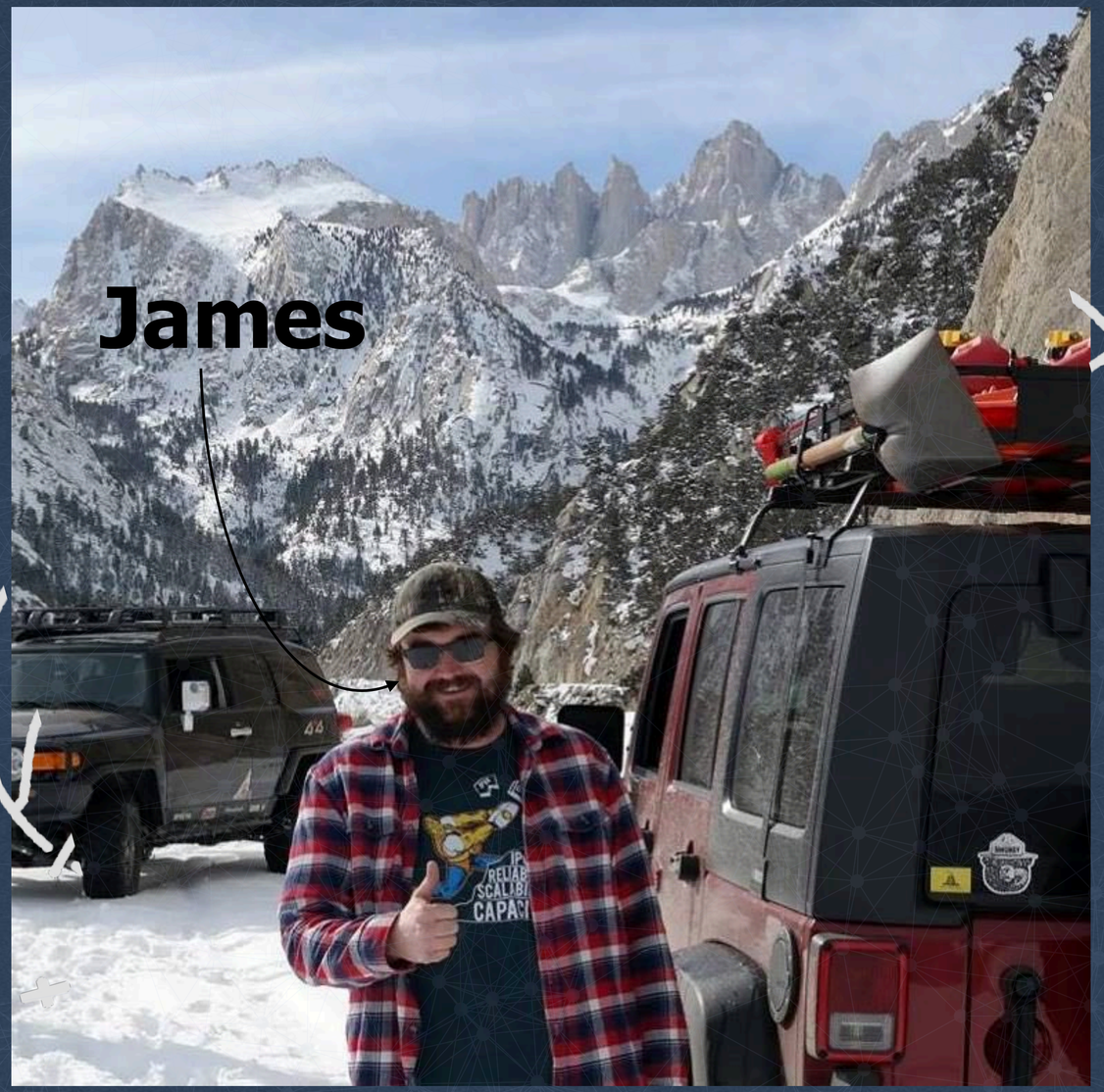
Open NetNorad

Jose Leitao & Daniel Rodriguez | NIE | Dublin



facebook scale

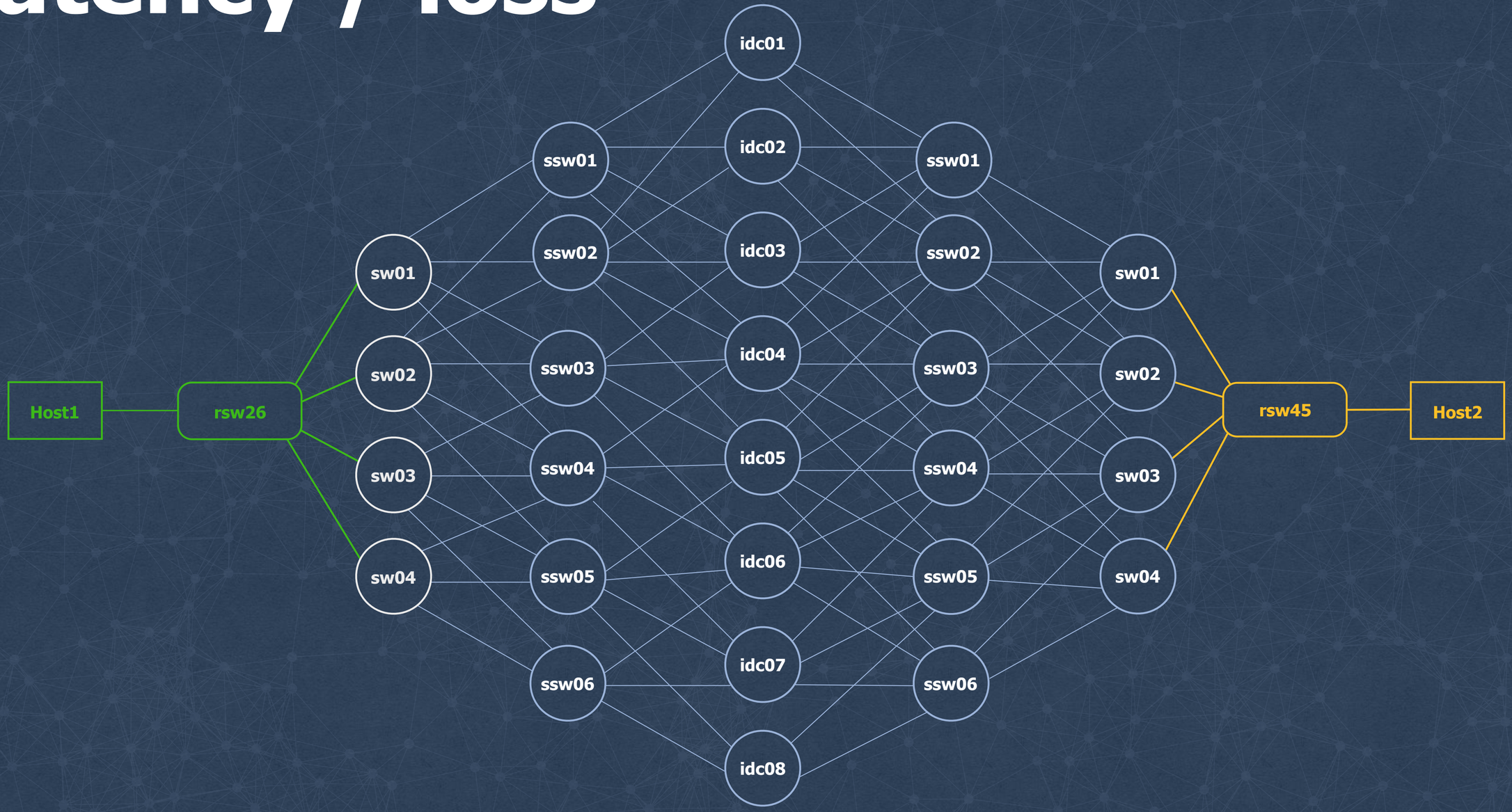
as of June 2017

1.32 billion daily
active users

2.01 billion monthly
active users

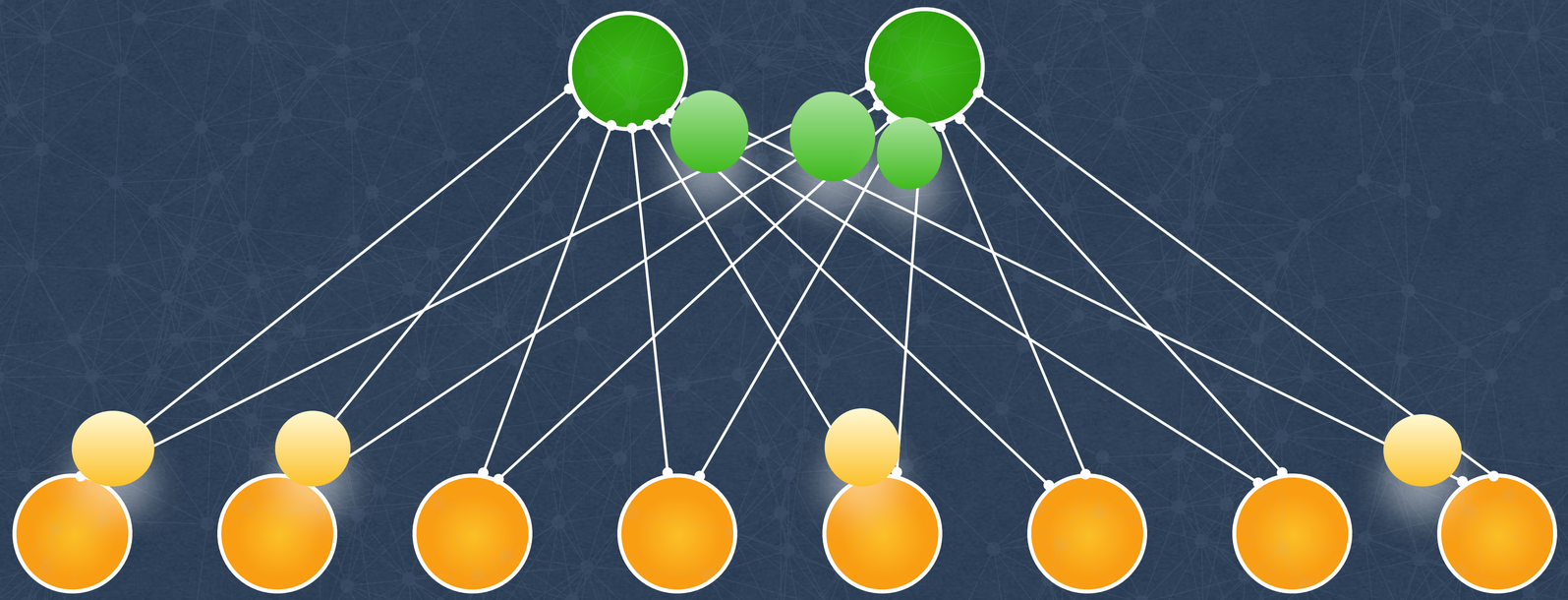
Loss on the net

Latency / loss



NetNORAD

Ping all the things!



Run pingers on some machines

Run responders on all machines

Collect and analyze data

Evolution

