LinkedIn™

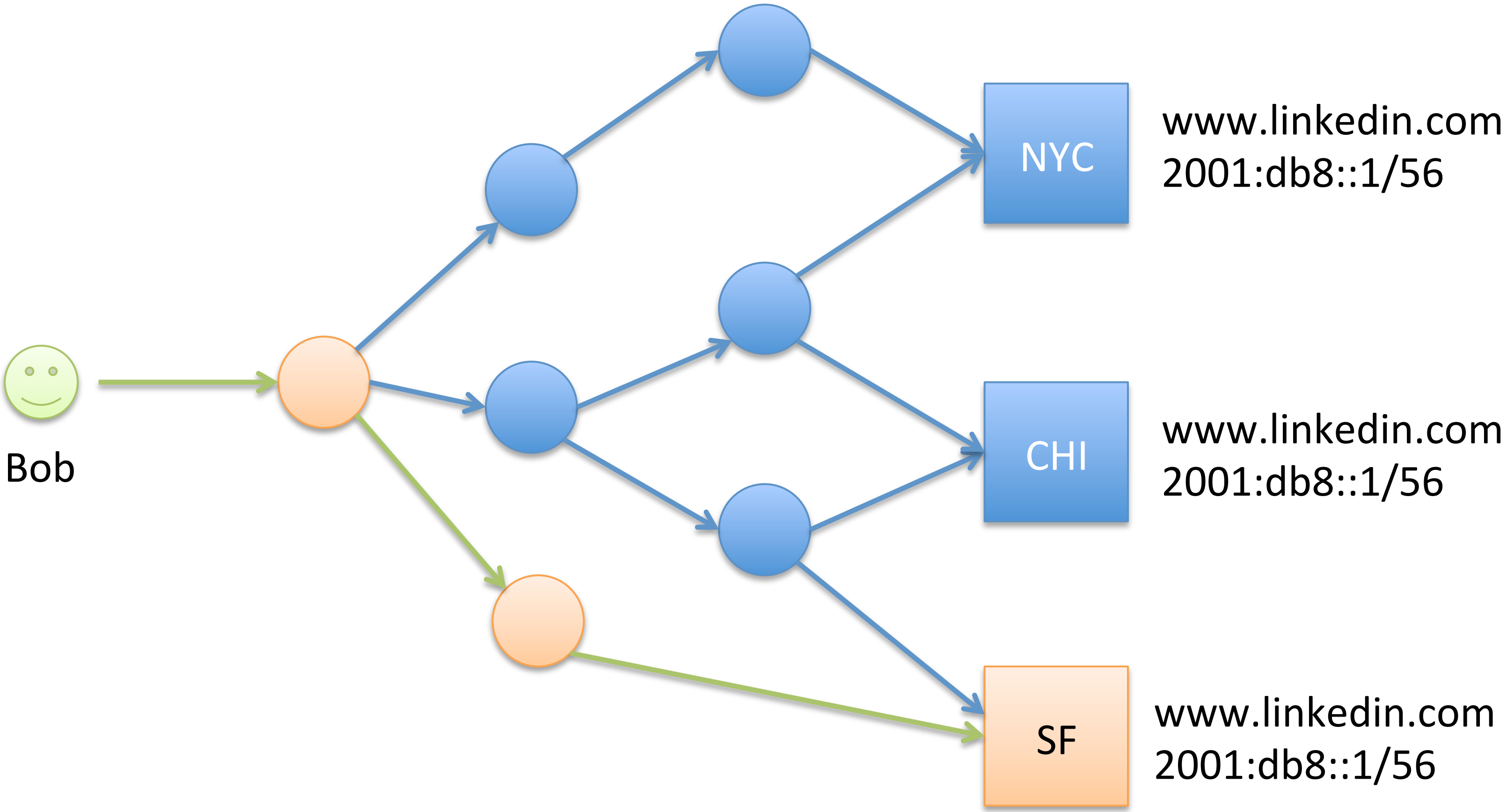
How LinkedIn used TCP Anycast to make the site faster

Ritesh Maheshwari

Shawn Zandi

Anycast

- Anycast provides a distributed service via routing.
- It is not really different than unicast.
  - NLRI object with multiple next-hops.
- It simply works for both TCP and UDP applications. (use with cautions!)



Bob

NYC

www.linkedin.com
2001:db8::1/56

CHI

www.linkedin.com
2001:db8::1/56

SF

www.linkedin.com
2001:db8::1/56

Anycast with ECMP

- Not a real issue in today’s internet
- Consistent flow routing is required (per packet load balancing breaks Anycast) – Pretty Much Standard
- Most BGP implementations do not load balance across different AS-PATHs even with same size.

