Monitoring What You Don't Own

Stephen Strowes | Monitorama EU | 2018-09-04

First thing's first

you can easily m

The Problem

But once you put something into the wild,
you're at the mercy of *other people's networks*

slaps roof of internet this network can fit so many networks into it

Not only that,

You may have any combination of:

your own datacenters or machines in somebody else's datacenters
VMs inside VPS providers, cloud providers
Higher-level cloud services
a CDN to get your content closer to users
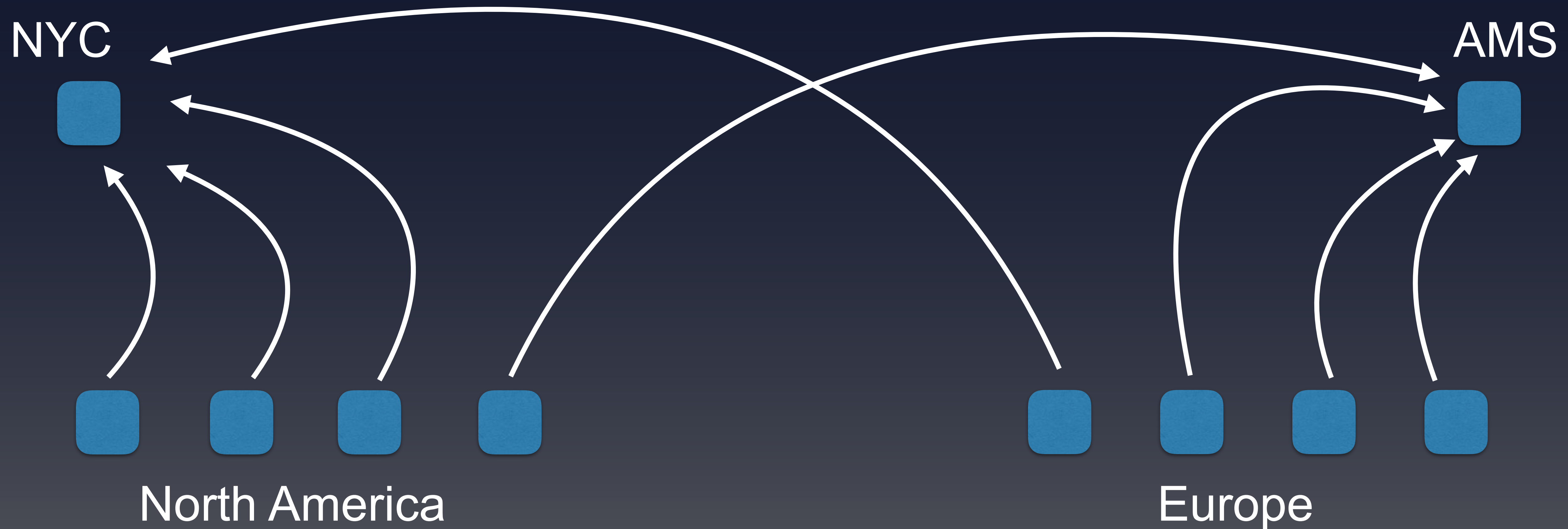
Not only that,

Any non-trivial user-facing deployment is likely to have some kind of anycast to split traffic across multiple locations

Is traffic going to the wrong place?

NYC

AMS

North America

Europe

What if: there's a problem out there across all this infra

What if: your alert is angry users on the internet? screaming, "**sivustosi ei toimi!**"

(this is alarmingly common)