Run /bin/ping from a python agent

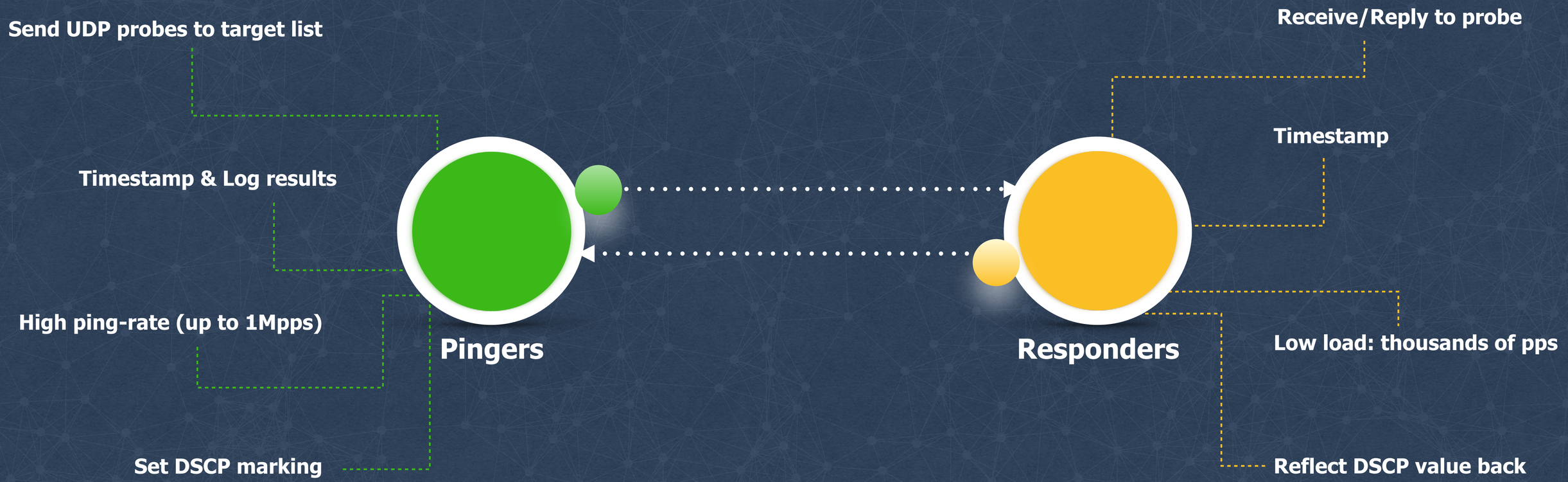
Raw Sockets, Fast TCP Probes

Raw Sockets, Fast ICMP Probes

UDP Probes and Responder

UDP Probes and Responder + Fast ICMP Probes

Pinger and responder



github.com/facebook/UdpPinger

Why UDP?

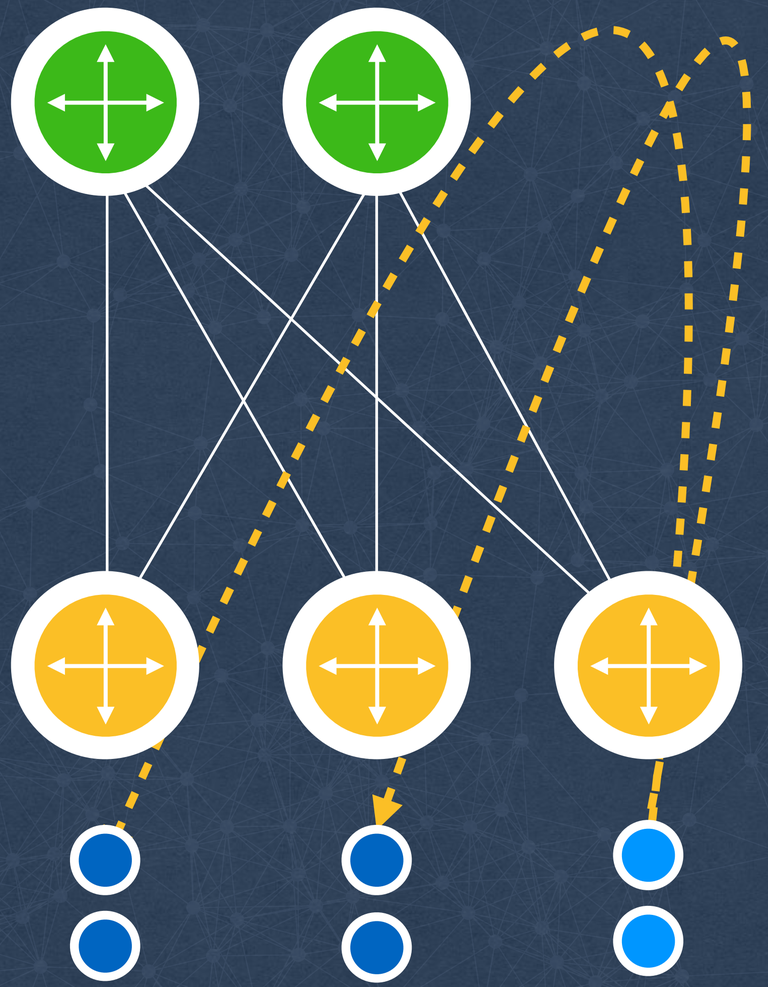
- No TCP RST packets
- Efficient ECMP coverage
- Extensible

Why UDP?

Probe structure

```
Signature

Send Time

Receive Time

Response Time

Traffic Class
```

