BENOCS

Stop guessing. Start knowing

The invisible impact of network handovers within content delivery

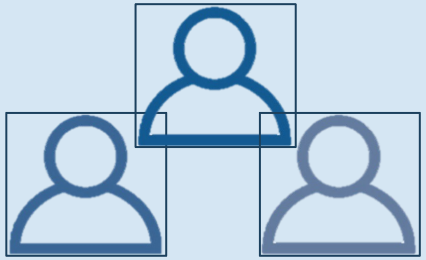
with CDNs and compute moving deeper into the edge,
a few challenges occur which need to be addressed jointly

2022

Focus areas CDN ⇔ ISPs

Capacity

- Utilization vs. headroom ⇒ Potentially conflicting priorities
- Multi-CDN vs. stability ⇒ Volatile traffic patters, unpredictable
- Indirects (peers, transit) for overflow ⇒ Might affect unrelated services, $$$
- Caches ⇒ Redundancy, cache-fill, space & power



Localization

- Deliver traffic close to end-user ⇒ Minimize RTT/latency
- Avoid unnecessary backhaul ⇒ Minimize backbone load
- Keep delivery profiles stable ⇒ Allow for high utilization

Capacity challenges

Joint planning and transparent parameters can reduce investment needs and improve resilience

Trend to the edge: will shrink backbones

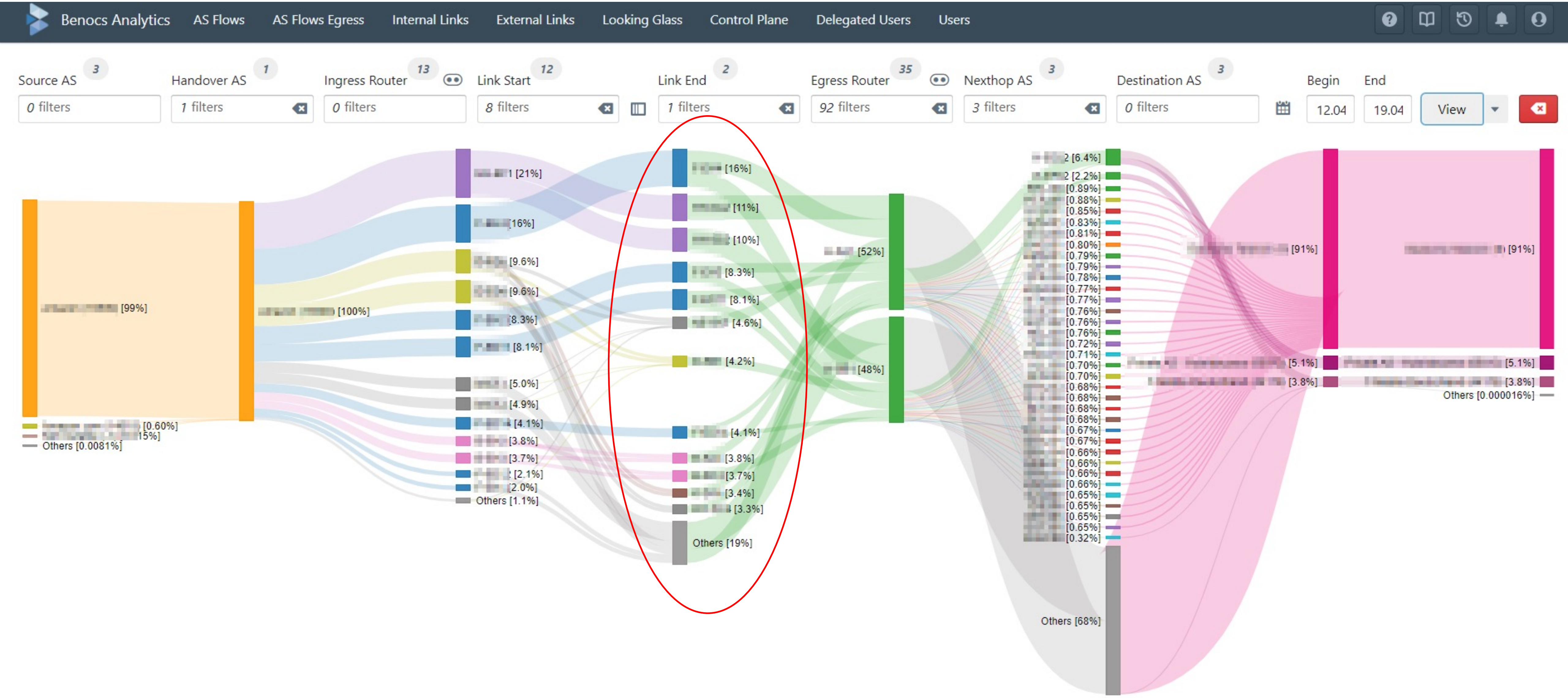
- Good for customer experience
- Good for traffic distribution cost (network)

