Scaling networks through software

March 16th 2015 | João Taveira Araújo
@jta

network systems @ fastly


GitHub

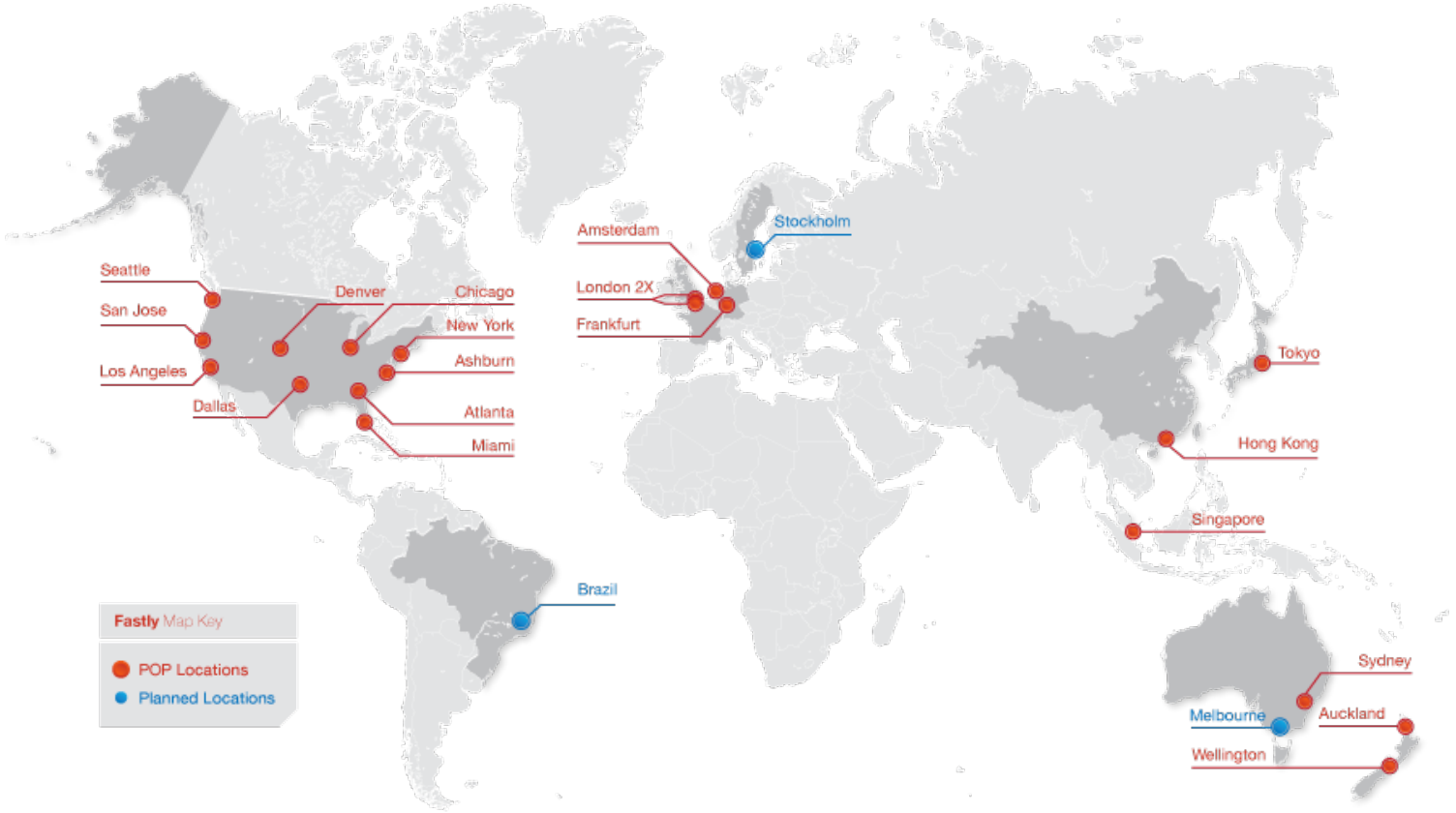

python
SOFTWARE
FOUNDATION


debian
GNU/Linux

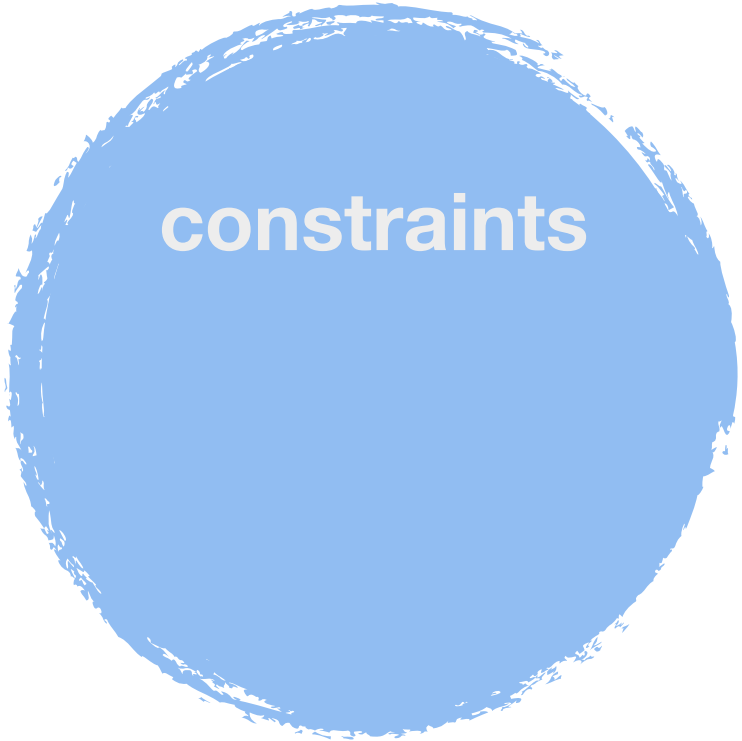

meta::cpan


imgur

FASTLY GLOBAL CONTENT DELIVERY NETWORK



scalability

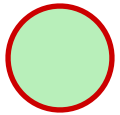


constraints

time

money

people

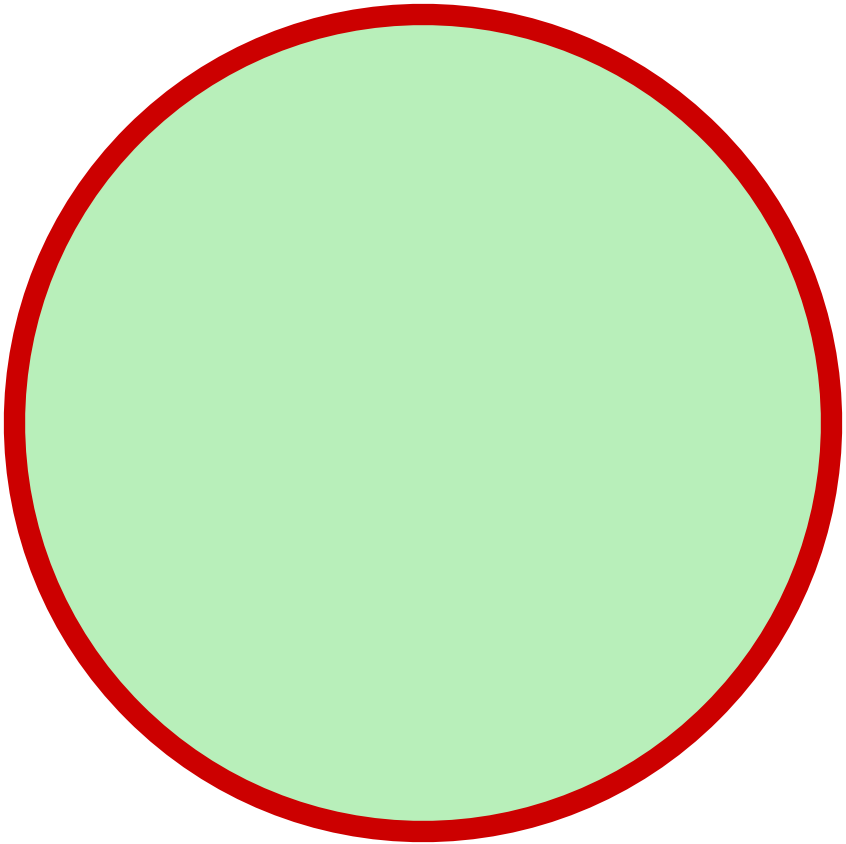


constraints

time

money

people



constraints
time
money
people

Becoming a multi terabit network

Number of PoPs ..... ~20

BGP announcements ..... ~2000

Requests per second ..... ~1000000

Becoming a multi terabit network

Number of PoPs ..... ~20

BGP announcements ..... ~2000

Requests per second ..... ~1000000

Network ops 2

Becoming a multi terabit network

Number of PoPs ..... ~20

BGP announcements ..... ~2000

Requests per second ..... ~1000000

Network ops 2

Network software ..... me

scalability

observations on network

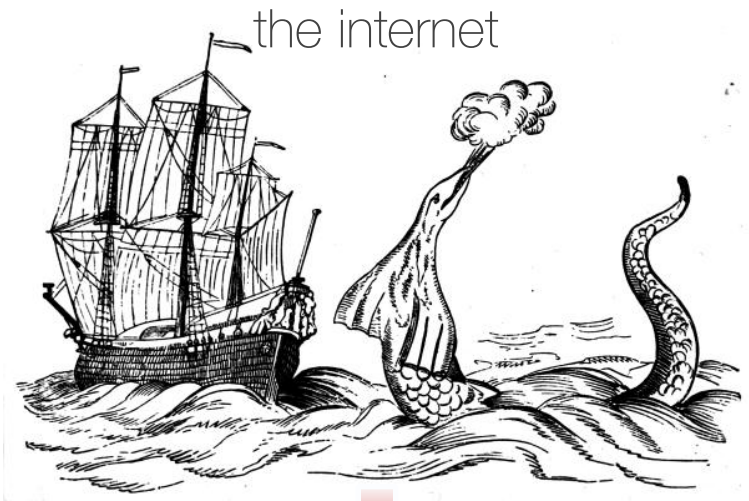
scalability

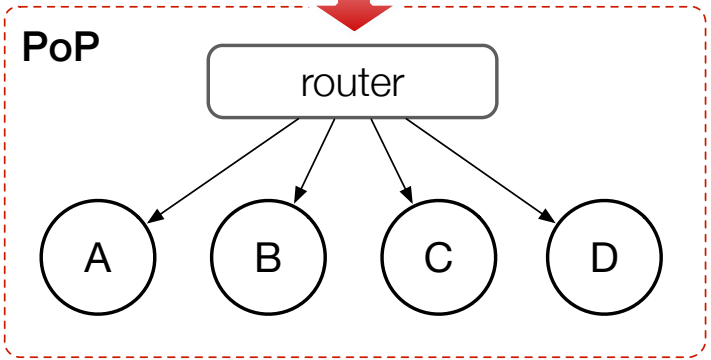
from a company that used to be a startup

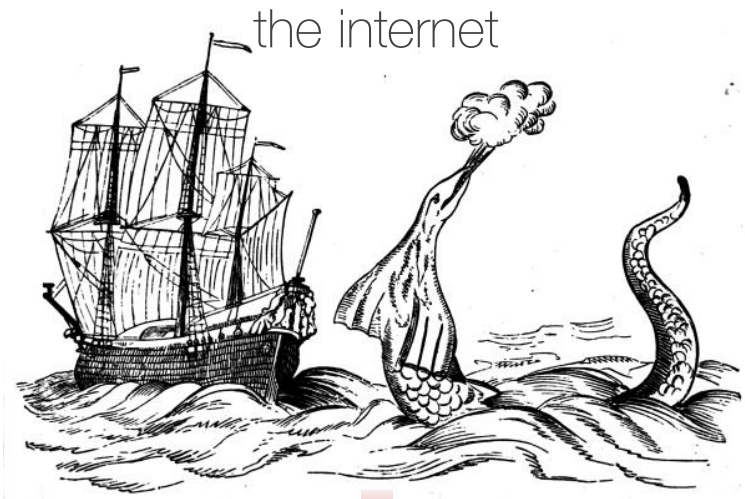
anything you don't explicitly control is an implicit liability