Pinging inside clusters



Detect issues with rack switches

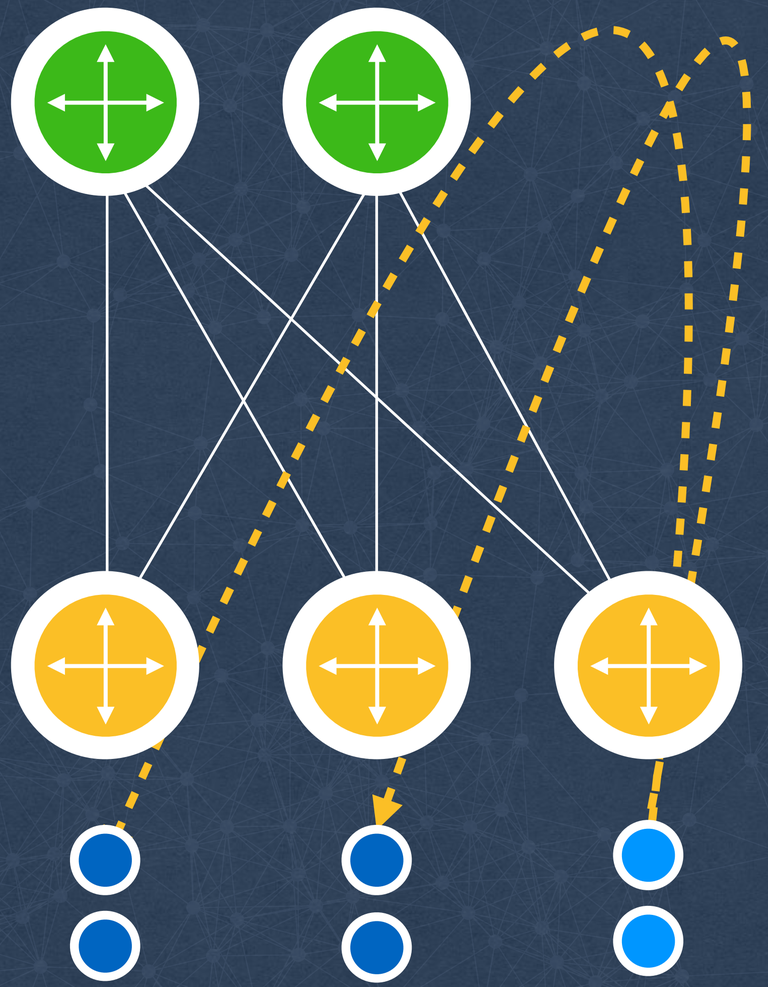
Dedicated pingers per cluster

Probe ALL machines in cluster

Store time-series per host/rack

Lags real-time by 2 minutes

Pinging inside clusters



Detect issues with rack switches

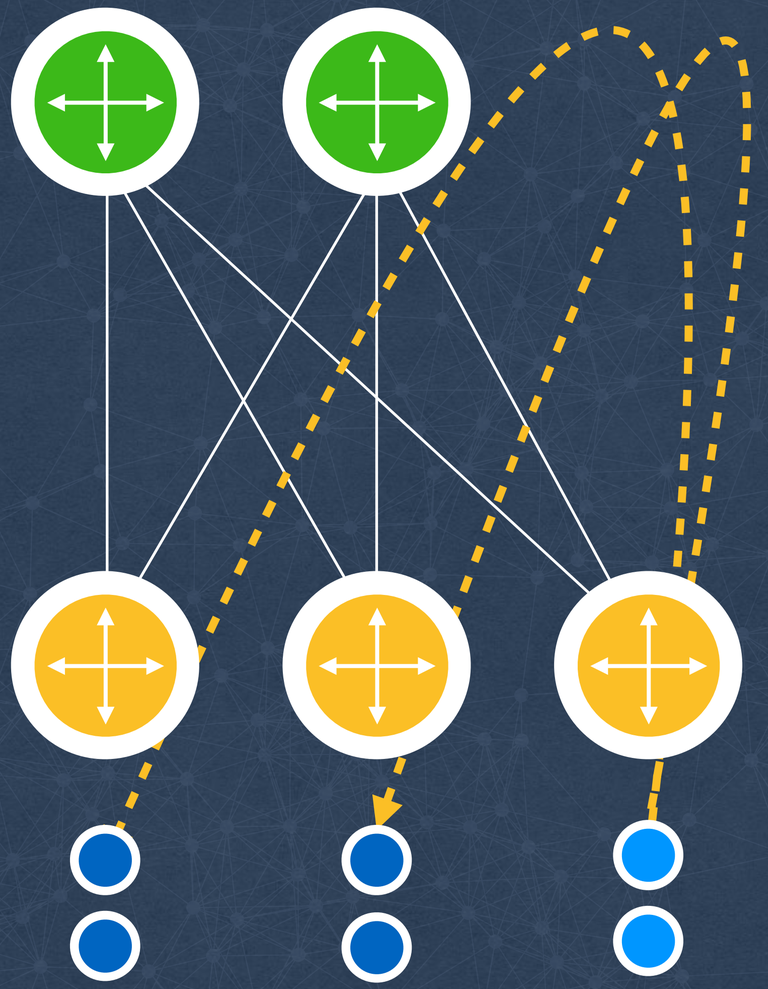
Dedicated pingers per cluster

Probe ALL machines in cluster

Store time-series per host/rack

Lags real-time by 2 minutes

Pinging inside clusters



Detect issues with rack switches

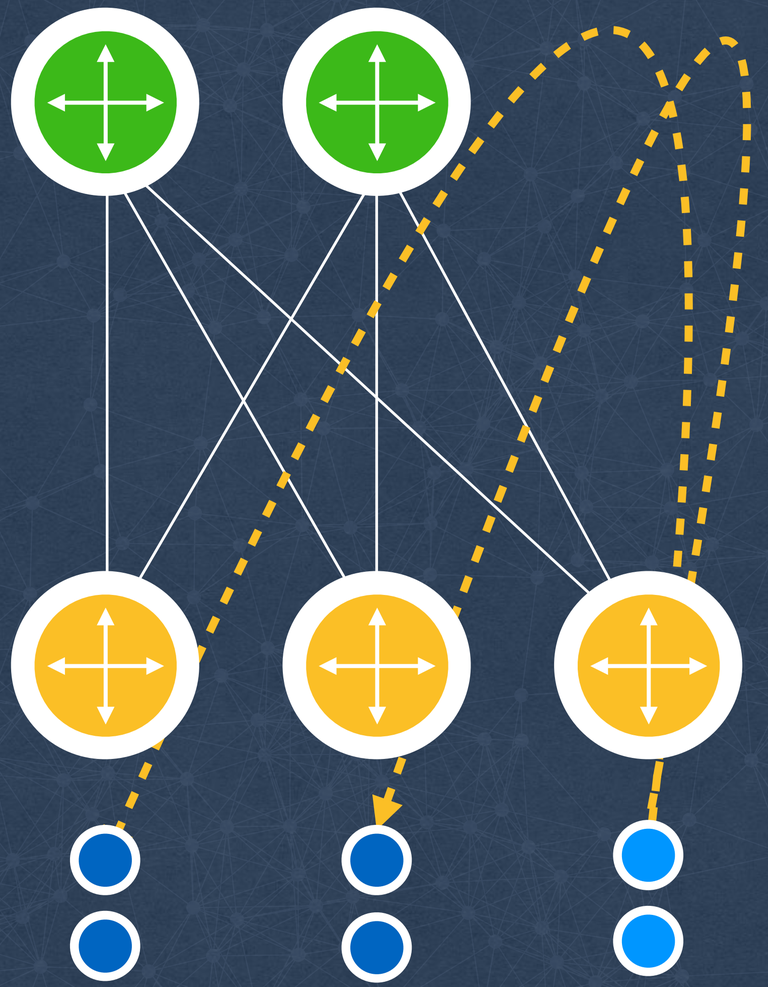
Dedicated pingers per cluster

Probe ALL machines in cluster