- More headroom per node needed for traffic spikes
- Failover-concepts are currently poorly aligned

=> Backbones will – at one point - not be able to handle overflow

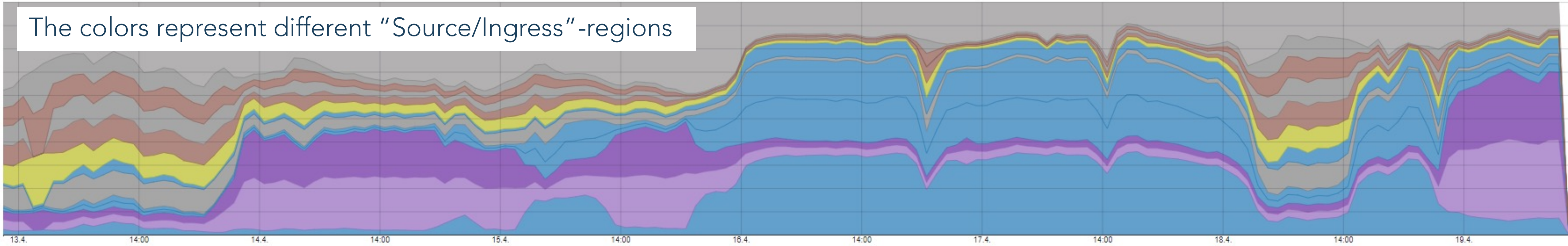
=> Localization and stability will become key requirements