Anycast Complications

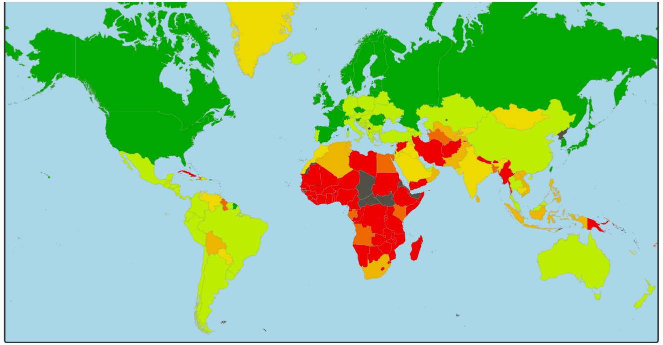
- Broken MTU Challenges
  - ICMP message may not reach the intended receiver to report MTU problem. Adjusting MSS can help.
- RPF Checks
- Multiple covering prefixes - Only one Service Address should be covered by each advertised prefix /24 or /56
- Monitoring!

But!

How to measure Anycast effectiveness?

What is RUM?

JavaScript (Client-code) to measure performance



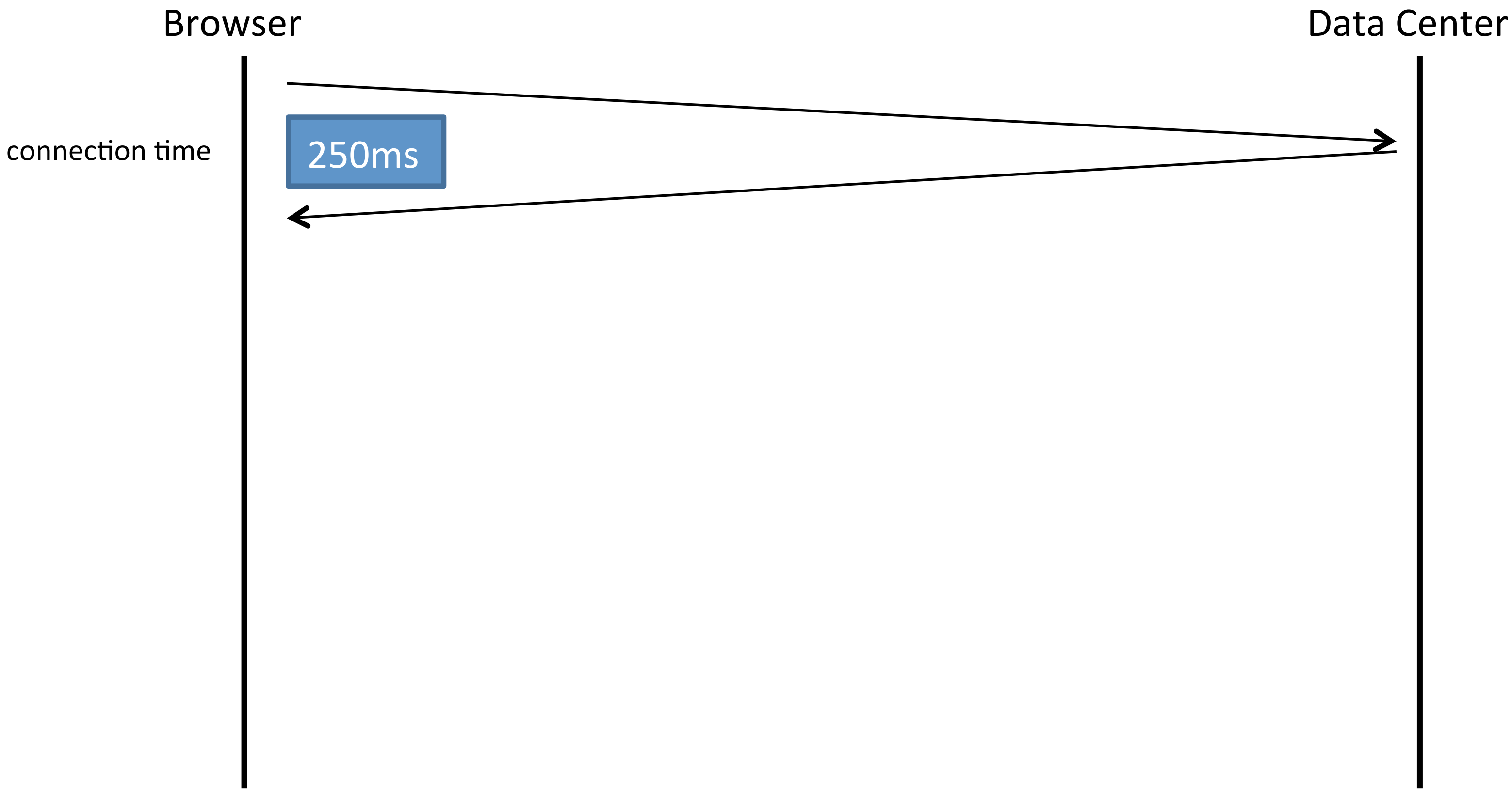
- DNS Time
- Connection time
- First Byte Time
- Download Time
- Page Load Time