Store time-series per host/rack

Lags real-time by 2 minutes

Pinging inside clusters



Detect issues with rack switches

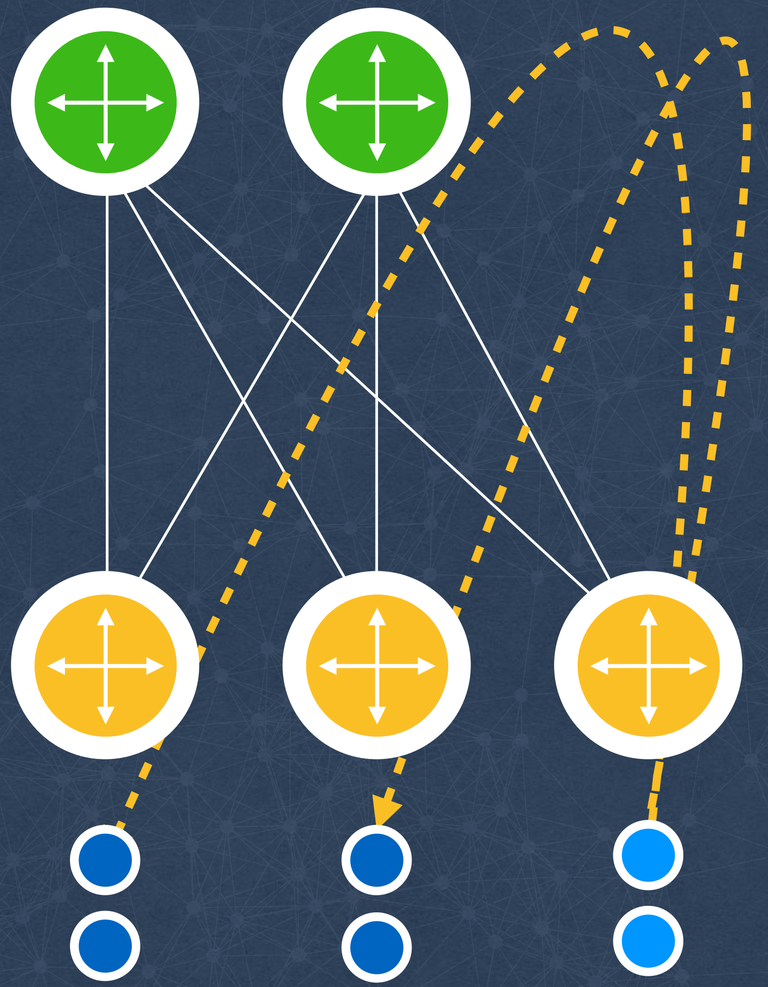
Dedicated pingers per cluster

Probe ALL machines in cluster

Store time-series per host/rack

Lags real-time by 2 minutes

Pinging inside clusters



Detect issues with rack switches

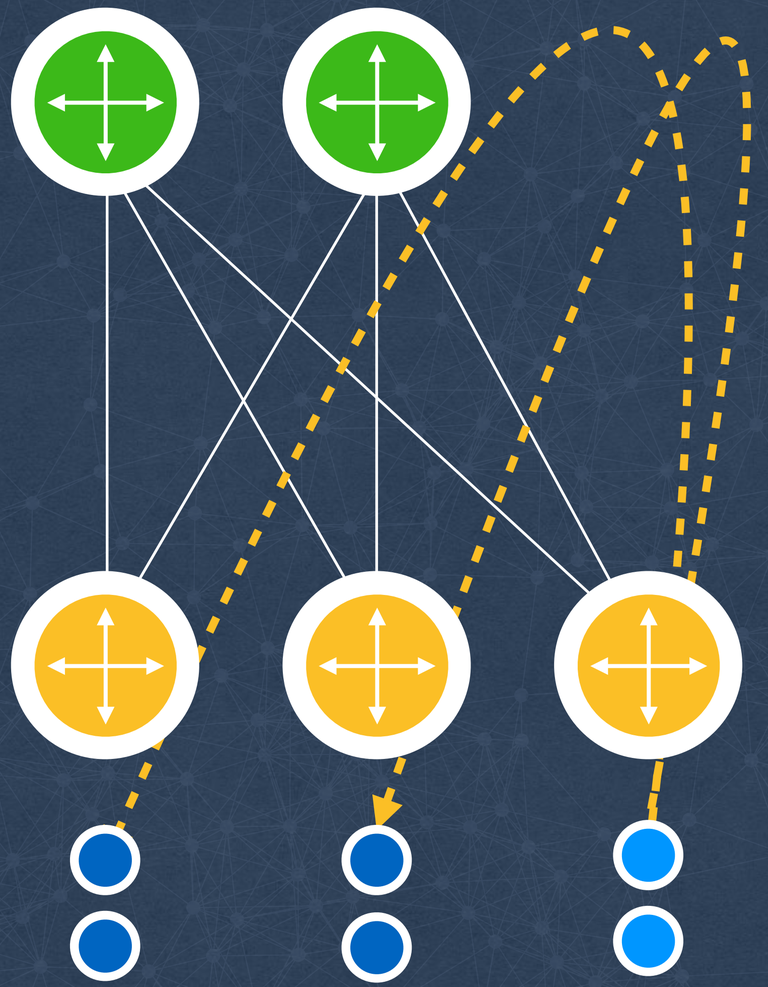
Dedicated pingers per cluster

Probe ALL machines in cluster

Store time-series per host/rack

Lags real-time by 2 minutes

Pinging inside clusters



Detect issues with rack switches

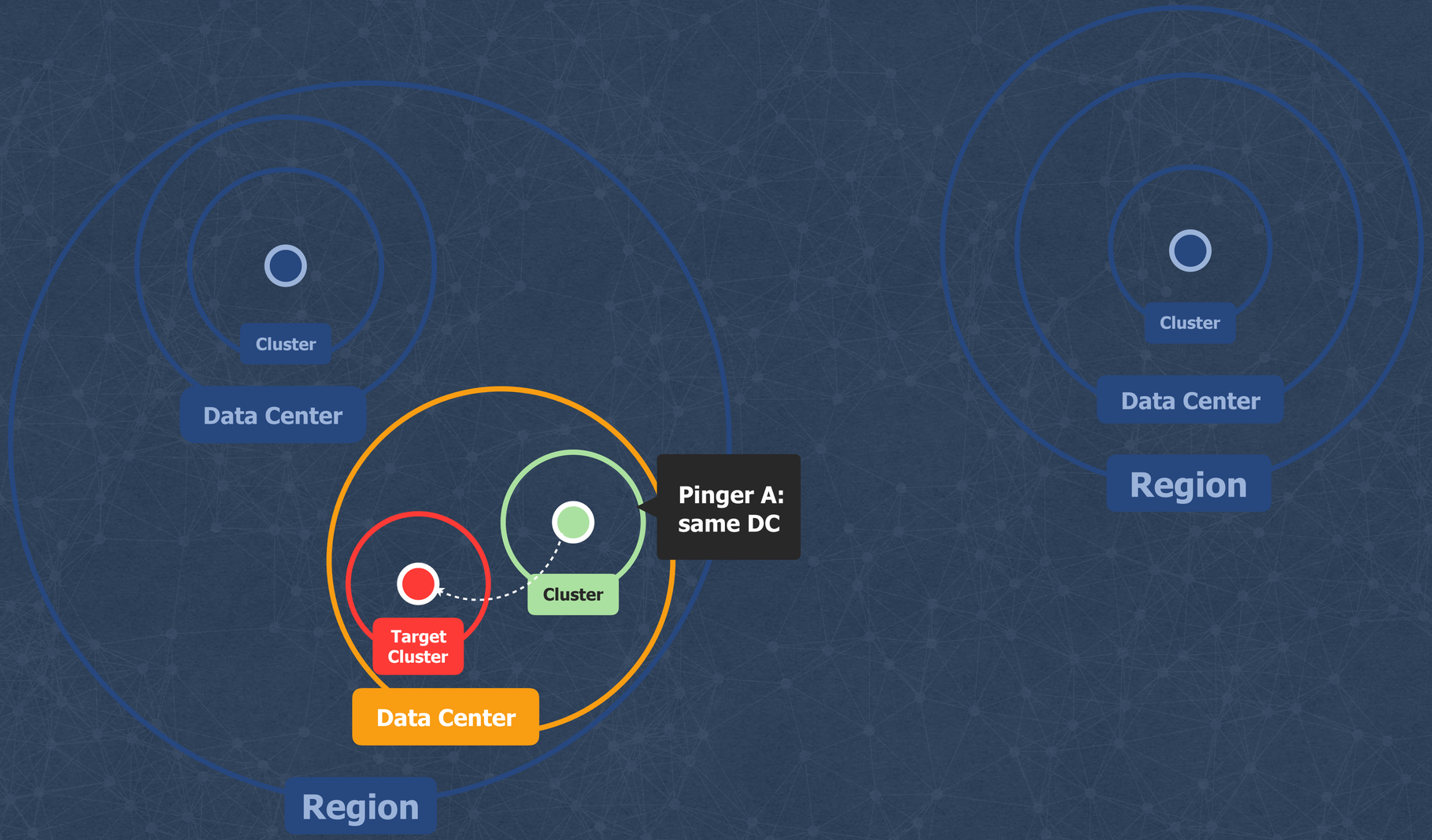
Dedicated pingers per cluster