Capacity planning – backbone links into Region A



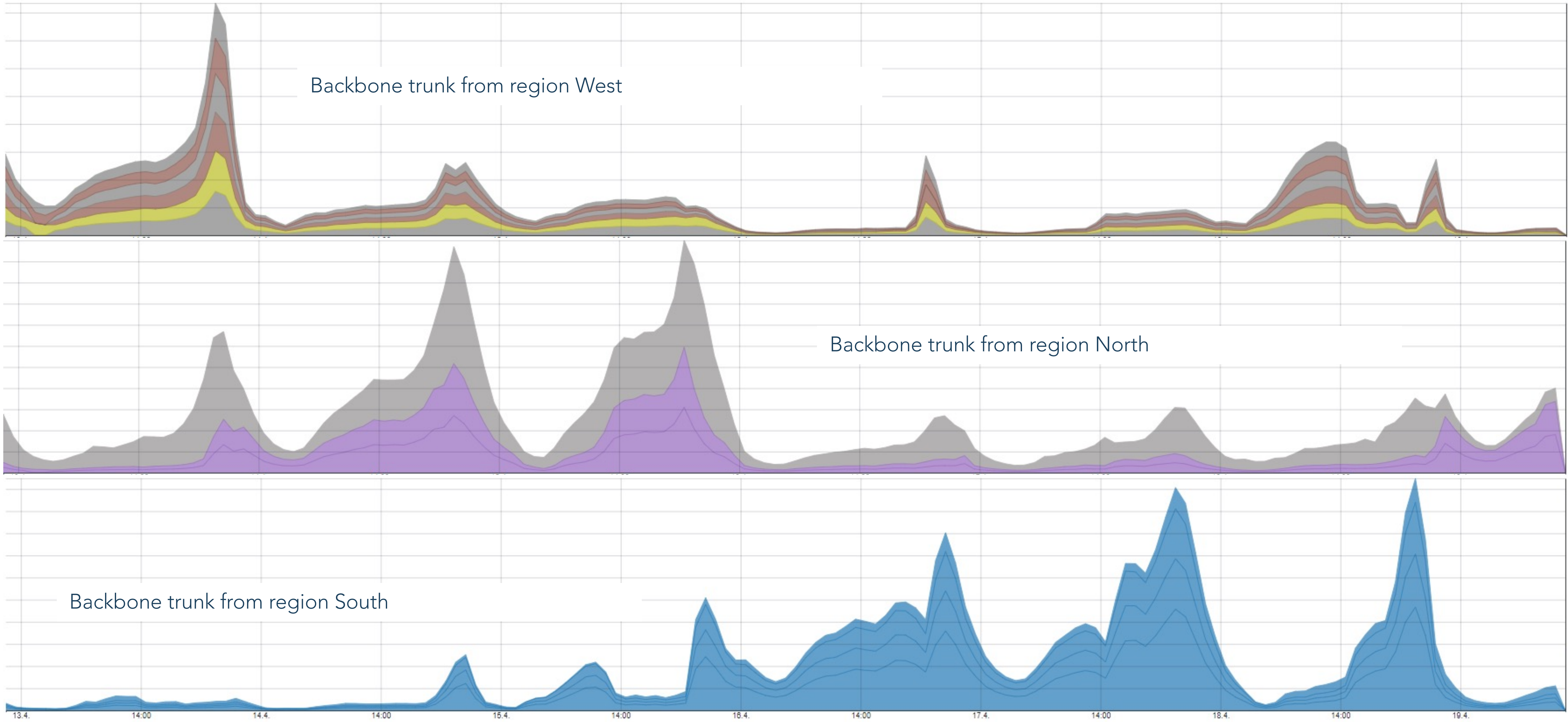
Capacity planning – fail

This is the 7-day relative traffic profile
of server-clusters serving a specific end-user region in an ISP network.



=> The ISP needs to maintain **3x backbone capacity** (from all three remote regions) due to volatility