What are PoPs?

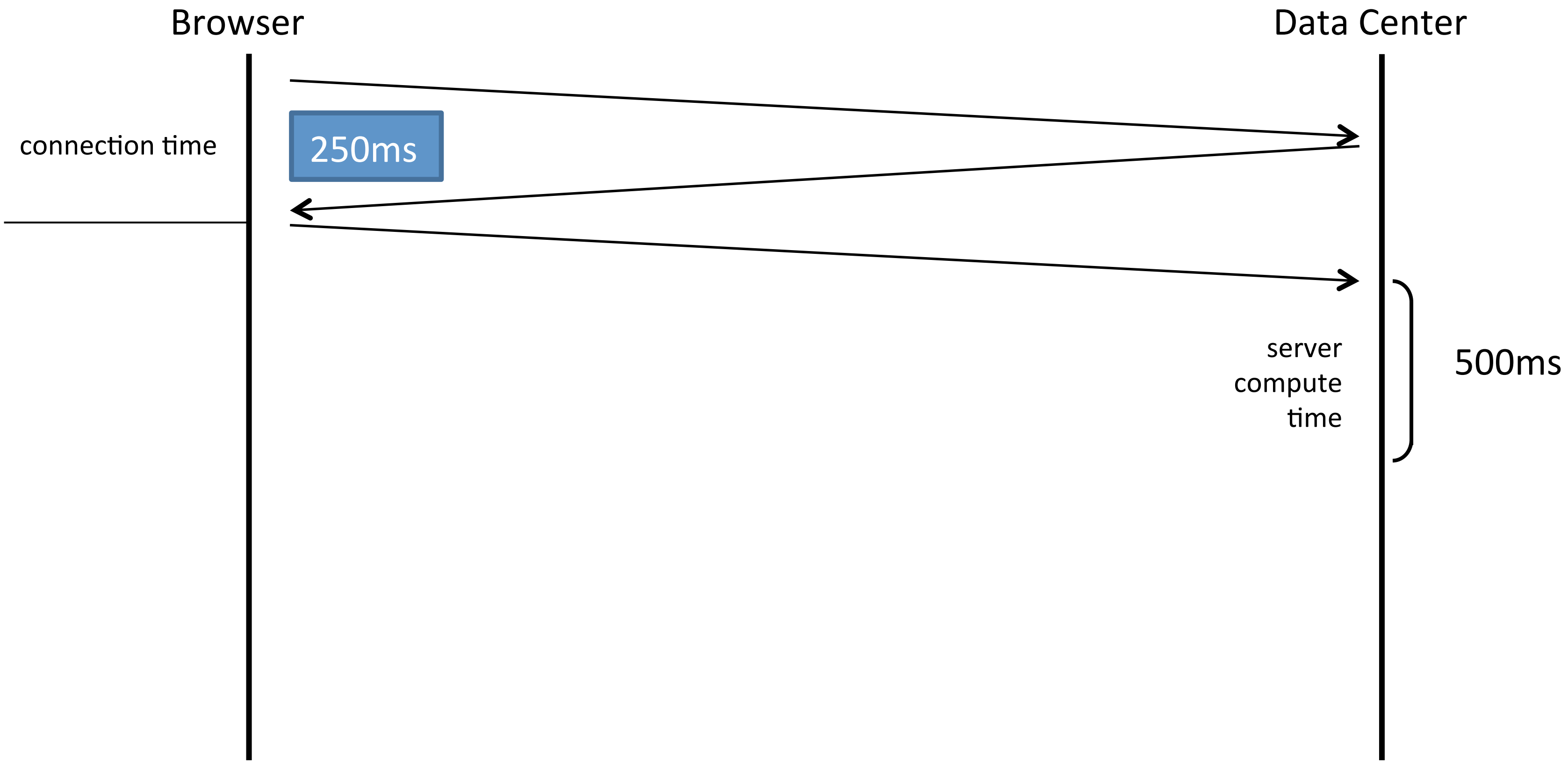
Point of Presence / PoP

- Small-scale data centers
- Proxy servers at LinkedIn (ATS)