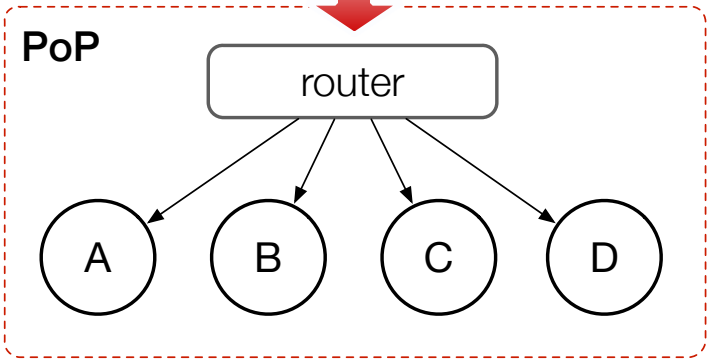
fastly

the internet



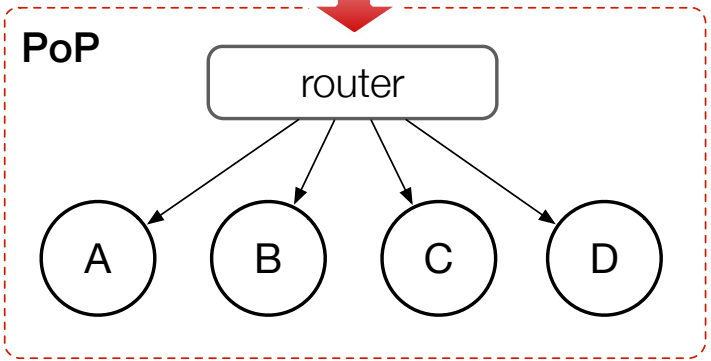


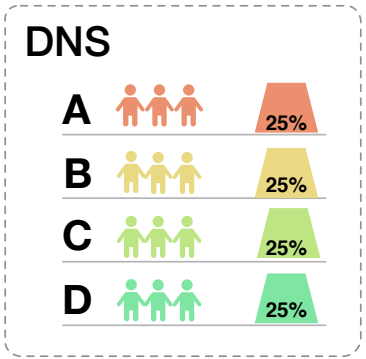


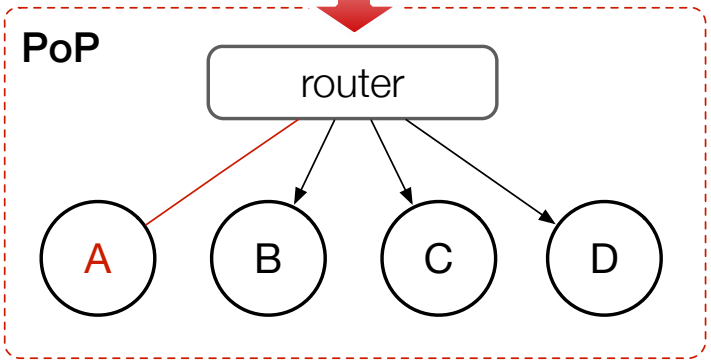


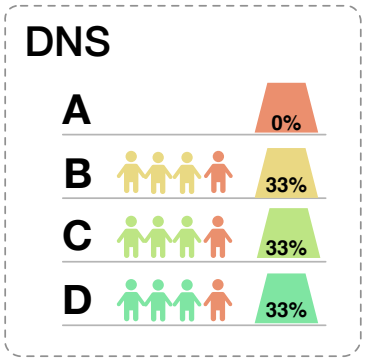
How do you:

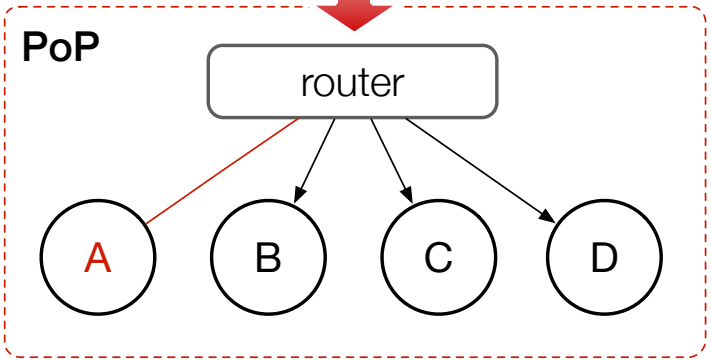
- load balance traffic
- gracefully failover if a server fails





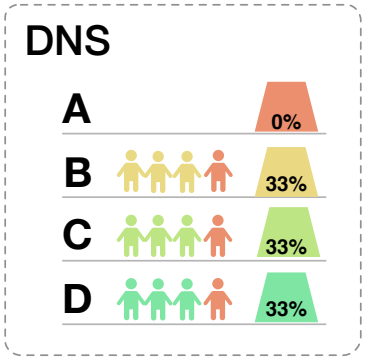


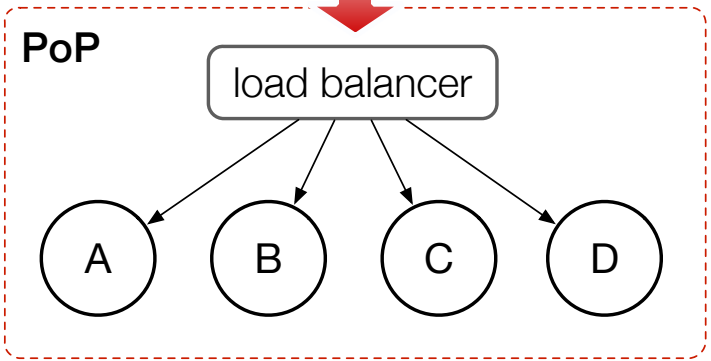


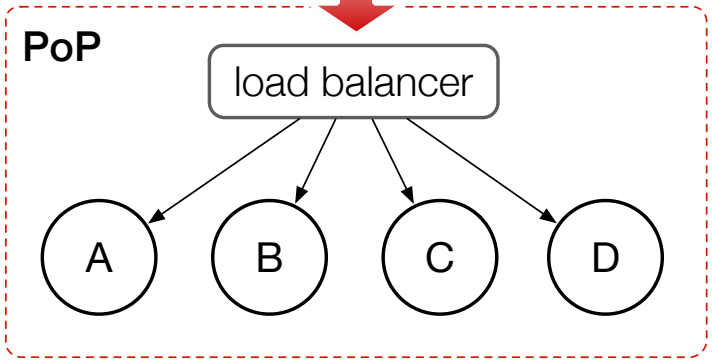


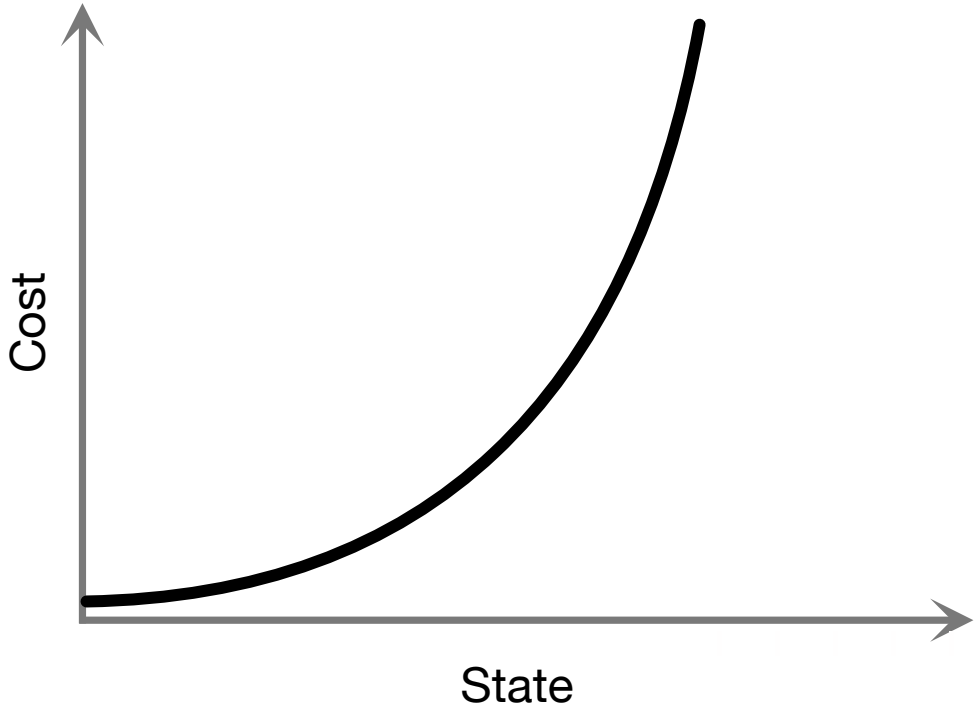
Bad idea:

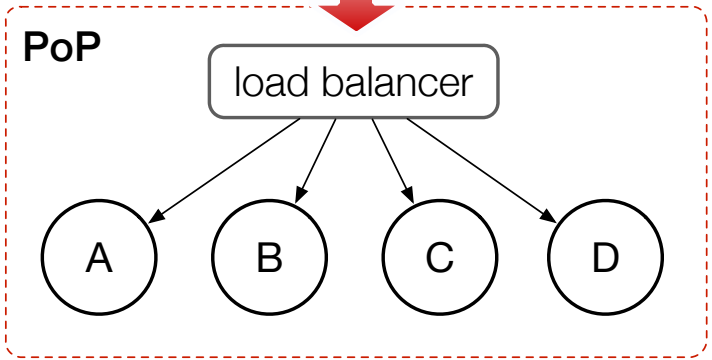
- gets hard to manage
- do one thing and do it well
- you don't control TTL

