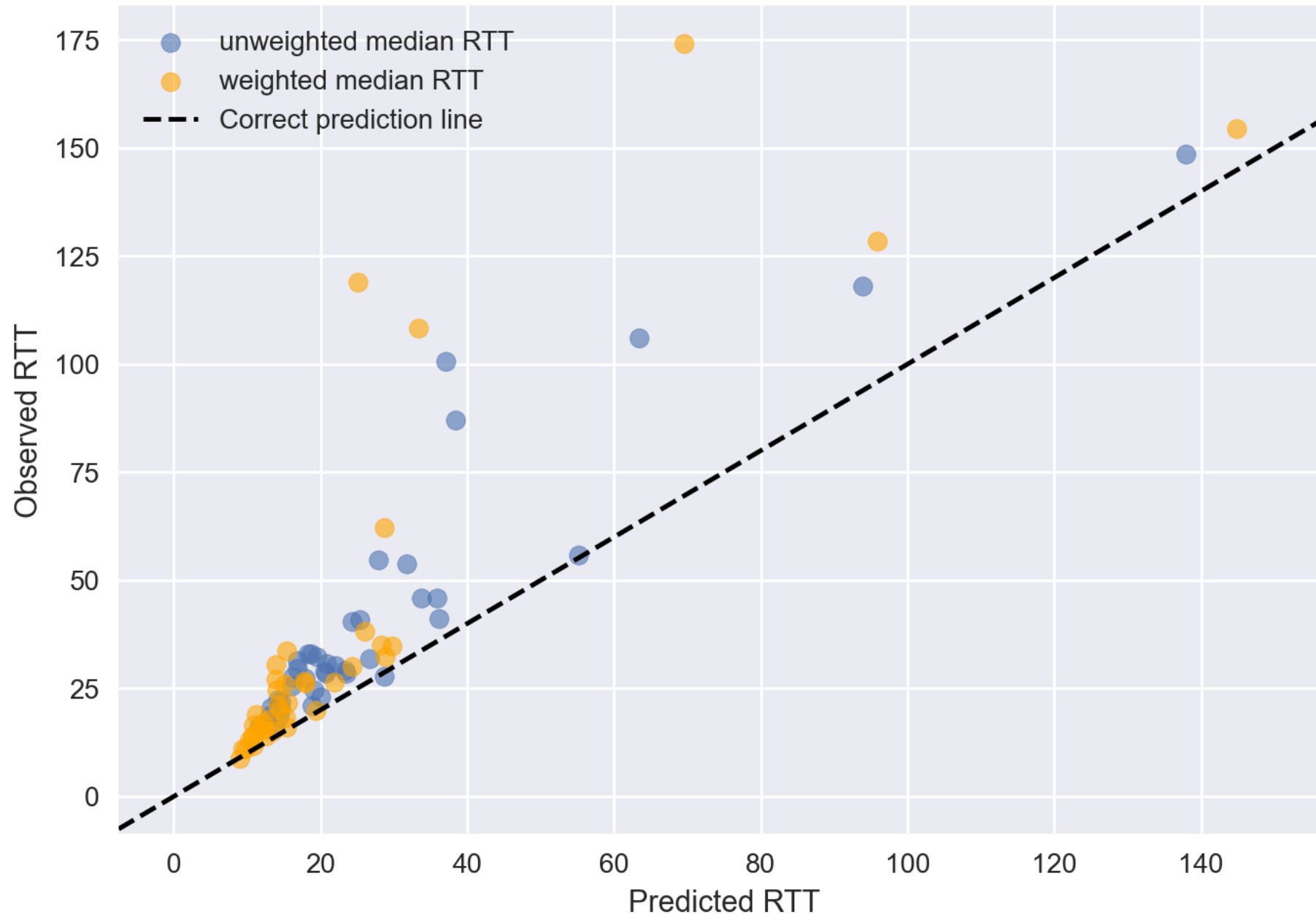


Predicted vs
observed latency
diff: 1.57ms

Weighted by nr
queries: 0.19ms

Results: random configurations

Observed vs predicted RTT for random anycast configurations

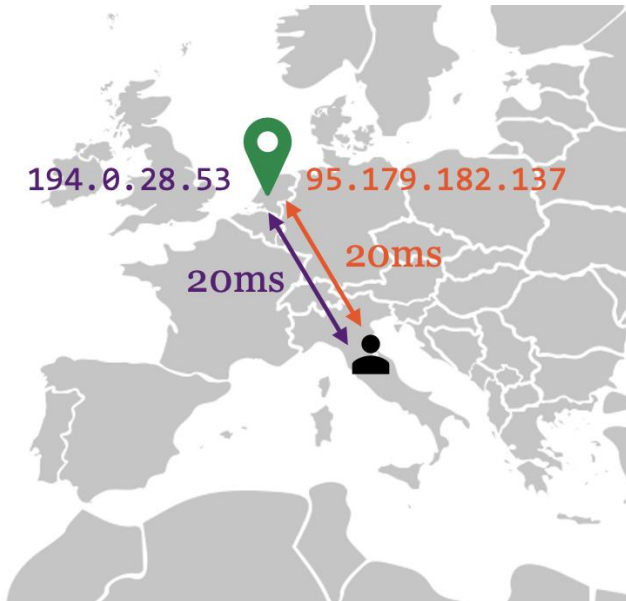
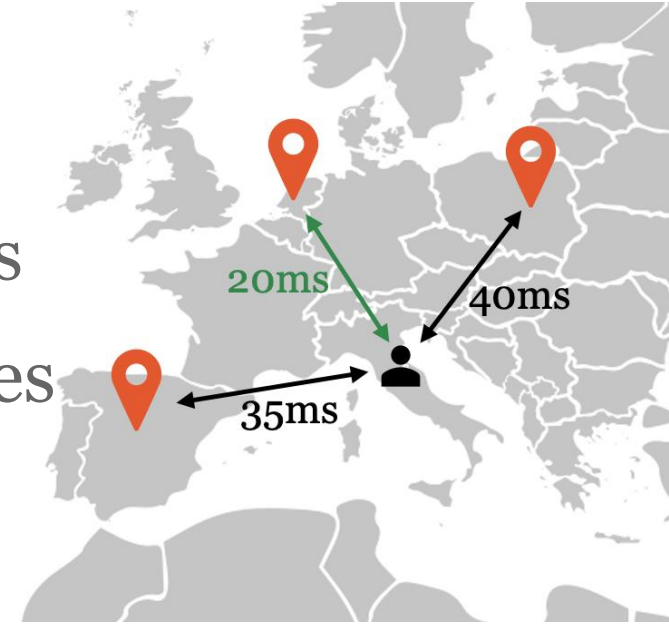


Re: Assumptions (or shortcuts)

Assumption 1 violations

Optimal deployment: 26.8% of clients, 11.8% of queries

Random deployment: 40.8% of clients, 34.5% of queries



Assumption 2

Median difference: 0.06ms

$\mu=0.29\text{ms}$, $\sigma=14.6\text{ms}$, Cohen's $d=0.02$, sample size of 284K

Conclusions

- It seems to work quite well!
- Assumption 1 holds well enough to predict performance
- Fast results and accurate predictions
- Use in production