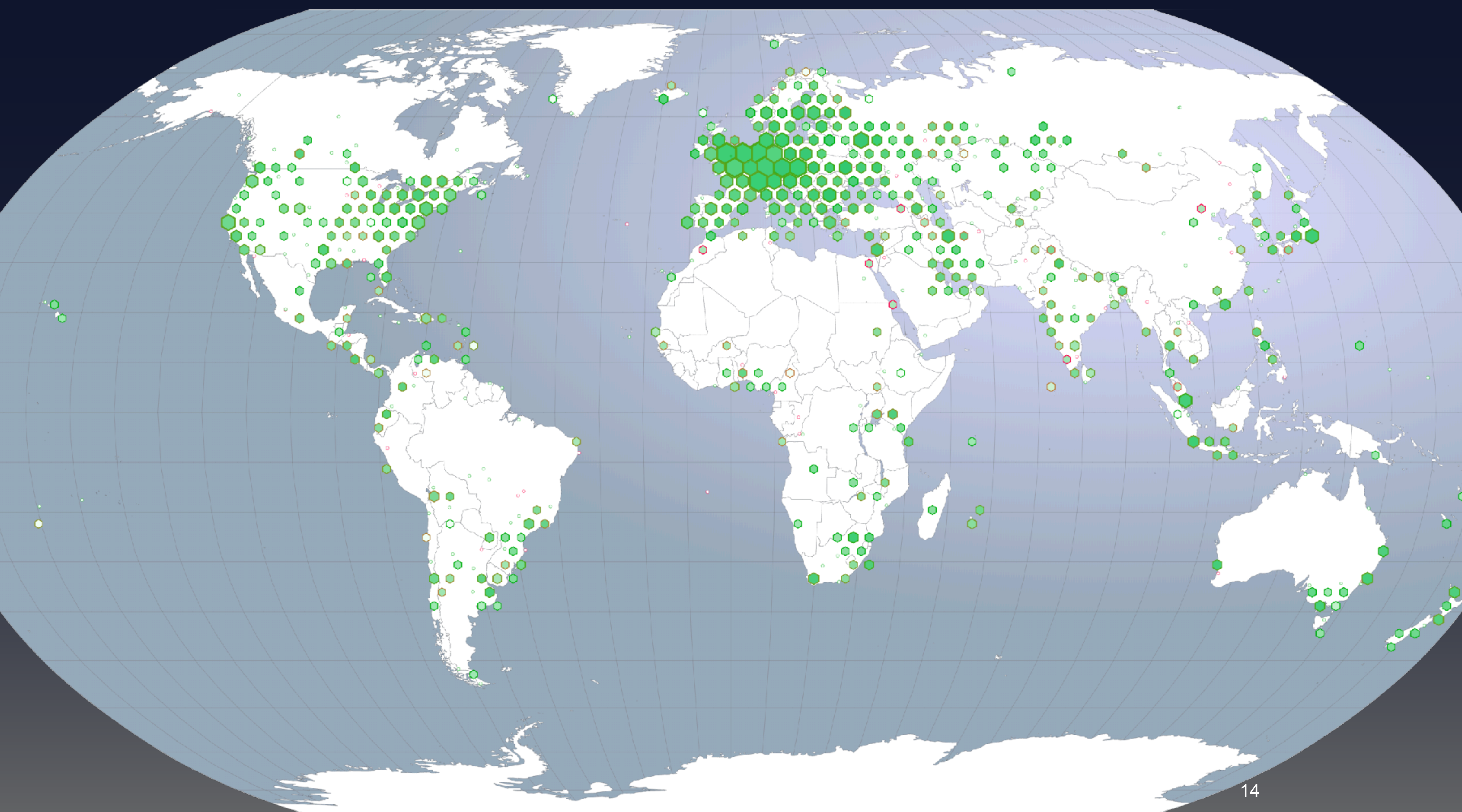Which of your users reach *which* of your locations?

Once you know that, can you isolate a network problem?

You need vantage points to measure, debug, or monitor the public network

RIPE Atlas

RIPE Atlas

- Measurements: ping, traceroute, DNS, NTP, etc

- All over IPv4 and IPv6

- Instruct the platform to run measurements

- Select probes to conduct those measurements

- Collect the results as JSON structs

Which of your users reach *which* of your locations?

Once you know that, can you isolate a network problem?

You need vantage points to measure, debug, or monitor the public network

DNS

Let's *resolve* this situation

- Say you want to understand what is being resolved out there

  - where do ISPs mess with your DNS records?

  - where are TTLs not honoured?