Probe ALL machines in cluster

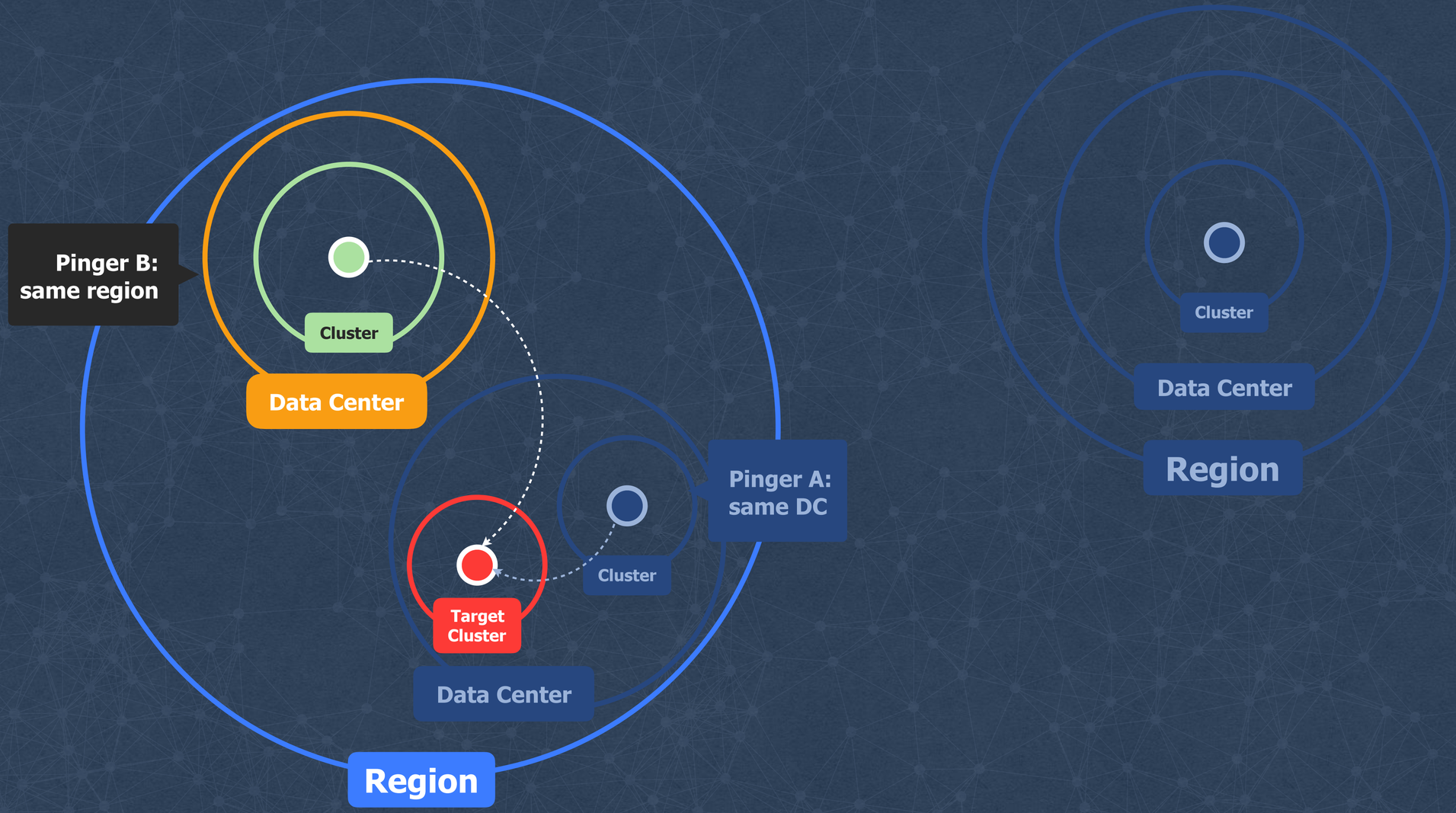
Store time-series per host/rack

Lags real-time by 2 minutes

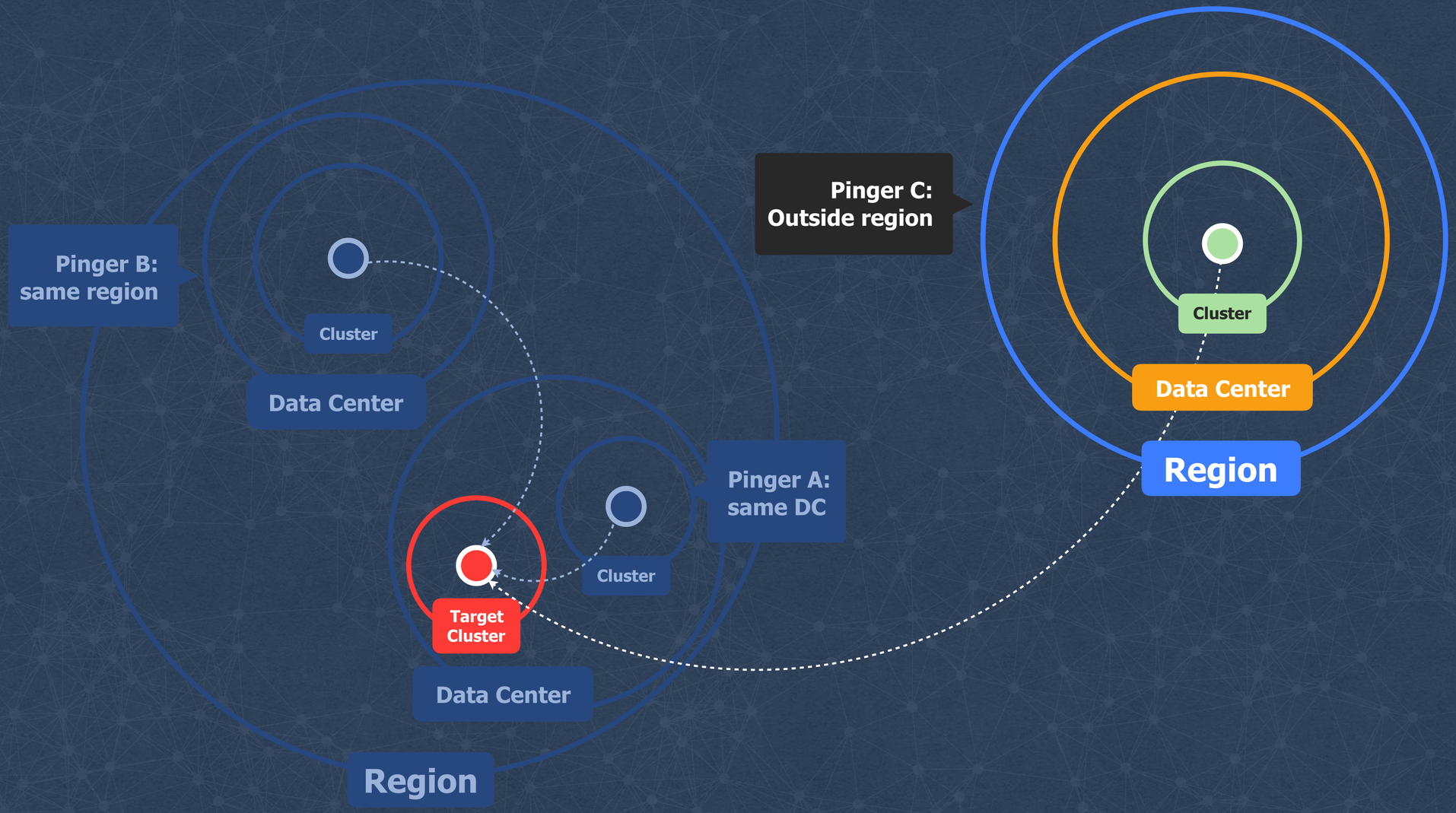
Pinging the clusters



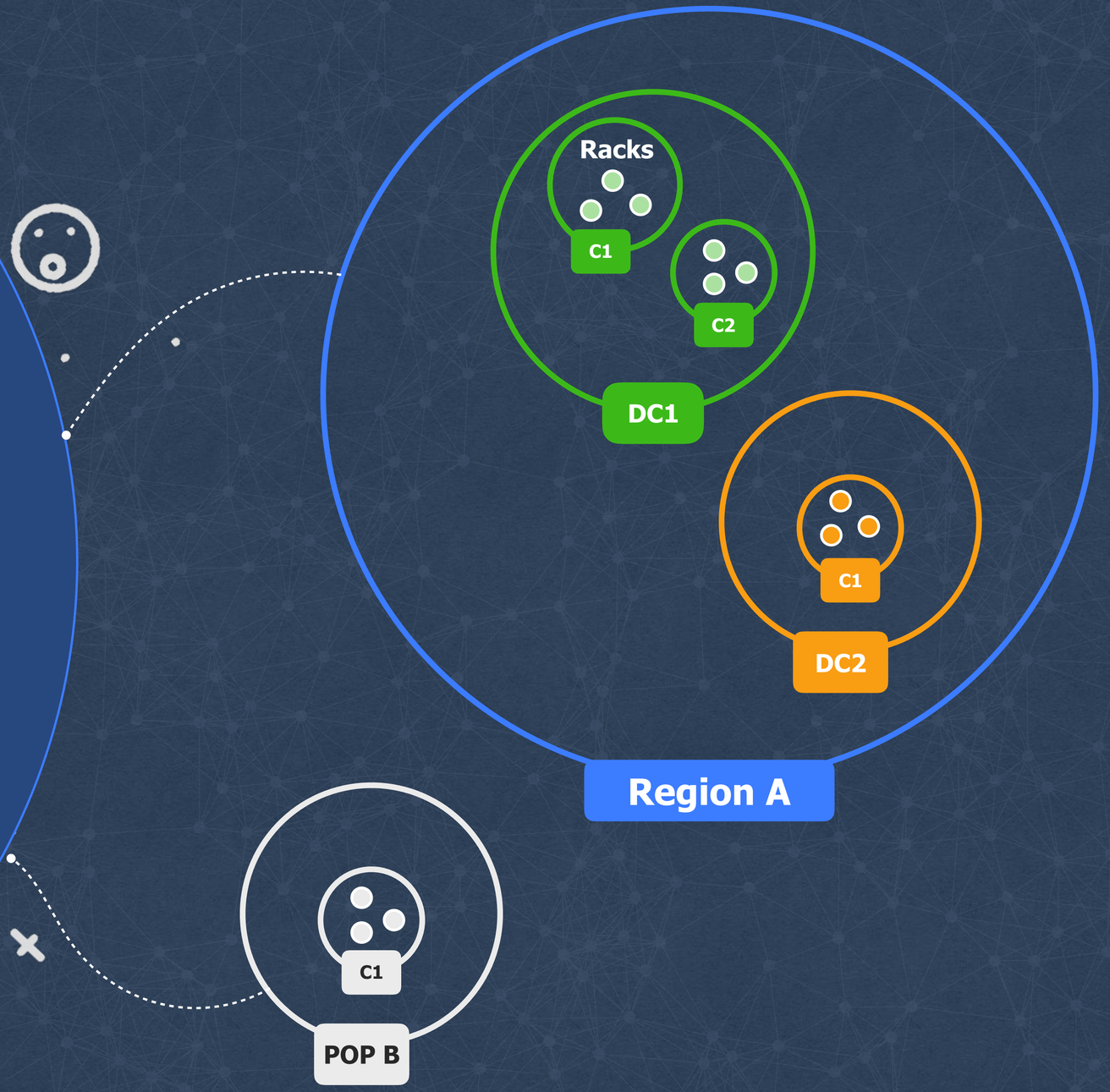
Pinging the clusters



Pinging the clusters



Backbone



Building an Open Source version

github.com/facebook/UdpPinger

github.com/fbsamples/OpenNetNorad

fbsamples / OpenNetNorad

Watch 4 | Star 0 | Fork 0

Code | Issues 0 | Pull requests 0 | Projects 0 | Insights

Sample system to manage Uping and Upong (UdpPinger) instances, used to measure / graph network latency and loss on Linux