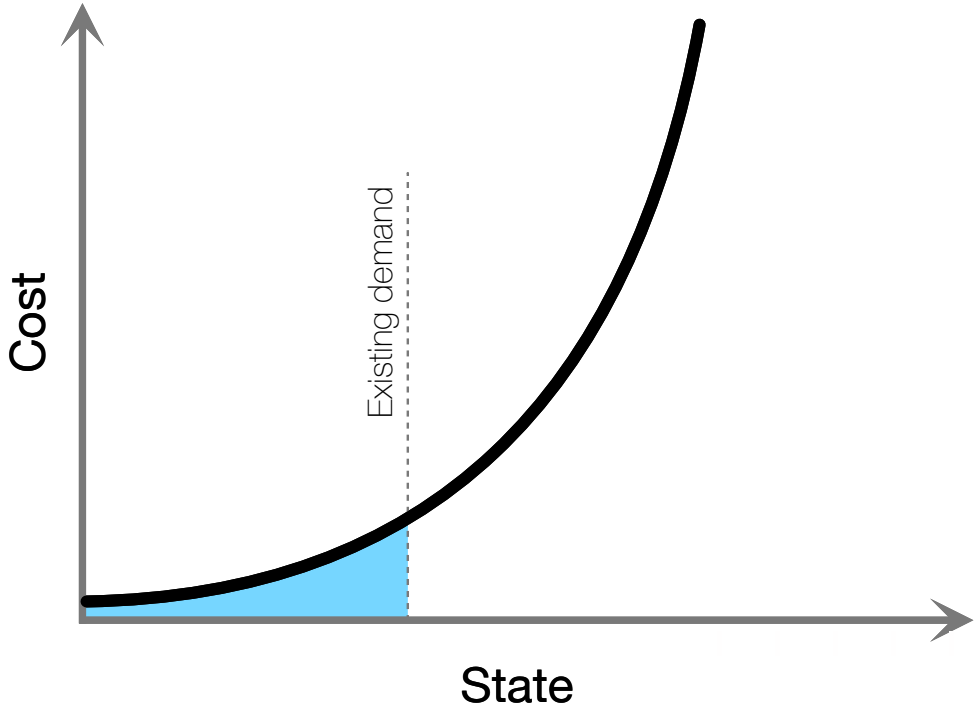


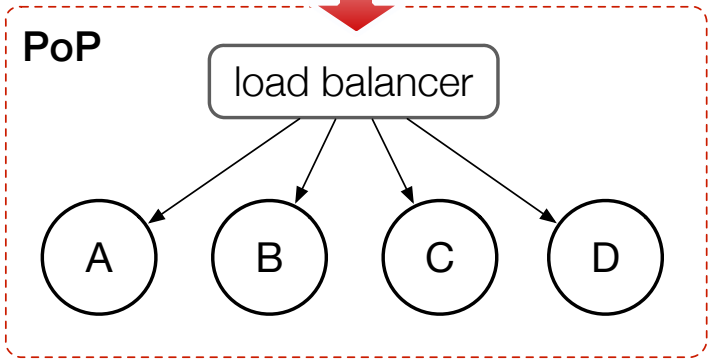


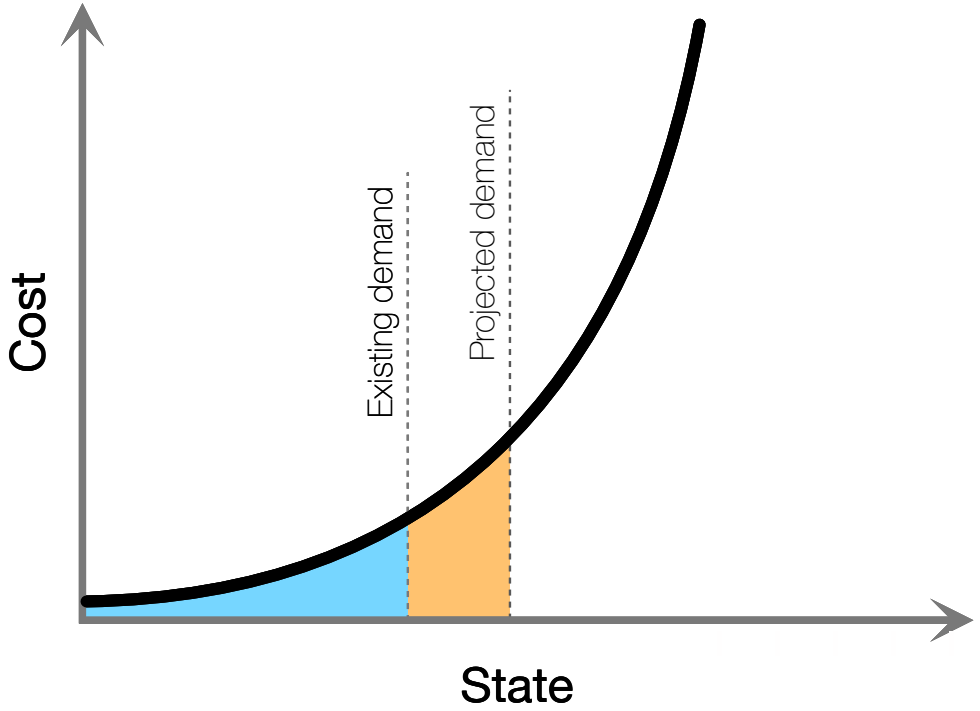


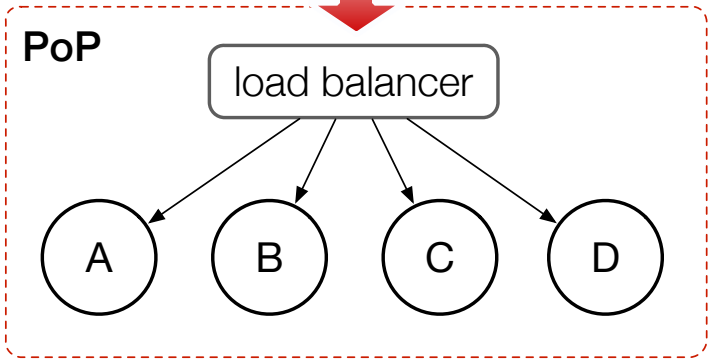


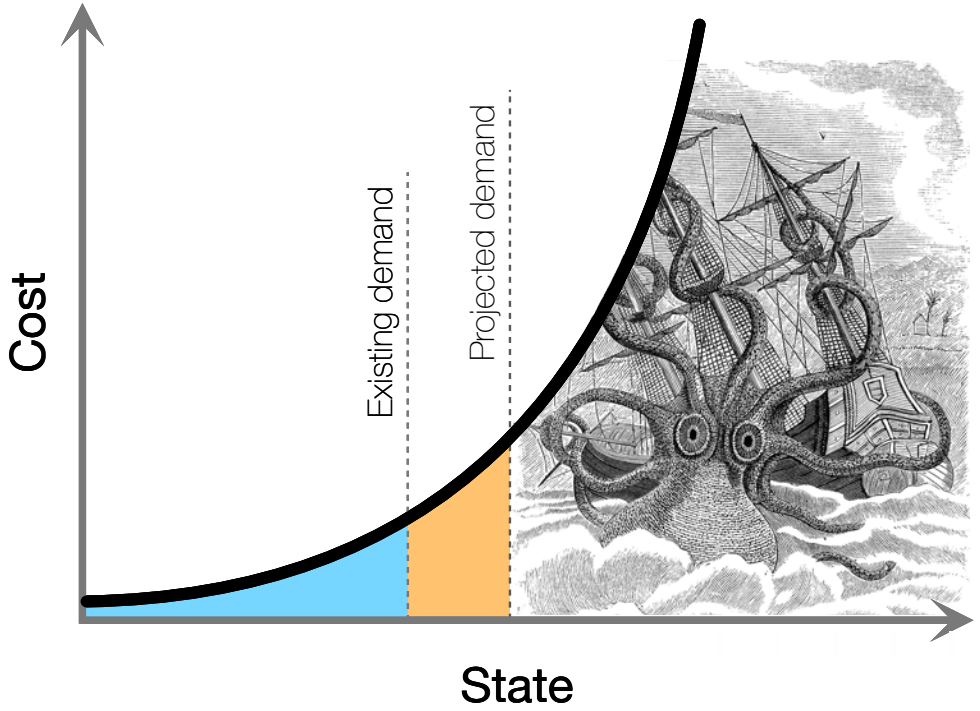


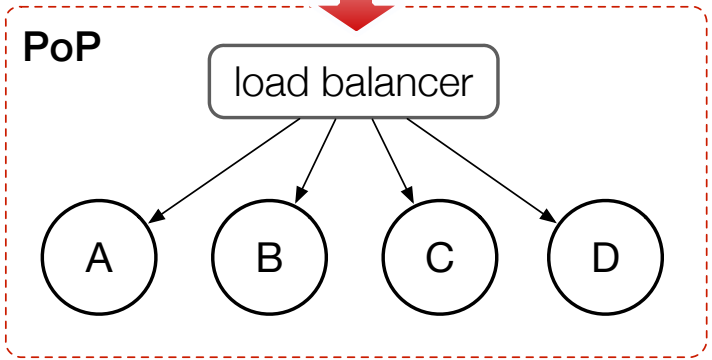


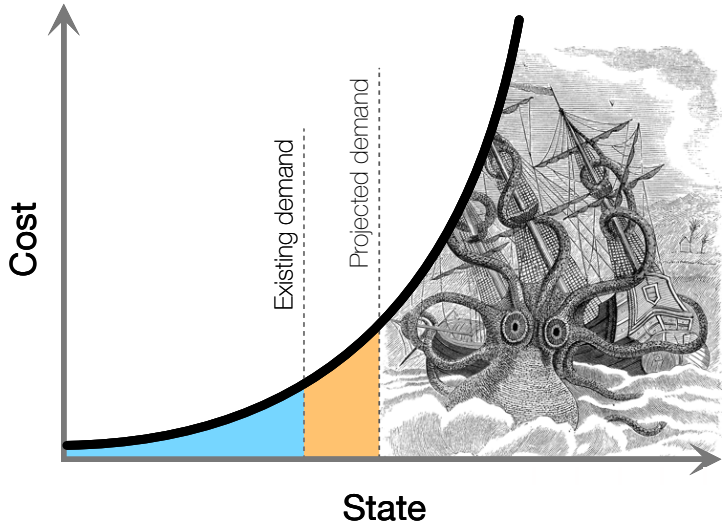






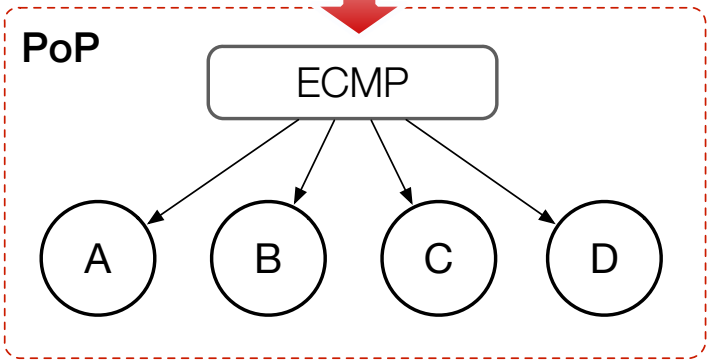


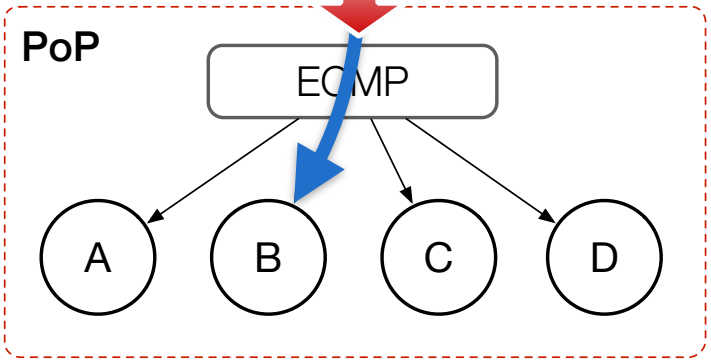




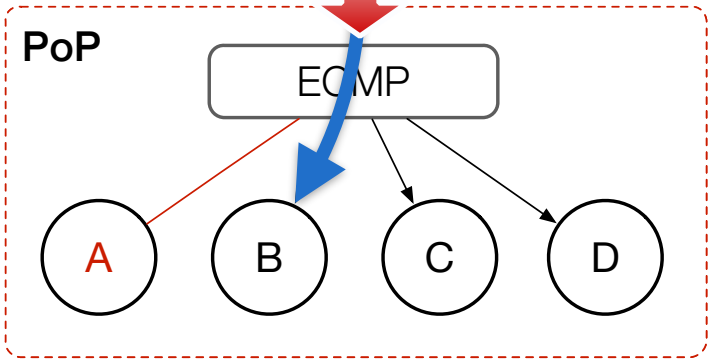
Bad idea:

- you don't control demand
- you don't control DDOS



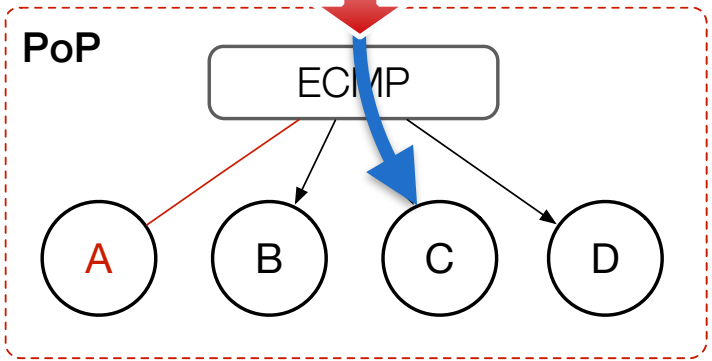


Destination network	Next hop
10.0.0.0/24	A
10.0.0.0/24	B
10.0.0.0/24	C
10.0.0.0/24	D



Destination network	Next hop
10.0.0.0/24	A
10.0.0.0/24	B
10.0.0.0/24	C
10.0.0.0/24	D

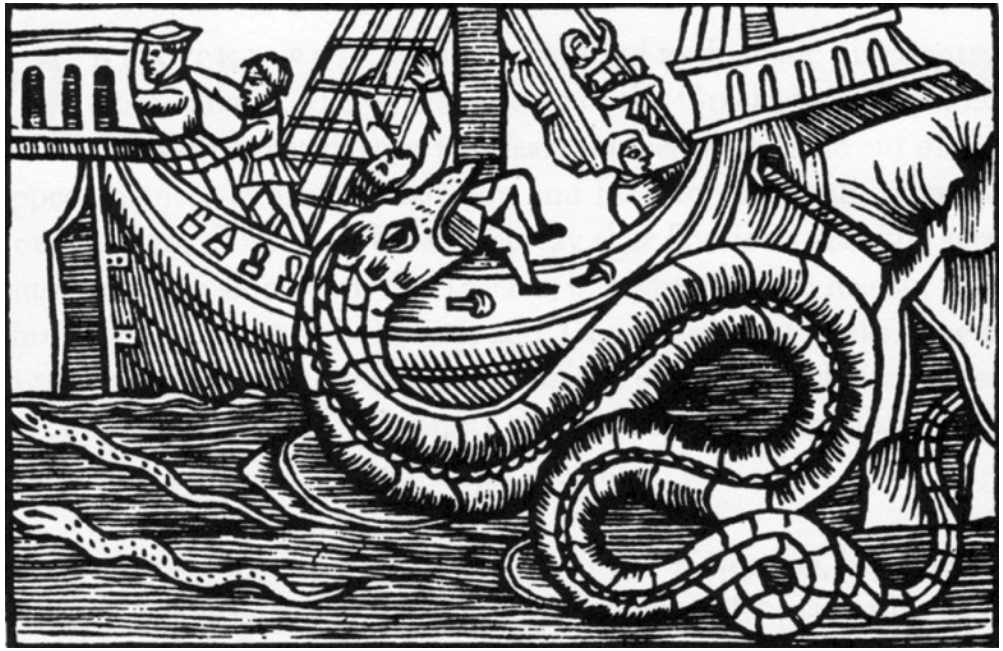
Destination network	Next hop
10.0.0.0/24	B
10.0.0.0/24	C
10.0.0.0/24	D



Bad idea:

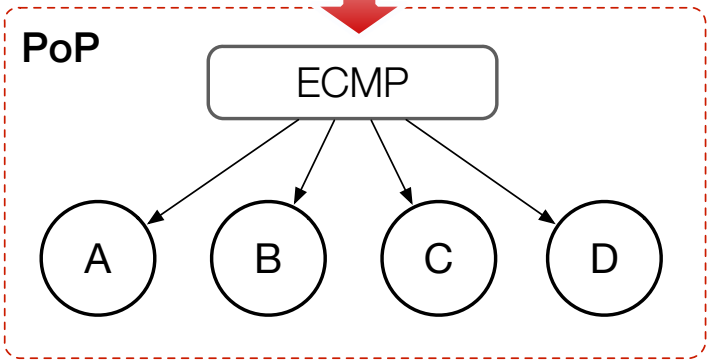
- connection resets
- you don't control rehashing
- you don't control vendor roadmaps

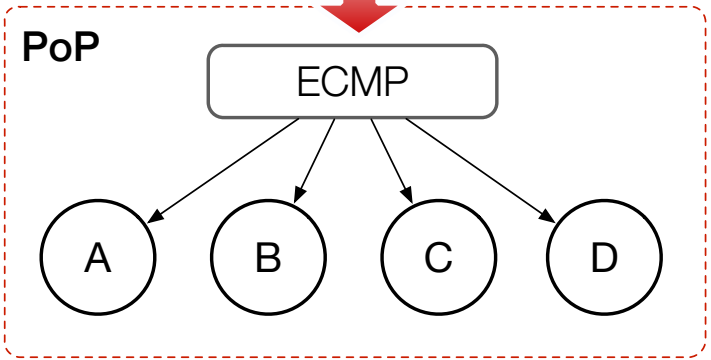
don't resign to fate just because everything sucks



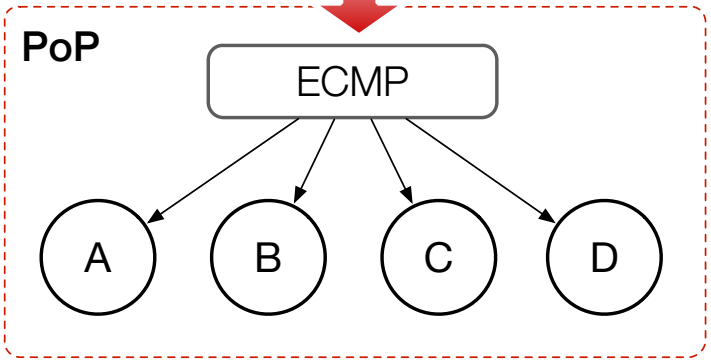
fastly

faild



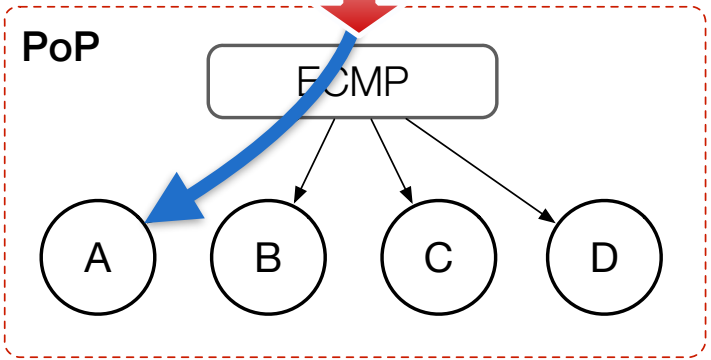


Destination network	Next hop
10.0.0.0/24	10.1.**A**.1
10.0.0.0/24	10.1.**A**.2
10.0.0.0/24	10.1.**A**.3
…	…



Destination network	Next hop
10.0.0.0/24	10.1.**A**.1
10.0.0.0/24	10.1.**A**.2
10.0.0.0/24	10.1.**A**.3
…	…

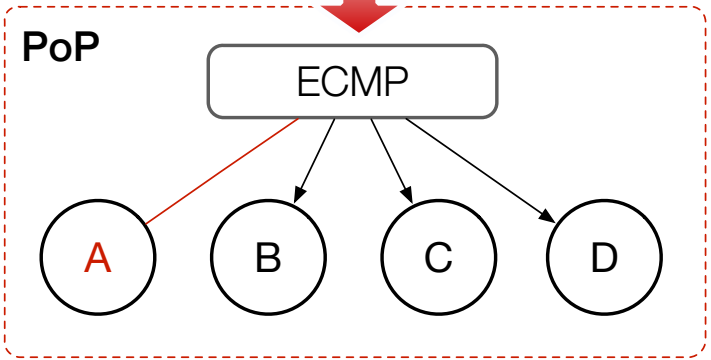
IP Address	MAC
10.1.**A**.1	A:A
10.1.**A**.2	A:A
10.1.**A**.3	A:A
…	…



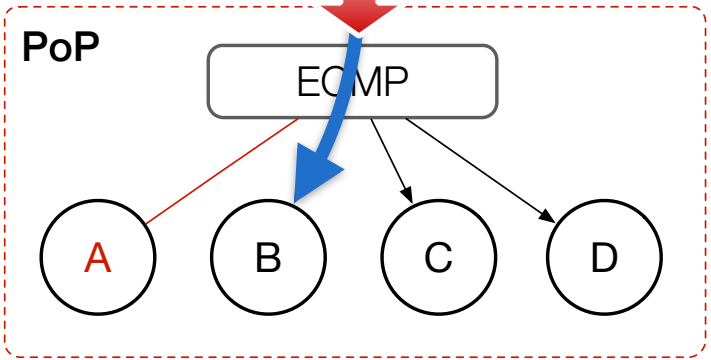
Destination network	Next hop
10.0.0.0/24	10.1.**A**.1
10.0.0.0/24	10.1.**A**.2
10.0.0.0/24	10.1.**A**.3
…	…

IP Address	MAC
10.1.**A**.1	A:A
10.1.**A**.2	A:A
10.1.**A**.3	A:A
…	…

drain a host

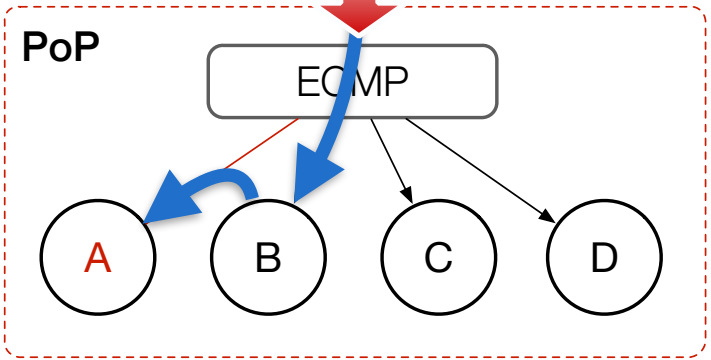


Destination network	Next hop
10.0.0.0/24	10.1.**A**.1
10.0.0.0/24	10.1.**A**.2
10.0.0.0/24	10.1.**A**.3
…	…



Destination network	Next hop
10.0.0.0/24	10.1.**A**.1
10.0.0.0/24	10.1.**A**.2
10.0.0.0/24	10.1.**A**.3
…	…

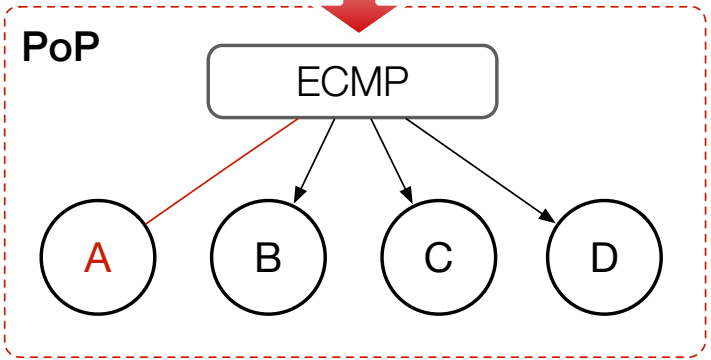
IP Address	MAC
10.1.**A**.1	**B:A**
10.1.**A**.2	C:A
10.1.**A**.3	D:A
…	…