Ingress instability – resulting in 3x capacity needs



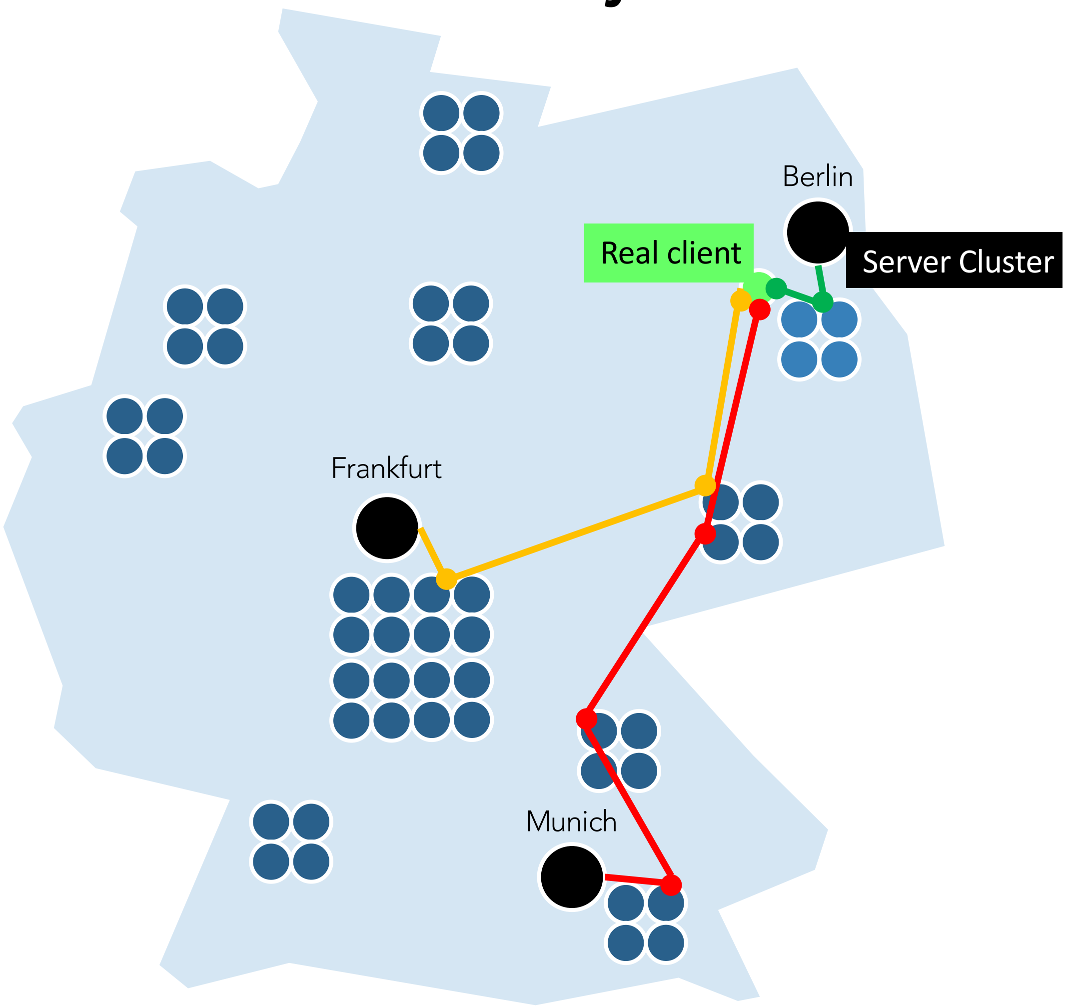
Localization challenges

When you invest in regionalizing servers,
make sure your traffic-regionalization keeps up

Why is localization important?

Ingress (Server-Cluster)	Metric (hopcount, km)
Berlin	1,25
Frankfurt	3,75
Munich	6,89

Germany



How CDNs currently resolve this?

Workaround	Mechanism	Downside
Anycast	Egress = Ingress	No failover, requires all content everywhere, no load control, ignores outbound policies
Identifying DNS resolvers to locate users	Group all subnets of region in one vDNS	Complex configuration, does not work properly in daily life, failover issues
Roundtrip measurement	Send via lowest RTT path	Roundtrip is misleading for asymmetrical in/outbound paths. eBGP ≠ iBGP
Geo-locating users with internal or external databases	Acquire Geo-IP from 3rd parties	Accuracy issues for neighboring locations, typically outdated, not capacity-aware

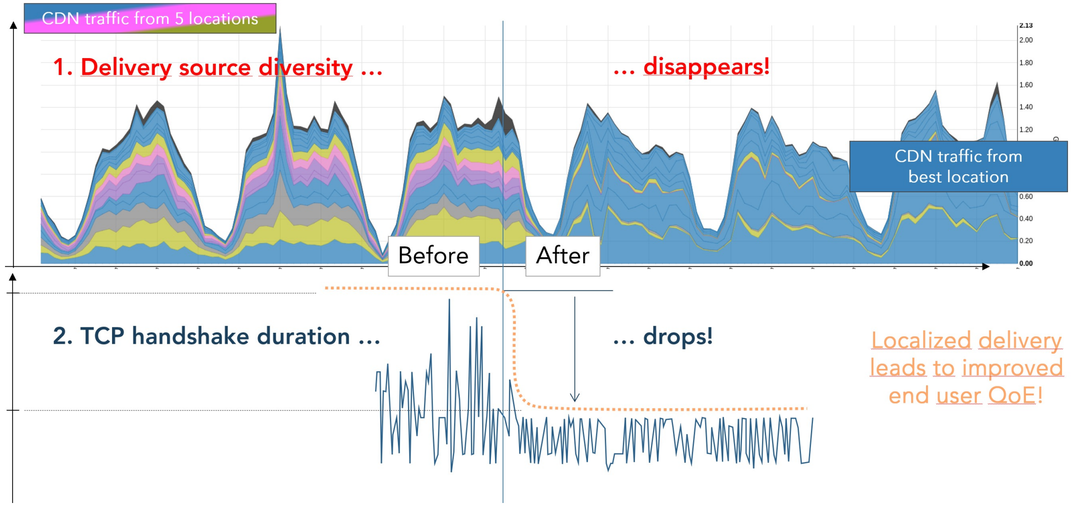
Identifying DNS-resolvers to locate users