2 commits | 1 branch | 0 releases | 1 contributor

Branch: master | New pull request | Find file | Clone or download

j-leitao committed on GitHub Update README.md — Latest commit dd2a32e 15 days ago

chronograf	first commit	15 days ago
debian	first commit	15 days ago
pong_logger	first commit	15 days ago
scripts	first commit	15 days ago
CONTRIBUTING.md	first commit	15 days ago
LICENSE.md	first commit	15 days ago
PATENTS.md	first commit	15 days ago
README.md	Update README.md	15 days ago

fbsamples / OpenNetNorad

Watch 5 | Star 1 | Fork 0

Code | Issues 0 | Pull requests 0 | Projects 0 | Insights

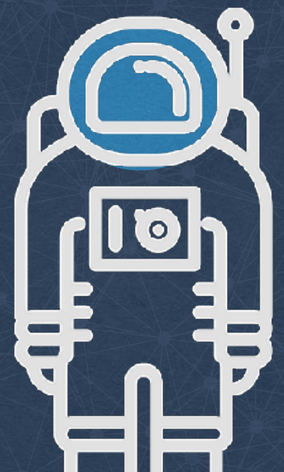
Branch: master | OpenNetNorad / debian / | Create new file | Find file | History

j-leitao first commit — Latest commit 54306fe 17 days ago

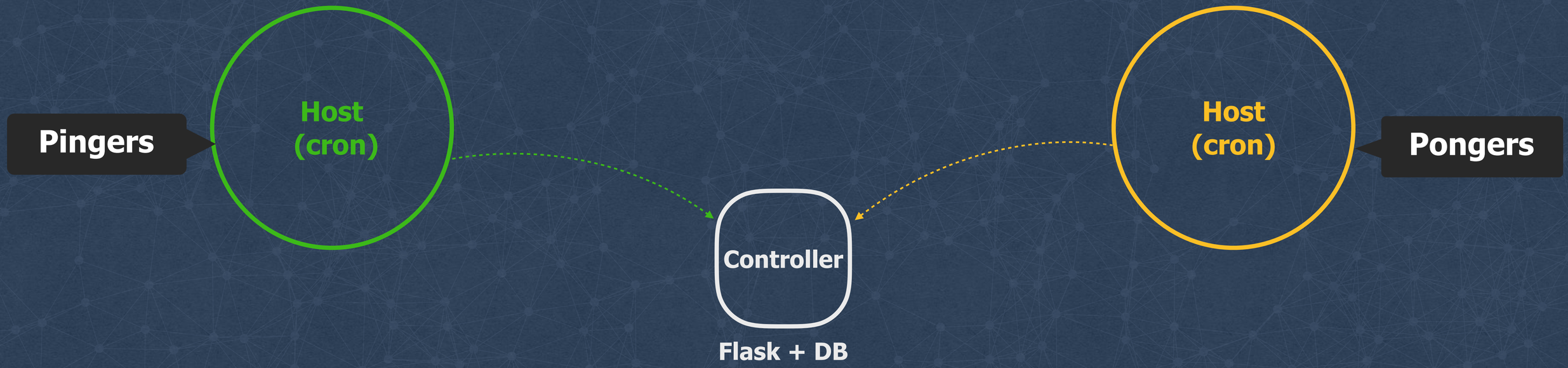
..

libfolly-dev_57.0-1_amd64.deb	first commit	17 days ago
libfolly57.0_57.0-1_amd64.deb	first commit	17 days ago
thrift_1-1_amd64.deb	first commit	17 days ago
udppinger_1-1_amd64.deb	first commit	17 days ago

Our solution



Our solution




```
#
*/1 * * * * root /etc/OpenNetNorad/report_upongd_systemd.sh
*/1 * * * * root /etc/OpenNetNorad/udppinger_collect_telegraf.sh -m 10.142.0.4 -l 10.142.0.4 -s 10.132.0.5
 -c "MAD01CL01" -r "MAD01CL01R01" -e "MAD01"
```

Branch: master ▾ **OpenNetNorad** / **scripts** / **report_upongd_systemd.sh** Find file | Copy path

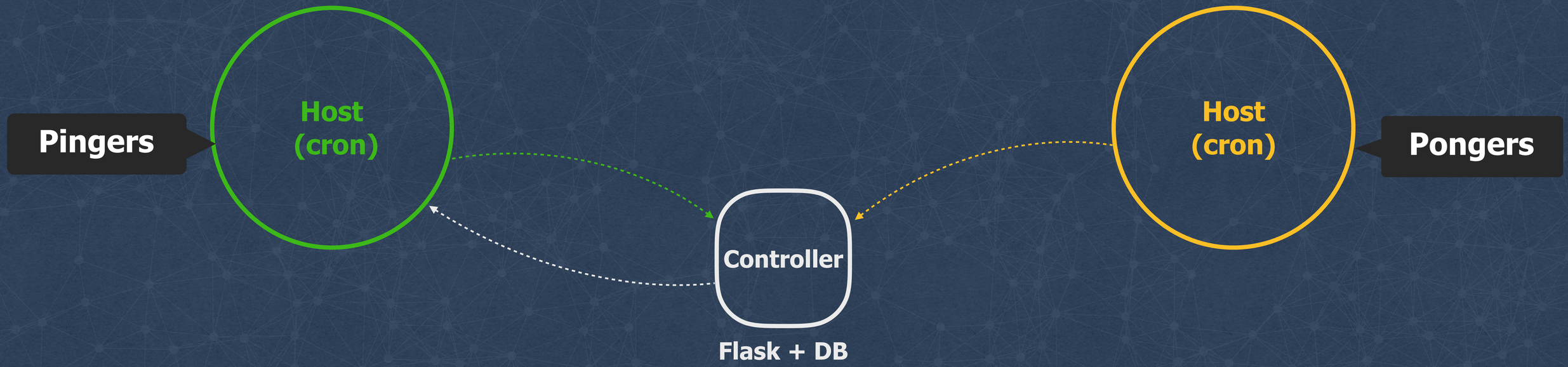
j-leitao first commit 54306fe 17 days ago