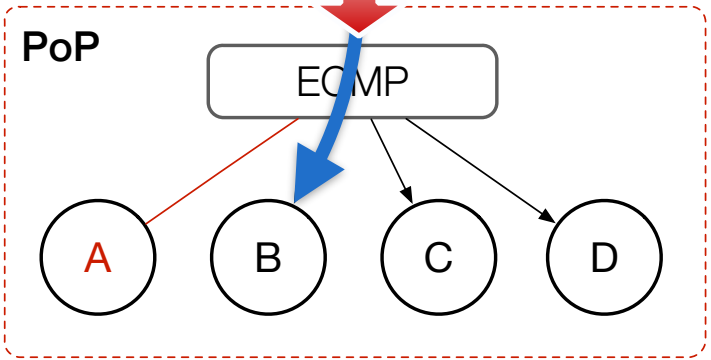
Destination network	Next hop
10.0.0.0/24	10.1.**A**.1
10.0.0.0/24	10.1.**A**.2
10.0.0.0/24	10.1.**A**.3
…	…

IP Address	MAC
10.1.**A**.1	**B:A**
10.1.**A**.2	C:A
10.1.**A**.3	D:A
…	…

cut off to failed state



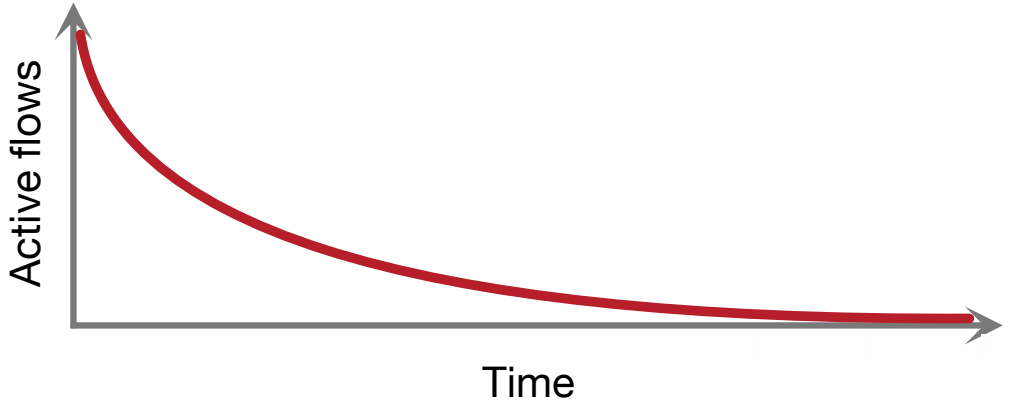
Destination network	Next hop
10.0.0.0/24	10.1.**A**.1
10.0.0.0/24	10.1.**A**.2
10.0.0.0/24	10.1.**A**.3
…	…



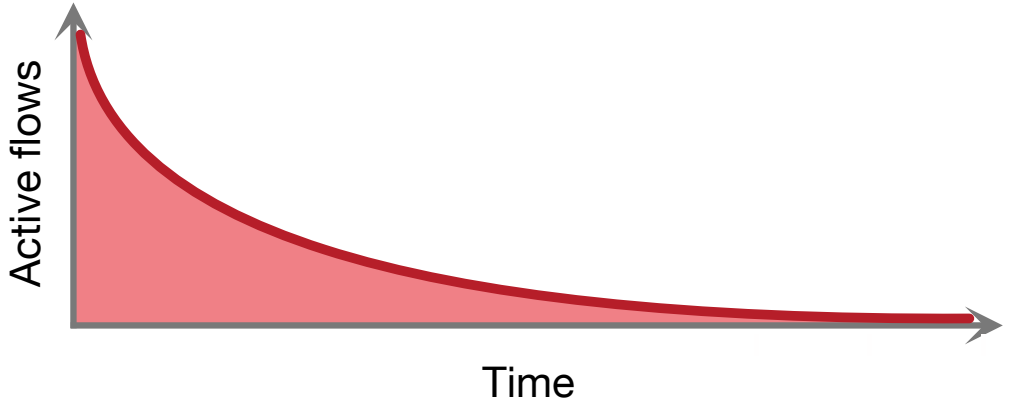
Destination network	Next hop
10.0.0.0/24	10.1.**A**.1
10.0.0.0/24	10.1.**A**.2
10.0.0.0/24	10.1.**A**.3
…	…