- Only few addresses to monitor
- DNS locations stable and easy to communicate
- No ECS/EDNS0 required

- DNS-resolvers get load-balanced and ”mis "configured (1)
- Fall-back resolver locations can be far-off
- Solution fails for DoH (8.8.8.8, 1.1.1.1) and Smart TV / IoT

Identifying DNS-resolvers – fail 1

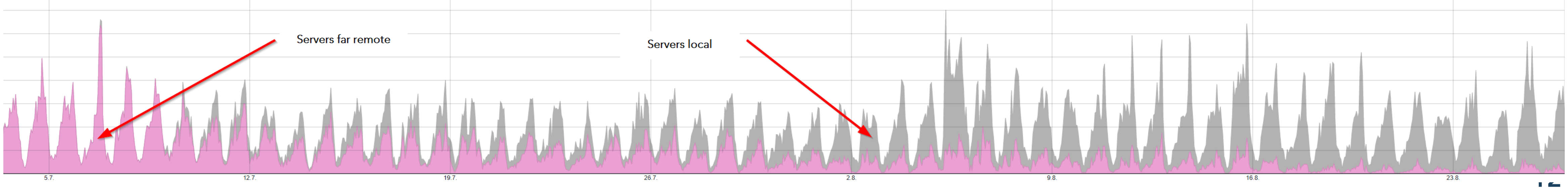
①



In this example, MNO had configured their resolvers in round-robin load-balance,
unaware of the impacts this would have to the CDNs mapping efforts.

②

In this example, ISP had accidently configured remote resolvers as primary.
CDN was thus using remote servers for delivery. Due to multi-months forced reconnect, is took > two months to age out.



Geo-locating users with internal or external database

- Databases are broadly available
- Straightforward and seemingly working
- No CDN-ISP engagement needed

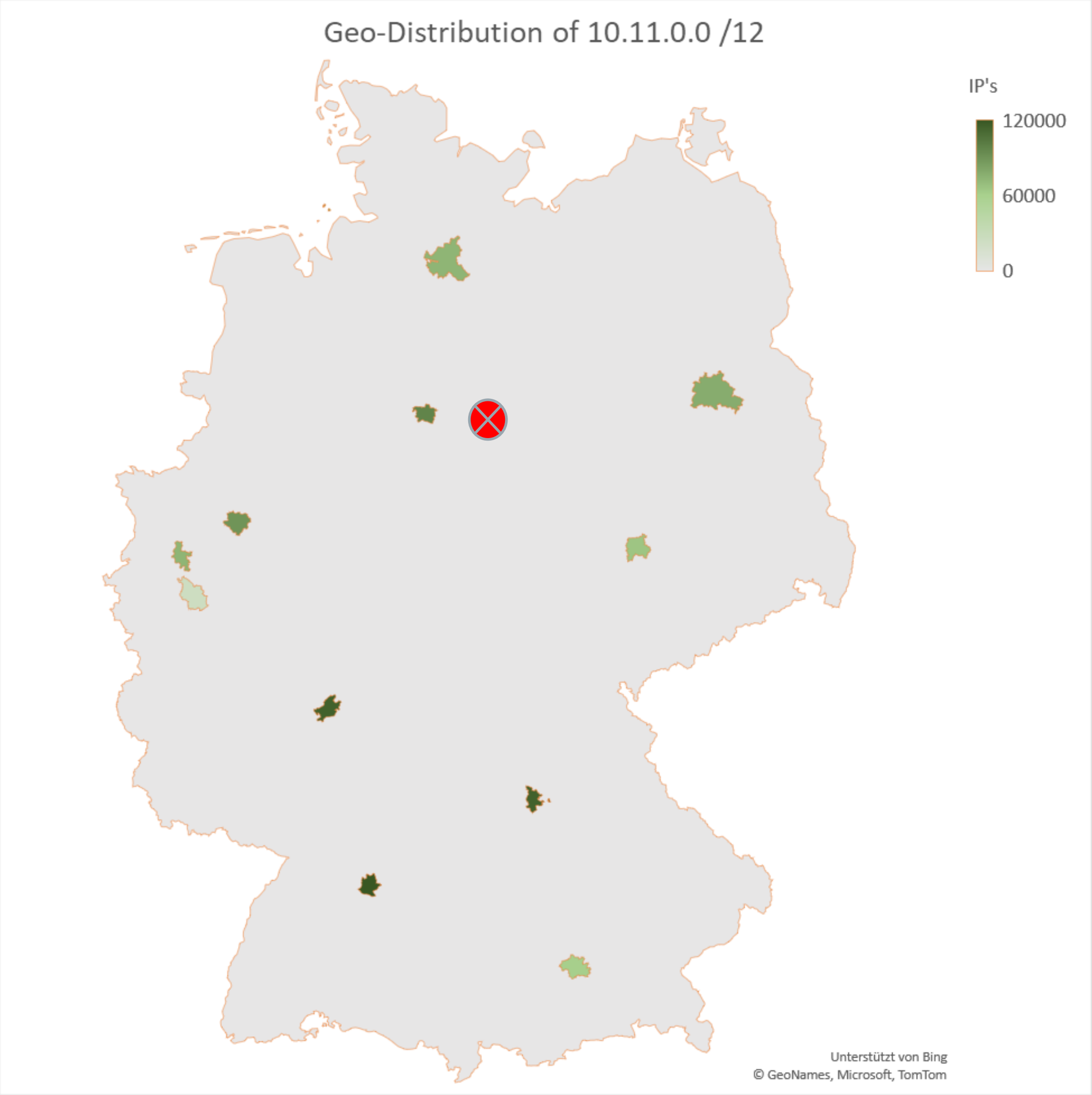
- eBGP aggregates ≠ iBGP subnets (1)
- Update-delay for refarmed subnets (2)
- Geo-distance ≠ Network-distance (3)
- Ignores roundtrip-reality (i.e. outbound path) (3)
- Often inaccurate, no reliable quality check