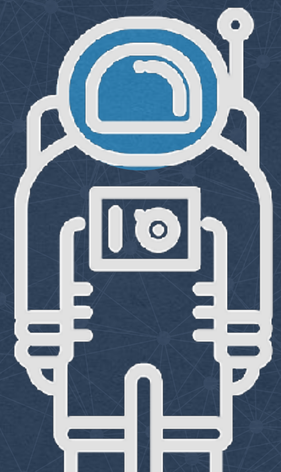
1 contributor

Executable File | 22 lines (16 sloc) | 715 Bytes Raw | Blame | History

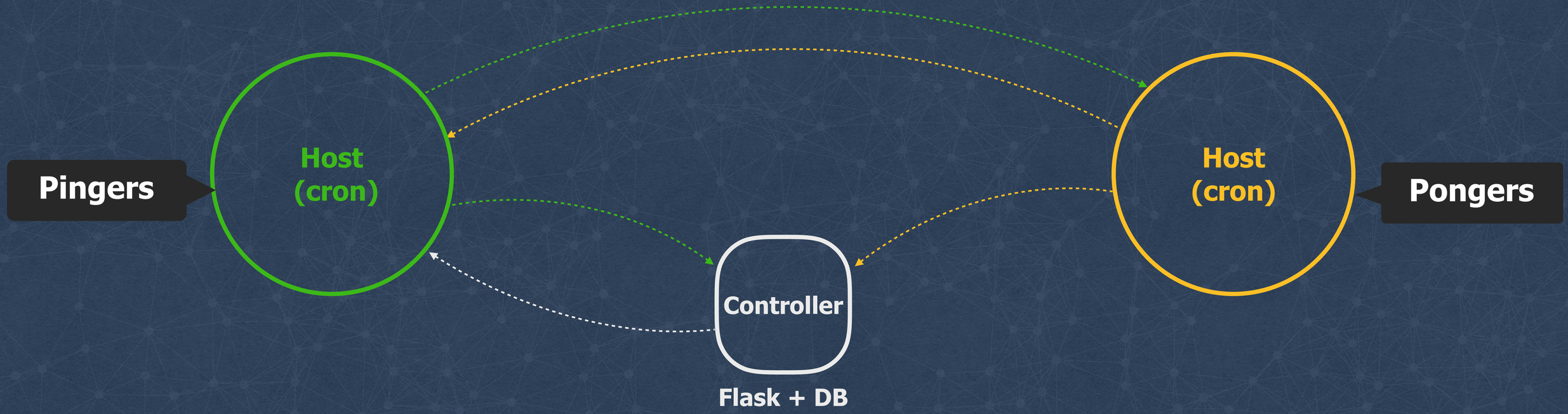
```
#!/bin/sh

# Copyright (c) 2017-present, Facebook, Inc.
# All rights reserved.

# This source code is licensed under the BSD-style license found in the
# LICENSE file in the root directory of this source tree. An additional grant
# of patent rights can be found in the PATENTS file in the same directory.

# Log server IP address
LOG_SERVER="192.168.1.125"

# Controller pull cmds
REGISTER_RUNNING="/usr/bin/wget -O - --post-data=is_active=1 http://$LOG_SERVER:5000/servers/update"
REGISTER_STOPPED="/usr/bin/wget -O - --post-data=is_active=0 http://$LOG_SERVER:5000/servers/update"

if [ -n "systemctl is-active sshd >/dev/null 2>&1" ]; then
    REG_RES=`$REGISTER_RUNNING`
else
    REG_RES=`$REGISTER_STOPPED`
fi
```