IP Address	MAC
10.1.**A**.1	**B:B**
10.1.**A**.2	C:C
10.1.**A**.3	D:D
…	…

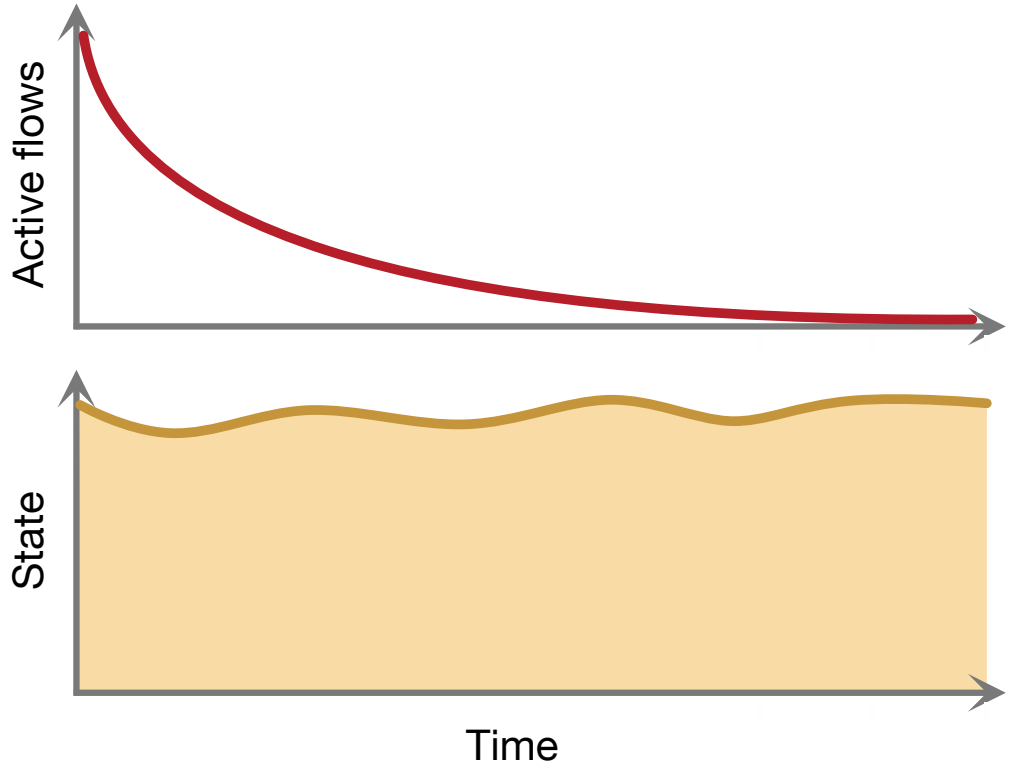
ecmp



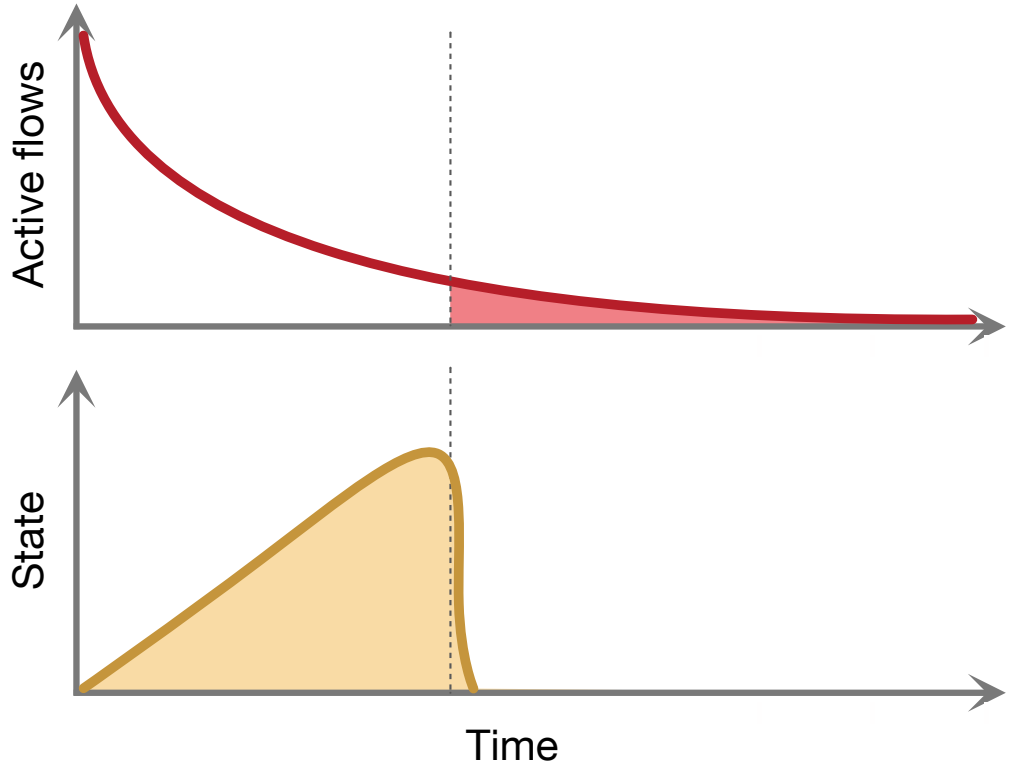
ecmp



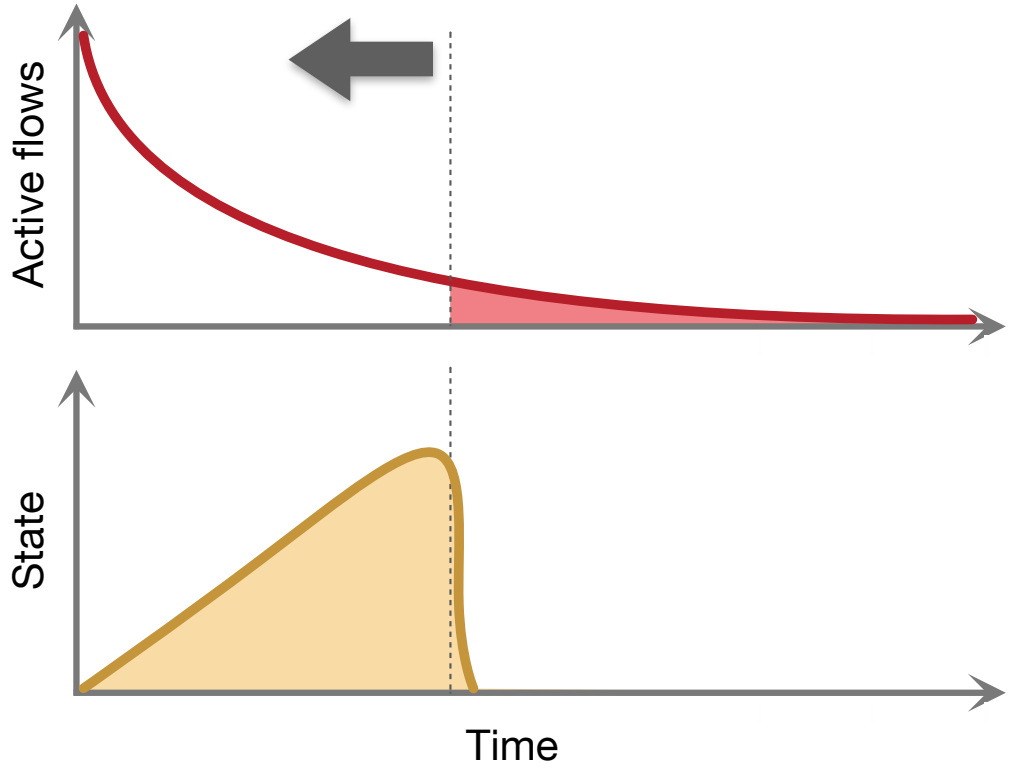
load balancer



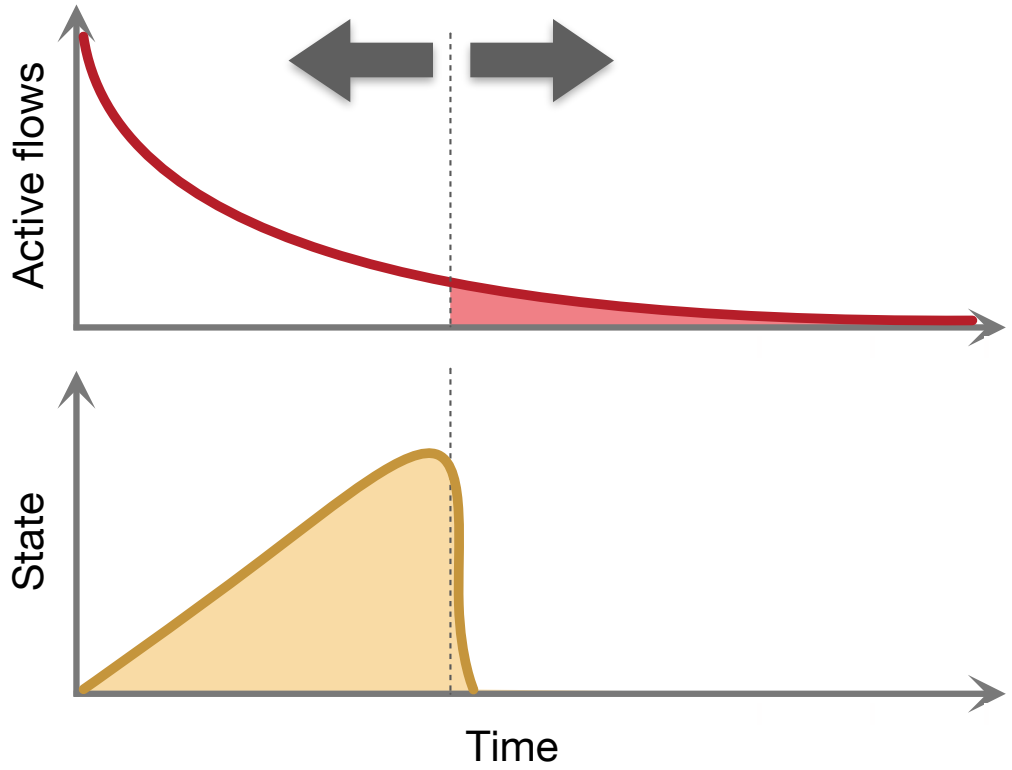
faild



faild



faild



if it's expensive you probably don't need it.

fastly

F5 BIG-IP 10350v

F5 BIG-IP 10350v

$200,000

F5 BIG-IP 10350v

$200,000

$0

load balancer

load ~~balancer~~ balancing

load ~~balancer~~ balancing

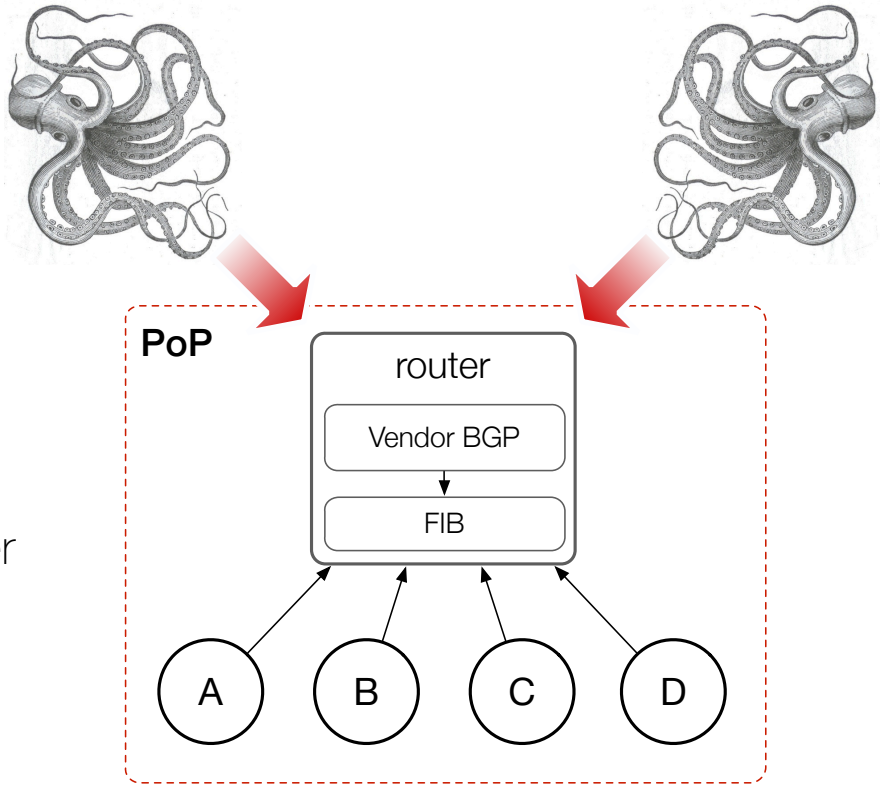
(a load balancer is just an **appliance** which provides load balancing)

distributed load ~~balancer~~ balancing

(a load balancer is just an **appliance** which provides load balancing)

How to build a PoP

- buy a router
- get BGP table from each provider
- install routes to FIB
- servers use default gateway



Juniper MX960 Router

Juniper MX960 Router

~$500,000

router

~~router~~ routing

~~router~~ routing

(a router is just an **appliance** which provides routing)

distributed ~~router~~ routing

(a router is just an **appliance** which provides routing)

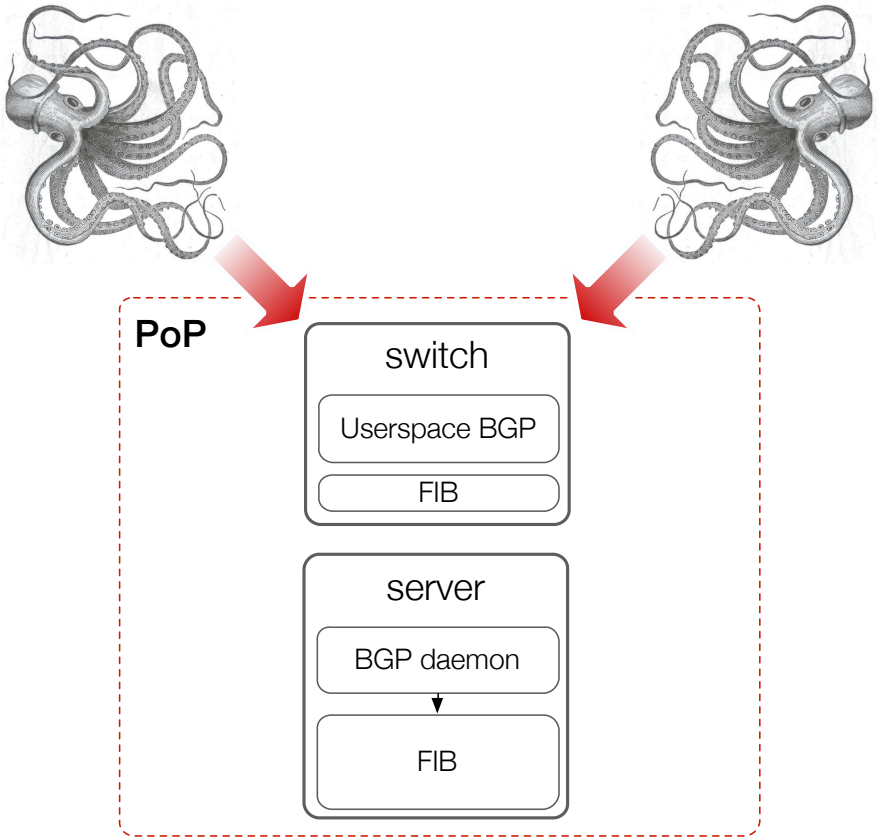
Arista DCS-7150S switch family

Arista DCS-7150S switch family

$29,995

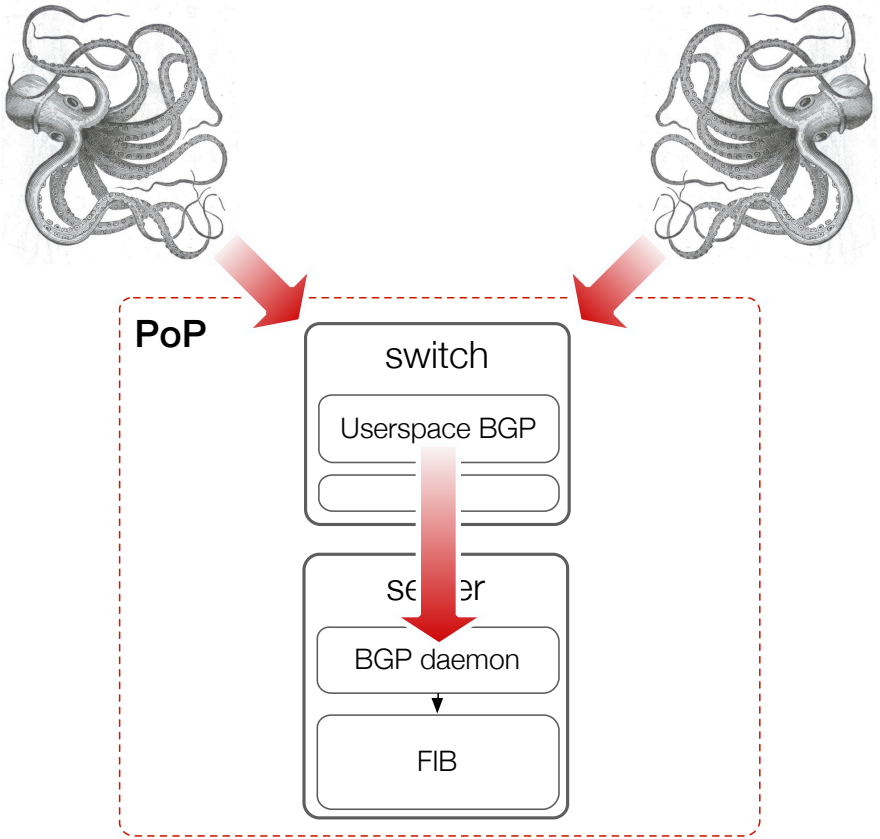
How to build a Fastly PoP

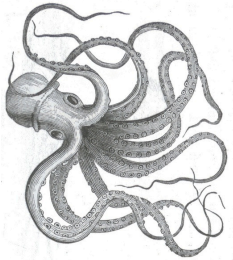
- buy switches
- reflect BGP down to servers
- inject multipath routes into FIB