Geo-locating fail 1: iBGP vs eBGP / aggregated vs specific

eBGP:

10.11.0.0 /12
⇒ 1.048.574 hosts

IP-Geo:
Braunschweig



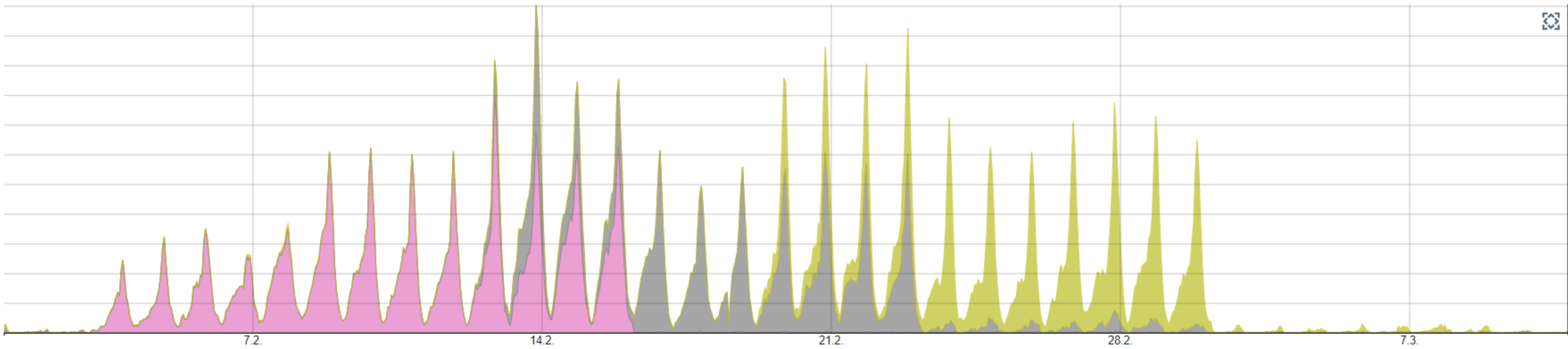
iBGP:

10.11.0.0 /12
⇒ 1.048.574 hosts
⇒ 220 (main) prefixes

Allocation (regions):
Munich
Stuttgart
Nuremberg
Frankfurt
Cologne
Düsseldorf
Dortmund
Leipzig
Hannover
Berlin
Hamburg