Let's *resolve* this situation

- DNS-based load-balancing

  - often relies on up-to-date geo datapacks

  - uses on the user's *resolver's* address unless you're handling EDNS0

- Querying DNS from problematic networks can reveal ISP resolver behaviour
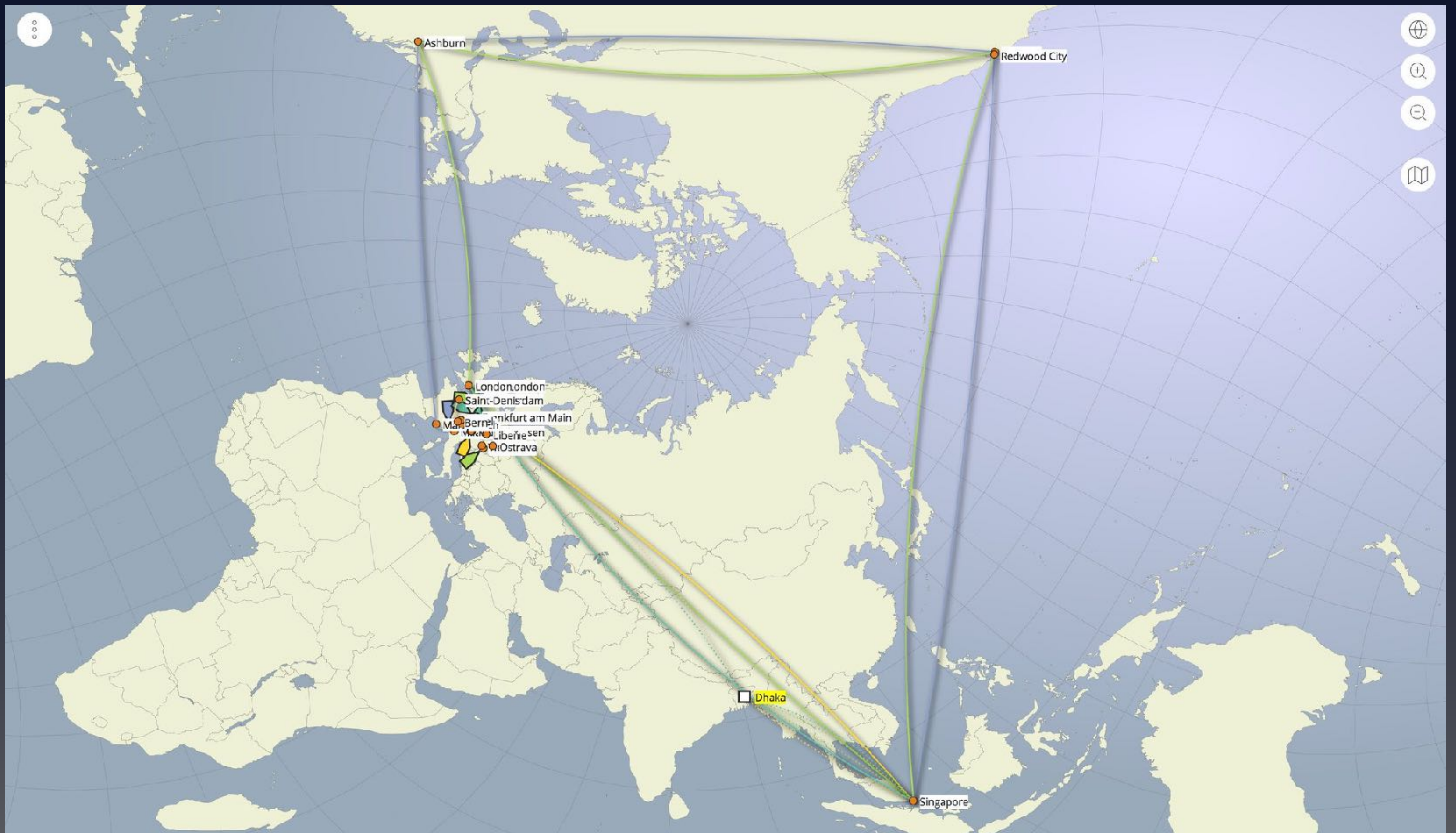
TRACEROUTE

Network Paths

- Routing is:

  - not totally within your control

  - often asymmetric

  - difficult to debug without actual path traces between specific locations

- Anycast to multiple sites makes this especially tricky

Network Paths

- traceroutes can tell you a bunch of things:

  - which networks user traffic is passing through

  - whether there's congestion/loss anywhere on those paths

  - whether things are consistently `accept()`ing connections on the other side

- traceroutes from many locations give you this in aggregate

- Can you route away from a problematic path?

Monitoring?

- Given sufficient vantage points, you can model load balancing around configuration changes

- Folks write custom tooling to understand where users are routed on their deployment

- The NCC monitors the k-root with Atlas

- Verizon/Edgecast has been very open with their Atlas usage:
  https://labs.ripe.net/Members/verizon_digital/seeing-the-world-with-ripe-atlas

Atlas for monitoring?

Because we mostly expose *data*,
folks are free to write hooks for
nagios, icinga, prometheus, etc

Atlas for monitoring?

We expose measurement results a few ways:

streaming API

ping: status-checks

results API

some tooling: ipmap, traceMON, DNSMON

Most of the data is public!

```python
1 from ripe.atlas.cousteau import AtlasStream
2 import json
3
4 def on_result_response(*args):
5     print json.dumps(args[0], indent=2)
6
7 atlas_stream = AtlasStream()
8 atlas_stream.connect()
9
10 channel = "atlas_result"
11 atlas_stream.bind_channel(channel, on_result_response)
12
13 stream_parameters = {"destinationPrefix": "2001:4860::/32", "type": "ping"}
14 atlas_stream.start_stream(stream_type="result", **stream_parameters)
15
16 atlas_stream.timeout()
17
18 atlas_stream.disconnect()
```

Finally,

We rely on community support and feedback

https://atlas.ripe.net/get-involved/
https://github.com/RIPE-Atlas-Community/ripe-atlas-community-contrib

Questions/Comments

Please talk to us if you're interested in
using any of these tools/data!

https://atlas.ripe.net/

sds@ripe.net
@sdstrowes