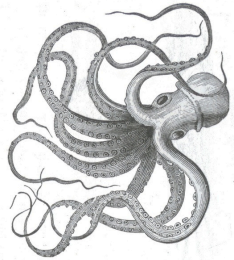


How to build a Fastly PoP

- buy switches
- reflect BGP down to servers
- inject multipath routes into FIB

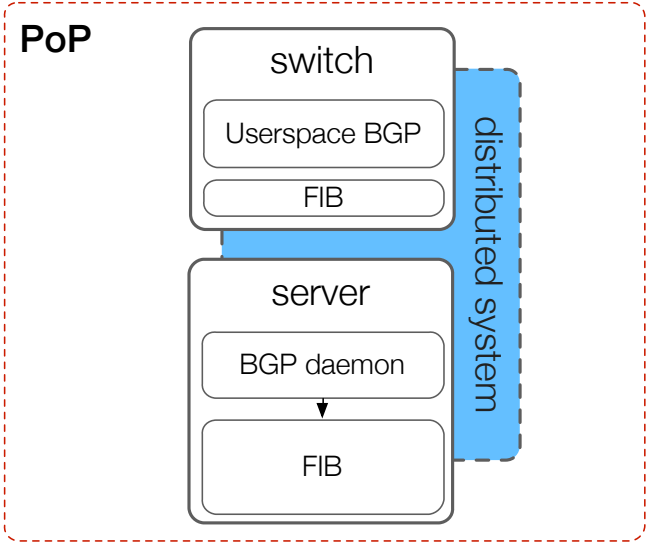






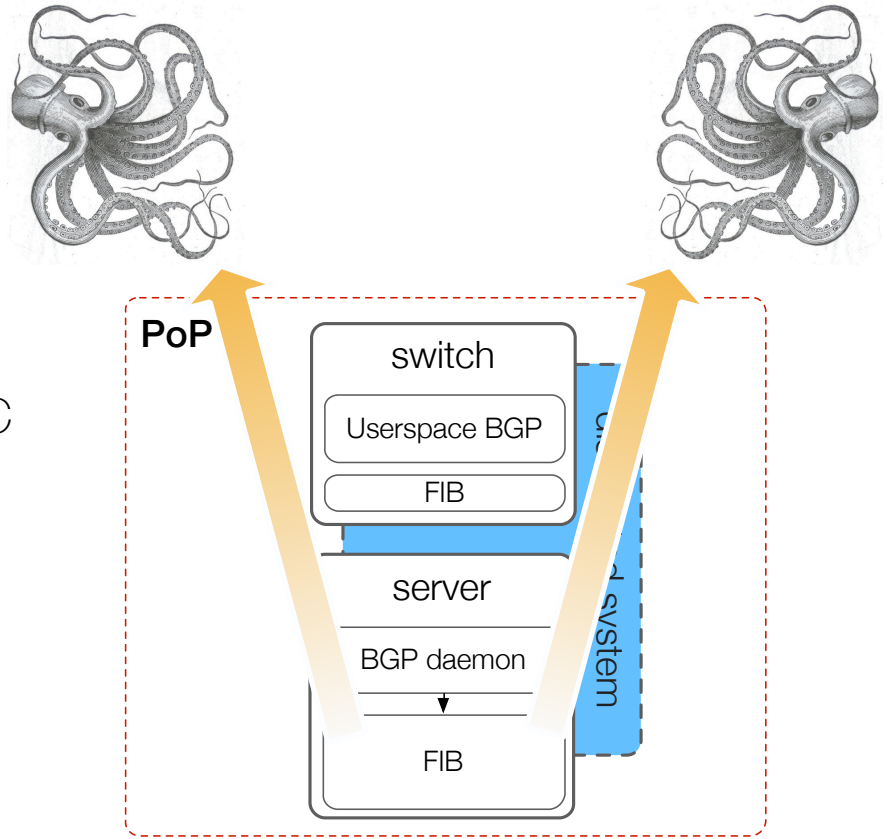
How packets egress Fastly

- switches emit nexthop IP and MAC
- servers configure p2p link / ARP
- send directly to provider nexthop!



How packets egress Fastly

- switches emit nexthop IP and MAC
- servers configure p2p link / ARP
- send directly to provider nexthop!



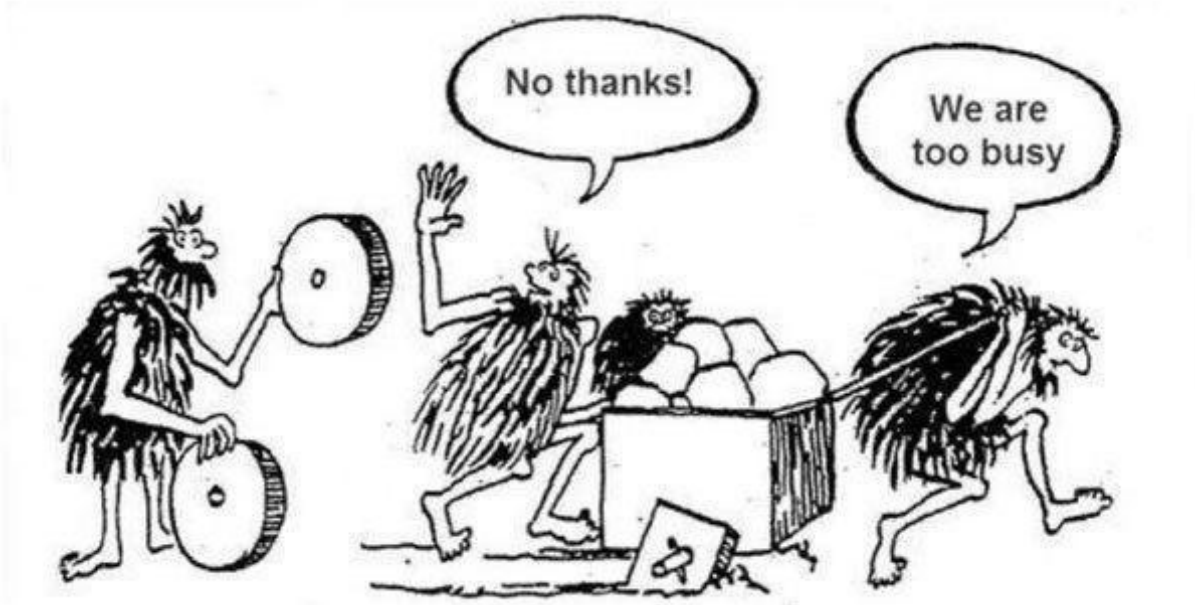
```
joao@cache        :~$ sudo birdc show route count
BIRD 1.4.4 ready.
2099355 of 2099355 routes for 524852 networks
```

Fastly PoPs: engineering perspective

Fastly PoPs: investor perspective

iv

It's easier to make people less busy than hire people.



fastly

Yes I would

software

networking

"you wouldn't do that to a switch"

software

“Networking is hard”

“Networking is hard”

resource constraints

“Networking is hard”

resource constraints

protocol standards

“Networking is hard”

resource constraints

protocol standards

security concerns

“Networking is hard”

resource constraints

protocol standards

security concerns

network vendors

where is time spent needlessly?

pinpointing path failures

st-ping: probe all upstreams

```
joao@cache         :~$ sudo st-ping 8.8.8.8
Pinging 8.8.8.8 via 12 upstreams.
  Upstream    Intf          Nexthop  Sent   Loss       Min       Avg       Max       Dev
  cogent      p5p1                     10   0.0%     1.023     1.042     1.056     0.022
  cogent      p3p2                     10   0.0%     1.018     1.042     1.079     0.034
  cogent      p3p1                     10   0.0%     1.014     1.029     1.059     0.011
  cogent      p5p2                     10   0.0%     1.024     1.036     1.063     0.039
  l3          p3p2                     10   0.0%     0.867     0.878     0.902     0.016
  l3          p5p2                     10   0.0%     1.347     1.357     1.383     0.038
  l3          p3p1                     10   0.0%       1.3     1.318     1.341     0.021
  l3          p5p1                     10   0.0%      0.88     0.887     0.902     0.027
* telia       p3p1                     10   0.0%    26.485    26.634    27.243      0.32
* telia       p3p2                     10   0.0%    27.963    28.587    29.692     0.674
* telia       p5p1                     10   0.0%     25.81    26.621     27.24     0.446
* telia       p5p2                     10   0.0%    27.953    29.058    29.669     0.634
```

changing route preferences

```
switch        #conf
switch        (config)#l3
switch        (config-if-Et3)#show active
 interface Ethernet3
   description l3_1 [asia,dns1,dns2,dns3,dns4,http1,http2,http3,http4,site] is up since 2015-02-27
   load-interval 5
   ip access-group inboundc in
   ip access-group outbound out
   queue-monitor length thresholds 1024 128
   no lldp receive
```

upstream alias

```
switch        #conf
switch        (config)#l3
switch        (config-if-Et3)#show active
 interface Ethernet3
   description l3_1 [asia,dns1,dns2,dns3,dns4,http1,http2,http3,http4,site] is up since 2015-02-27
   load-interval 5
   ip access-group inboundc in
   ip access-group outbound out
   queue-monitor length thresholds 1024 128
   no lldp receive
```

announced prefixes

```
switch        #conf
switch        (config)#l3
switch        (config-if-Et3)#show active
 interface Ethernet3
   description l3_1 [asia,dns1,dns2,dns3,dns4,http1,http2,http3,http4,site] is up since 2015-02-27
   load-interval 5
   ip access-group inboundc in
   ip access-group outbound out
   queue-monitor length thresholds 1024 128
   no lldp receive
```

live BGP info

```
switch[redacted](config-if-Et3)#desc +15169
switch[redacted](config-if-Et3)#show active
 interface Ethernet3
   description l3_1 [asia,dns1,dns2,dns3,dns4,http1,http2,http3,http4,site] {+15169} is up since 2015-02-27
   load-interval 5
   ip access-group inboundc in
   ip access-group outbound out
   queue-monitor length thresholds 1024 128
   no lldp receive
```

increase Google localpref

```
switch[redacted](config-if-Et3)#desc +15169
switch[redacted](config-if-Et3)#show active
 interface Ethernet3
   description l3_1 [asia,dns1,dns2,dns3,dns4,http1,http2,http3,http4,site] {+15169} is up since 2015-02-27
   load-interval 5
   ip access-group inboundc in
   ip access-group outbound out
   queue-monitor length thresholds 1024 128
   no lldp receive
```

localpref overrides

```
joao@cache        :~$ sudo st-ping 8.8.8.8
Pinging 8.8.8.8 via 12 upstreams.
  Upstream    Intf          Nexthop  Sent   Loss       Min       Avg       Max       Dev
  cogent      p3p1                     10   0.0%     1.018     1.028     1.037     0.035
  cogent      p5p1                     10   0.0%      1.02     1.037     1.052     0.044
  cogent      p3p2                     10   0.0%     1.011     1.031      1.06     0.028
  cogent      p5p2                     10   0.0%     1.026     1.033     1.049     0.026
* l3          p3p1                     10   0.0%       1.3     1.319     1.363     0.035
* l3          p5p2                     10   0.0%     1.344     1.357     1.383     0.034
* l3          p3p2                     10   0.0%     0.866     0.879     0.899     0.033
* l3          p5p1                     10   0.0%     0.869     0.885     0.925     0.038
  telia       p3p1                     10   0.0%    25.802     26.55    27.202     0.379
  telia       p5p1                     10   0.0%    26.481    26.713    27.231     0.346
  telia       p5p2                     10   0.0%    27.943    28.803     29.47     0.619
  telia       p3p2                     10   0.0%    27.948    28.579    29.669     0.667
```

changing prefix announcements

```
switch        (config-if-Et3)#desc !http
switch        (config-if-Et3)#show active
 interface Ethernet3
   description l3_1 [!http1,!http2,!http3,!http4,asia,dns1,dns2,dns3,dns4,site] {+15169} is feed since 2015-02-27
   load-interval 5
   ip access-group inboundc in
   ip access-group outbound out
   queue-monitor length thresholds 1024 128
   no lldp receive
```

withdraw all HTTP anycast prefixes

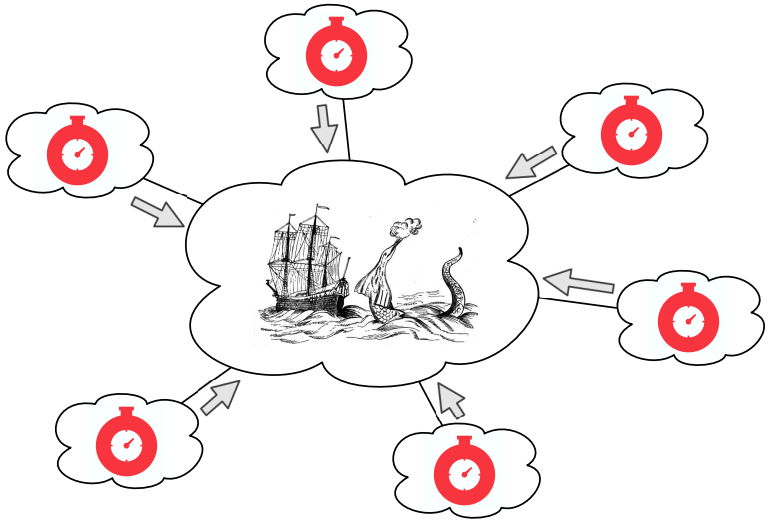
```
switch        (config-if-Et3)#desc !http
switch        (config-if-Et3)#show active
 interface Ethernet3
   description l3_1 [!http1,!http2,!http3,!http4,asia,dns1,dns2,dns3,dns4,site] {+15169} is feed since 2015-02-27
   load-interval 5
   ip access-group inboundc in
   ip access-group outbound out
   queue-monitor length thresholds 1024 128
   no lldp receive
```