Geo-locating users – fail 2: update delay

- Here you see traffic delivered from a remote source to 3 local BNGs.
- After subnets were refarmed from one region to another,
  it took the CDN 2 weeks to learn the new geo-location



Geo-locating – fail 3: geo-distance ≠ network distance

Step 1:
Prefixes identified for Heilbronn (HLB):
10.11.16.0/20
10.12.48.0/20
10.13.80.0/20
10.14.144.0/20
…)

Step 2:
Geo distance roundtrip:
MUC-HLB: 414km (+77% vs best choice)
FRA-HLB: 234km

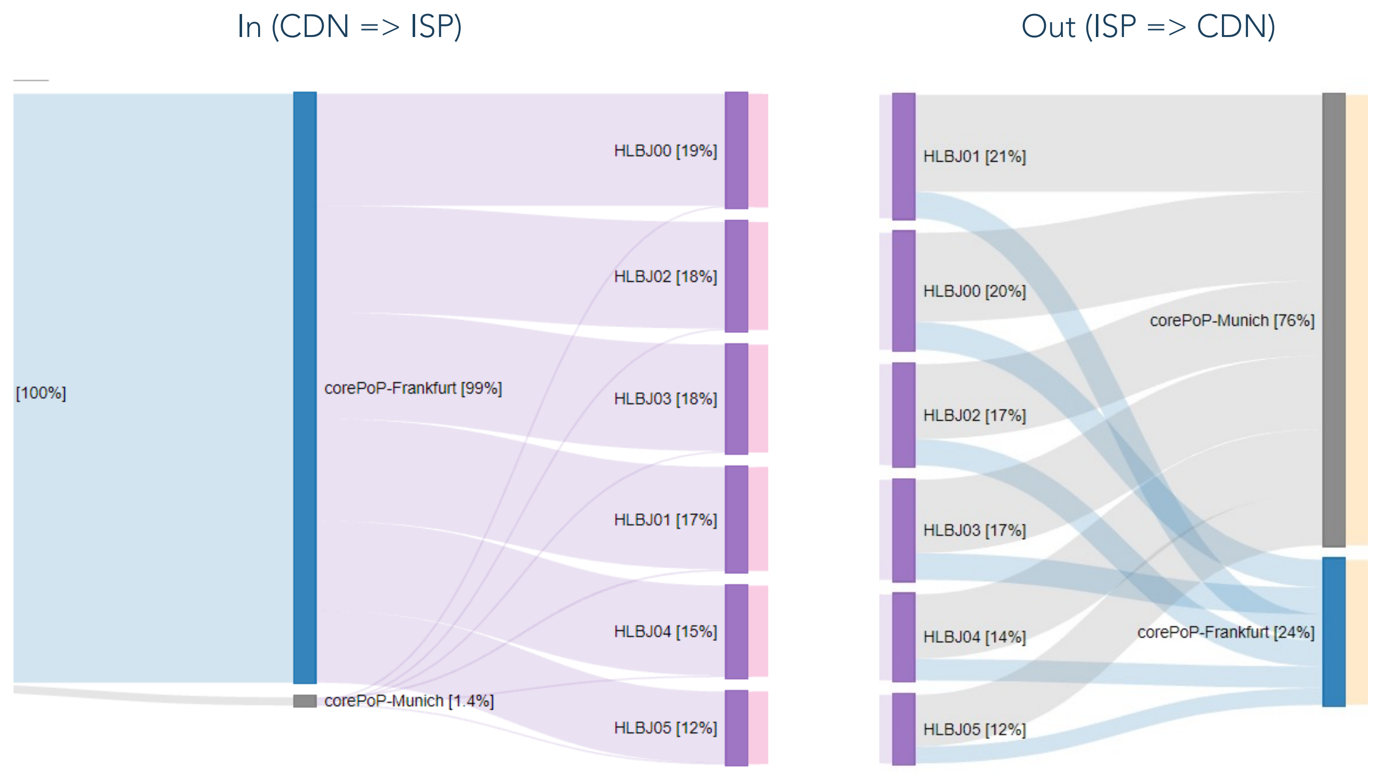
Network distance roundtrip:
MUC-STU-HLB: 462km (+18% vs best choice)
FRA-STU-HLB: 392km

Step 3:
Reality check with actual routing (roundtrip):

MUC-STU-HLB-STU-MUC: 462km
FRA-STU-HLB-STU-MUC-FRA: 735km (+60% vs best choice, 3x of best Geo-choice)

Geo-locating users – fails ct'd

③



Be aware of asymmetrical paths! In this example, all traffic to this city is delivered from Frankfurt, but most sessions return via Munich!