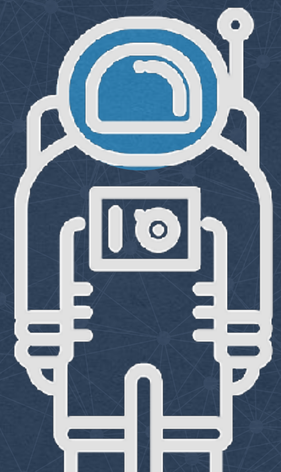
Our solution



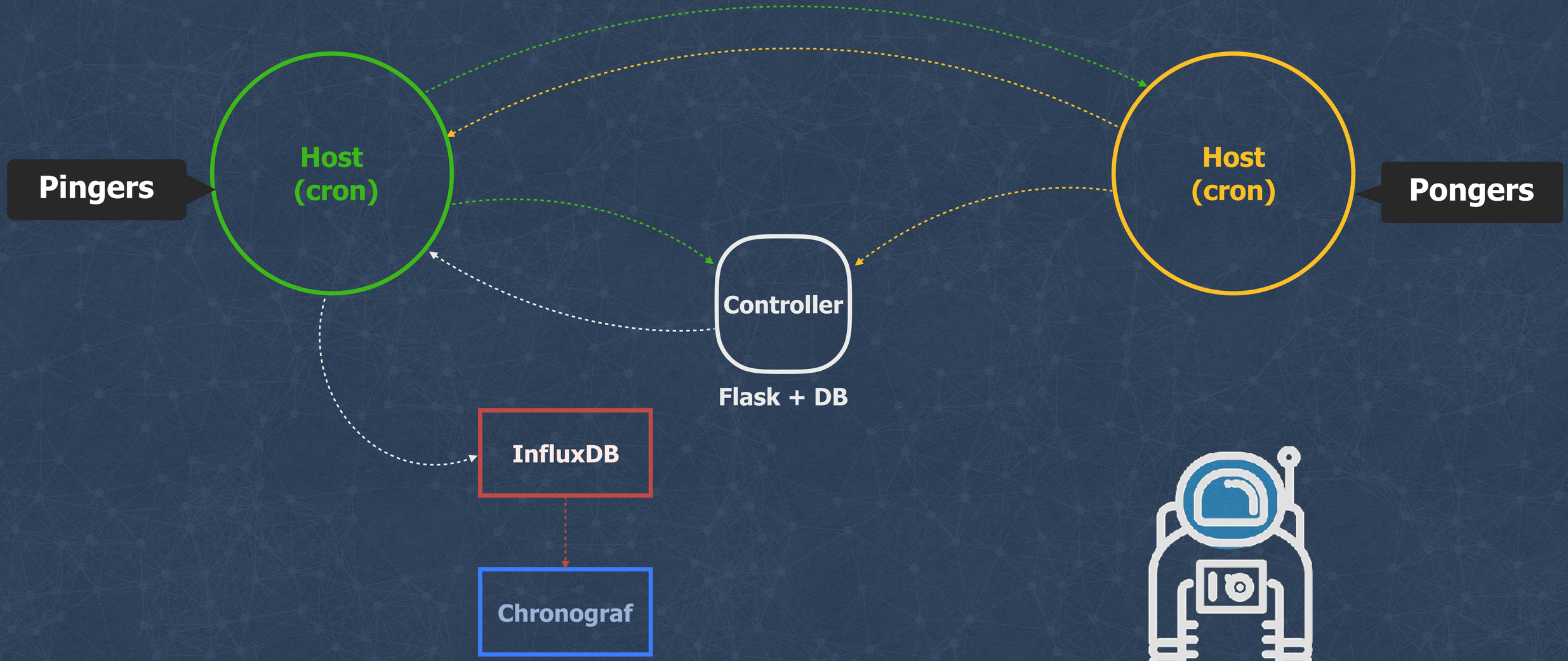


Our solution

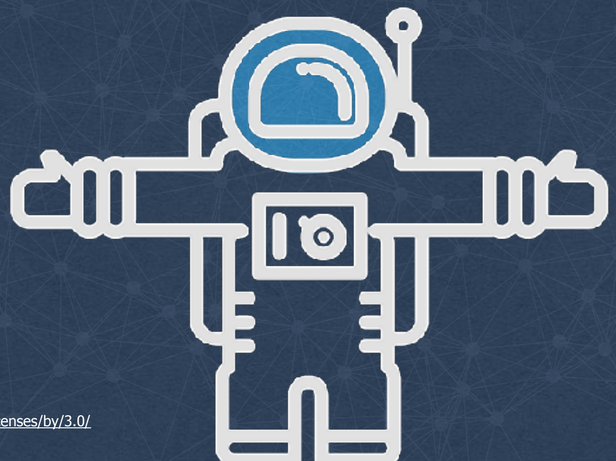




Our solution

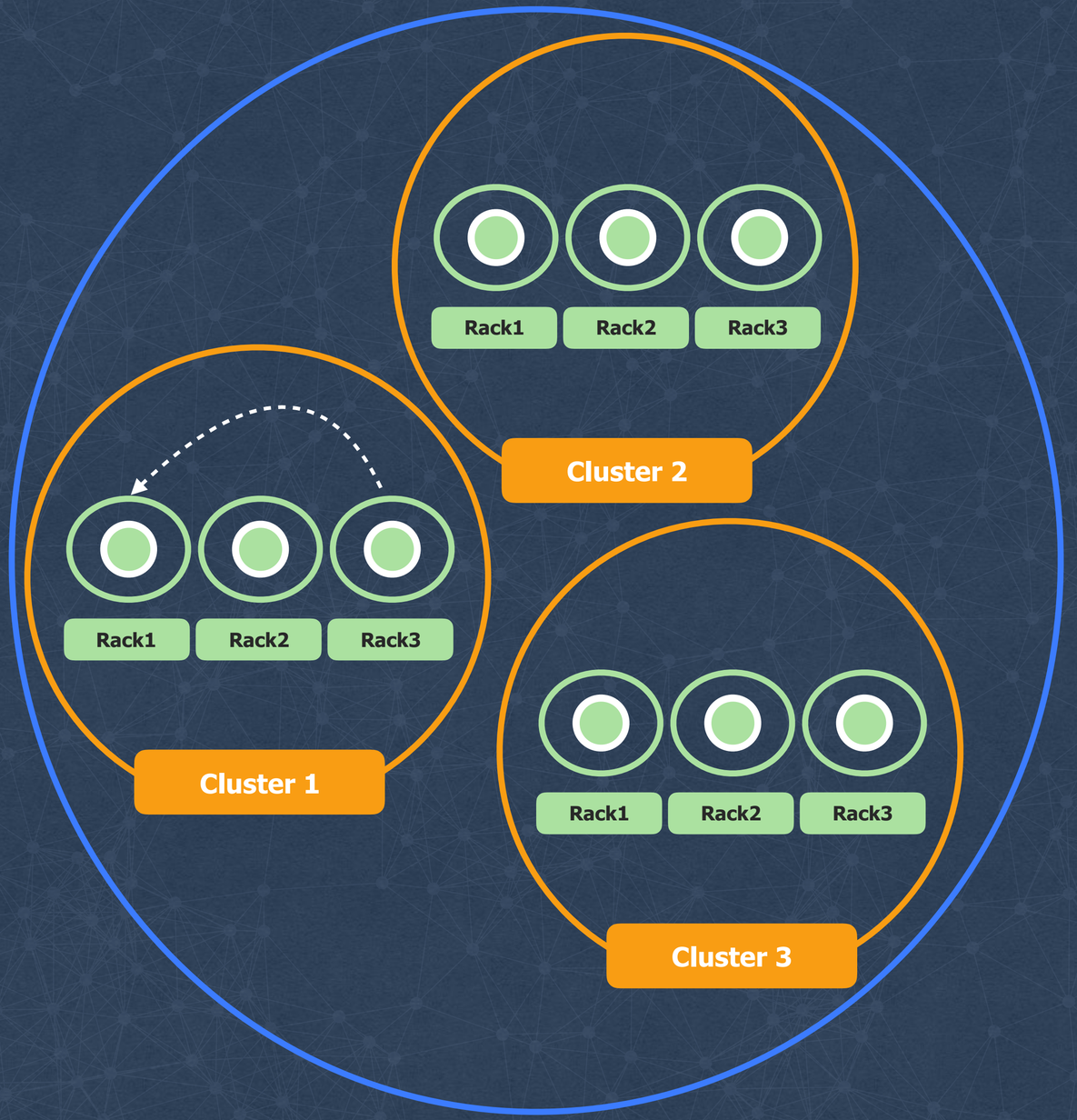


Demo



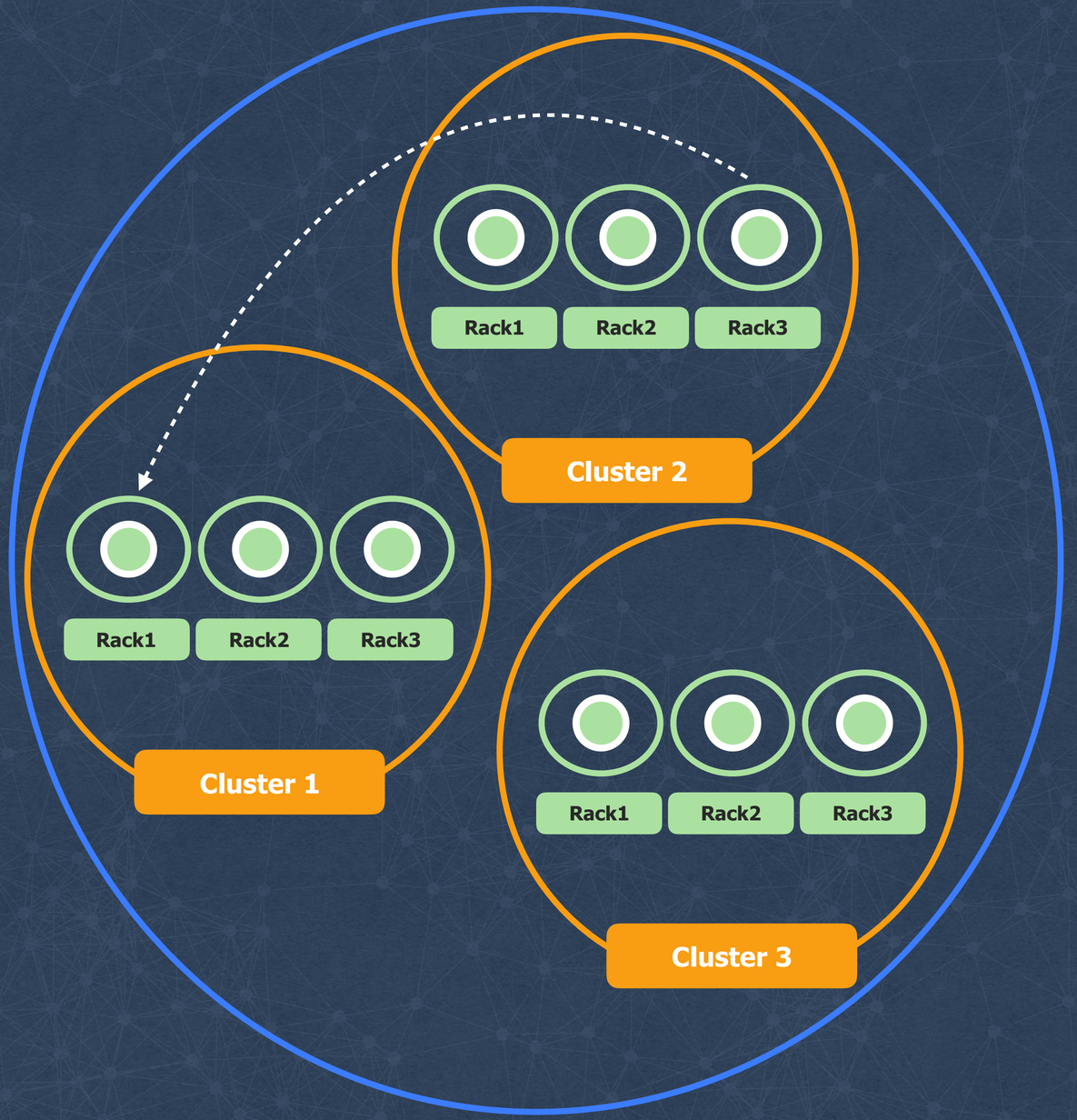
Demo

Loss within the same cluster

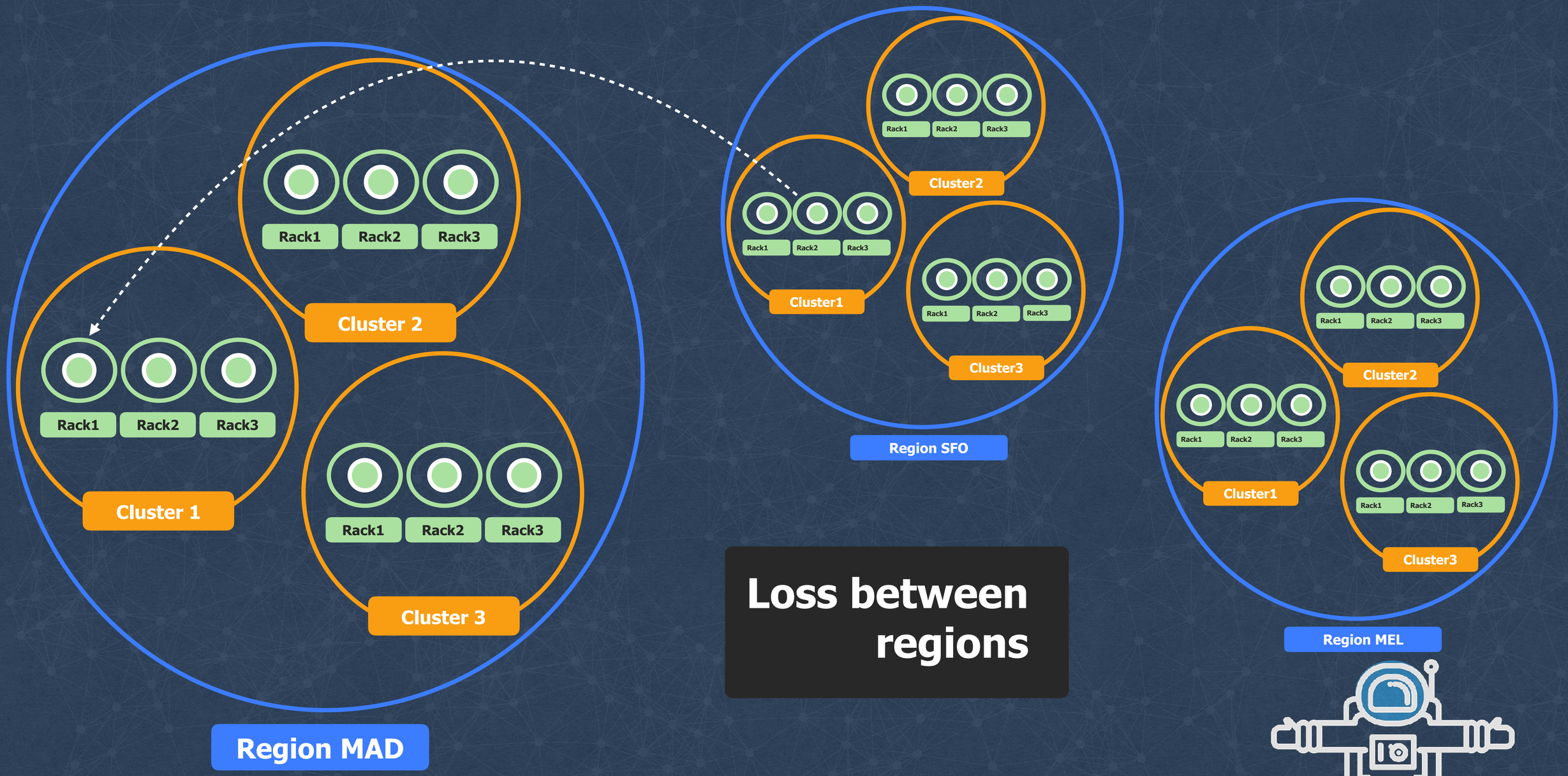


Demo

Loss between clusters



Demo



Loss between regions

Q&A