BGP session status

```
switch        (config-if-Et3)#desc !http
switch        (config-if-Et3)#show active
 interface Ethernet3
   description l3_1 [!http1,!http2,!http3,!http4,asia,dns1,dns2,dns3,dns4,site] {+15169} is feed since 2015-02-27
   load-interval 5
   ip access-group inboundc in
   ip access-group outbound out
   queue-monitor length thresholds 1024 128
   no lldp receive
```

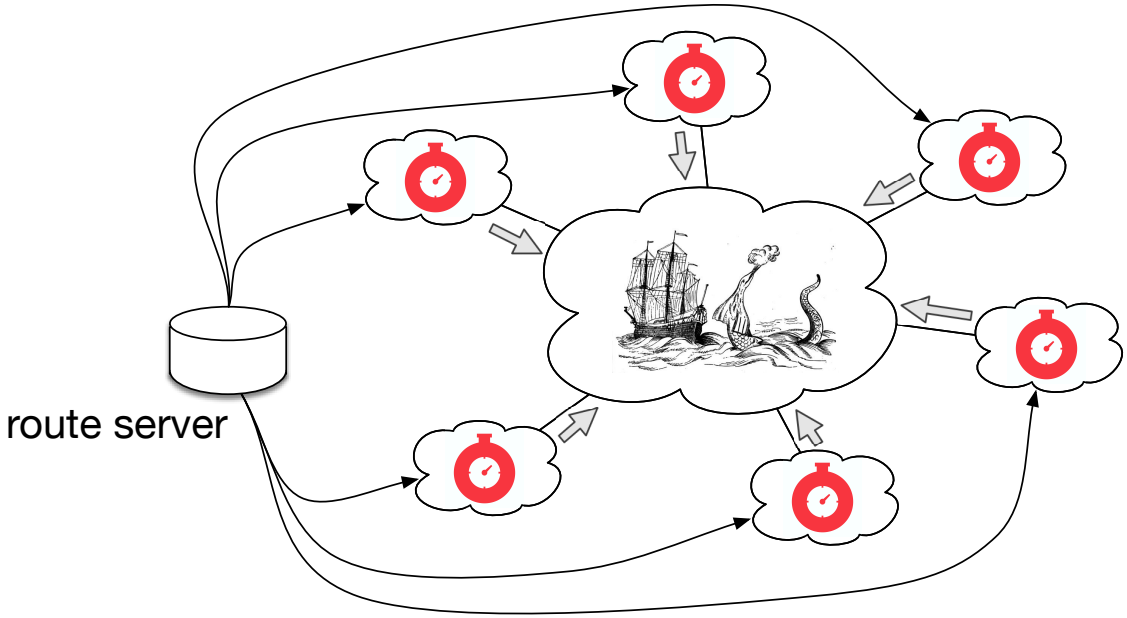
BGP session status

changing global routing policy



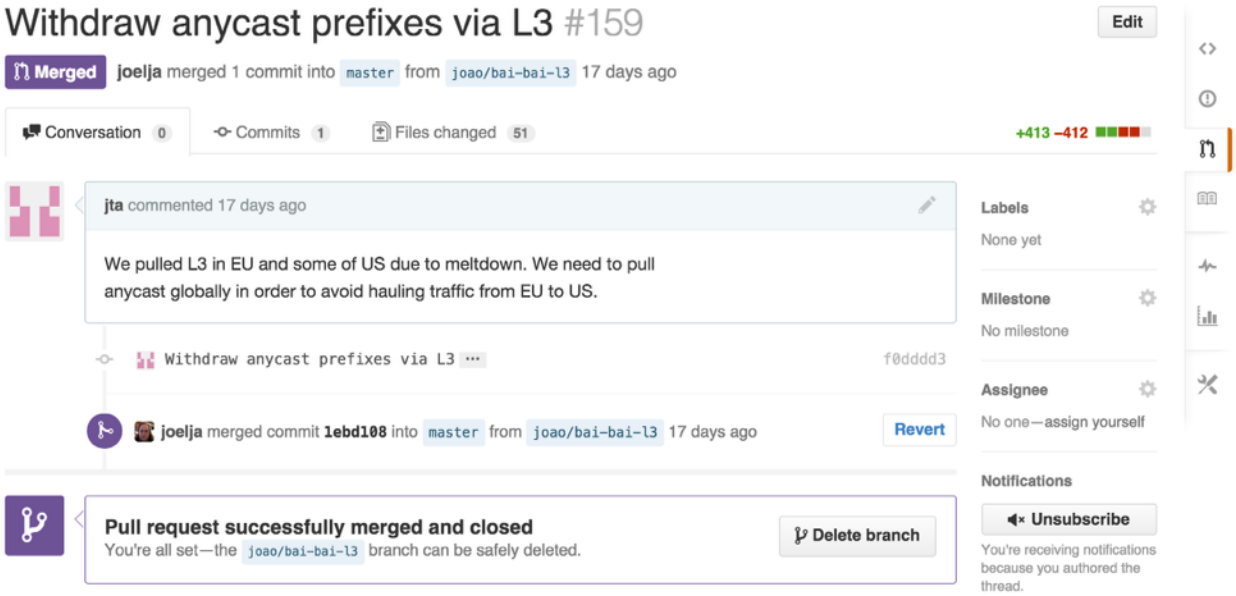
We generate lots of BGP announcements

- changing policy manually is hard
- changing policy per-device takes long



We generate lots of BGP announcements

- changing policy manually is hard
- changing policy per-device takes long



Stage and deploy via Github

- generate diff of routing policy and exported routes
- peer reviewed, endlessly revertible

```
@@ -33,6 +33,7 @@ function policy_anycast(int pop; string switch) {
33                                                  33
34          #neteng-414 no singtel                  34          #neteng-414 no singtel
35          l3_no_export_asn(7473);                 35          l3_no_export_asn(7473);
                                                    36  +       no_export();
36                                                  37
37      }                                           38      }
38                                                  39
```

Staging lists affected switches and prefixes

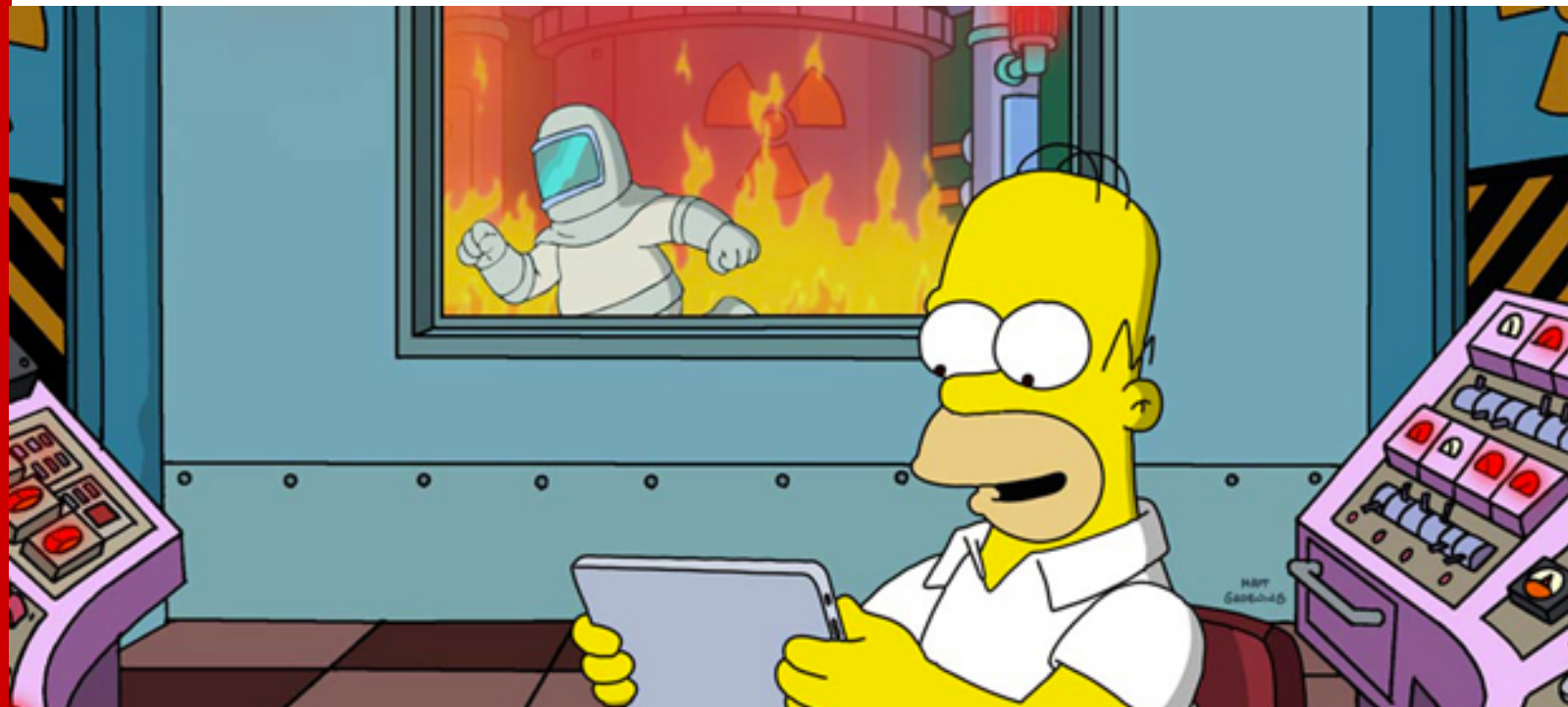
- human error could withdraw Fastly from the Internet
- hard to automate, so make sure people can get it right first

Seems so simple…

- reduced time spent needlessly
- reduced human error dramatically
- allowed us to train netops out of our datacenter team
- Arista eAPI allows description changes: instant RESTful orchestration

V

existing best practices won't save you.



fastly

Saving money

- buy bare essentials
- distribute everything
- efficiency matters

Saving time

- correct architecture helps!
- reduce cognitive overhead
- solve ops first, automate later

Be wary of:

- best practices
- cool stuff
- perfect

www.fastly.com/about/careers

March 16th 2015 | João Taveira Araújo

@jta