With simple Geo-location, CDNs traffic risks taking the scenic tour through the ISP's network, and it needs to be carried back on CDN's backbone

Ignores roundtrip-reality (i.e. outbound path)

Our solution: Real-time data exchange between ISP & CDN

Reads full topology and traffic status, exports to CDN in real-time which requires installation in the network. This is mapping answers based on the ground truth.

Metric data exchange between ISP-CDN via API

#	Subscriber - Prefix	City	Router	Frankfurt F-ED11	Frankfurt F-EH1	Amsterdam AMS-SB1	Berlin B-EH3	Berlin B-EH2	Amsterdam AMS-SC1	Munich M-EF1	Munich M-EF2	Frankfurt F-ED12	Frankfurt F-EE1
1	149.224.0.0/18	Hamburg	HH-EA7	4,34	4,63	3,57	2,13	2,13	3,45	7,85	7,89	4,34	4,58
2	149.224.128.0/17	Hamburg	HH-EB5	4,34	4,63	3,57	2,13	2,13	3,45	7,85	7,89	4,34	4,58
3	149.224.64.0/19	Berlin	B-EC4	3,75	3,98	4,42	1,25	1,25	4,20	6,89	6,94	3,75	3,87
4	149.224.96.0/20	Berlin	B-EA7	3,75	3,98	4,42	1,25	1,25	4,20	6,89	6,94	3,75	3,87
5	149.233.64.0/18	Munich	M-EA8	3,24	3,12	9,32	6,54	6,87	9,20	1,17	1,32	3,24	3,15
6	149.237.201.0/24	Frankfurt	F-EB4	1,42	1,74	4,82	3,68	3,86	4,56	3,10	3,39	1,25	1,30
7	149.240.0.0/16	Munich	M-EA9	3,24	3,12	9,32	6,54	6,87	9,20	1,17	1,32	3,24	3,15
8	149.240.0.0/17	Munich	M-EB12	3,24	3,12	9,32	6,54	6,87	9,20	1,17	1,32	3,24	3,15
...	...	...	Others	...	...	...	...	...	...	...	...	...	...
...	...	...	Others	...	...	...	...	...	...	...	...	...	...
...	...	...	Others	...	...	...	...	...	...	...	...	...	...
24.560	149.240.128.0/17	Hamburg	HH-EB6	4,34	4,63	3,57	2,13	2,13	3,45	7,85	7,89	4,34	4,58
24.561	149.243.232.0/22	Munich	M-EB7	3,24	3,12	9,32	6,54	6,87	9,20	1,17	1,32	3,24	3,15
24.562	2003:0:1604:8000::/50	Berlin	B-EC6	3,75	3,98	4,42	1,25	1,25	4,20	6,89	6,94	3,75	3,87
24.563	2003:0:1604:c000::/50	Frankfurt	F-EA11	1,42	1,74	4,82	3,68	3,86	4,56	3,10	3,39	1,25	1,30
24.564	2003:0:1700:4000::/50	Hamburg	HH-EA8	4,34	4,63	3,57	2,13	2,13	3,45	7,85	7,89	4,34	4,58
24.565	2003:0:1700:8000::/50	Munich	M-EA12	3,24	3,12	9,32	6,54	6,87	9,20	1,17	1,32	3,24	3,15
24.566	2003:0:1700:c000::/50	Munich	M-EB9	3,24	3,12	9,32	6,54	6,87	9,20	1,17	1,32	3,24	3,15

Real-Time Database with path-ranking for thousands of prefixes

1. Metric can include hop count, kilometer, capacity, utilization
2. Thousands of prefixes and hundreds of routers
3. Updated in real-time

Thanks for the time. Any questions?

Hari Jayaraman
hjayaraman@benocs.com