Without PoPs

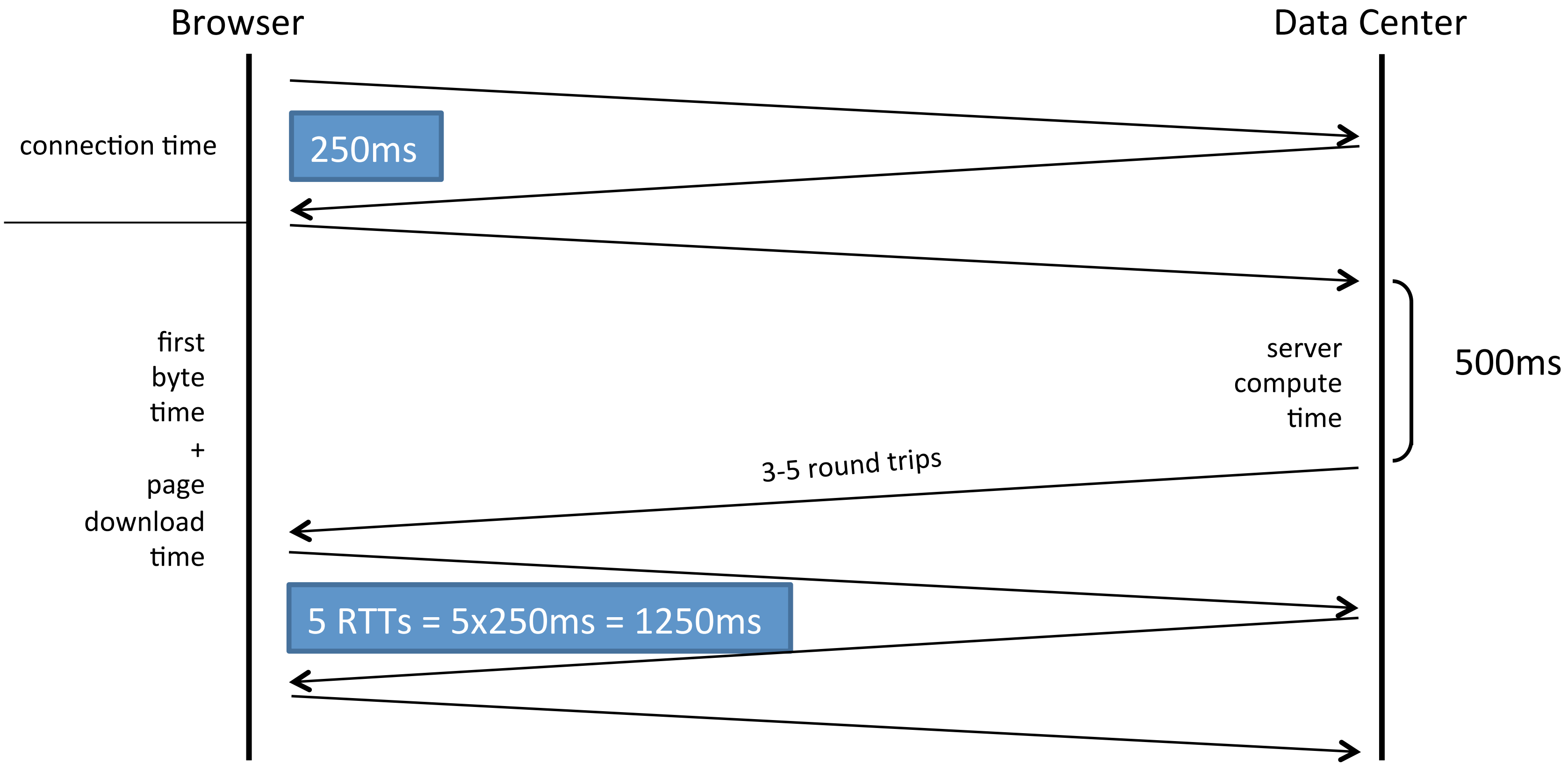


Without PoPs

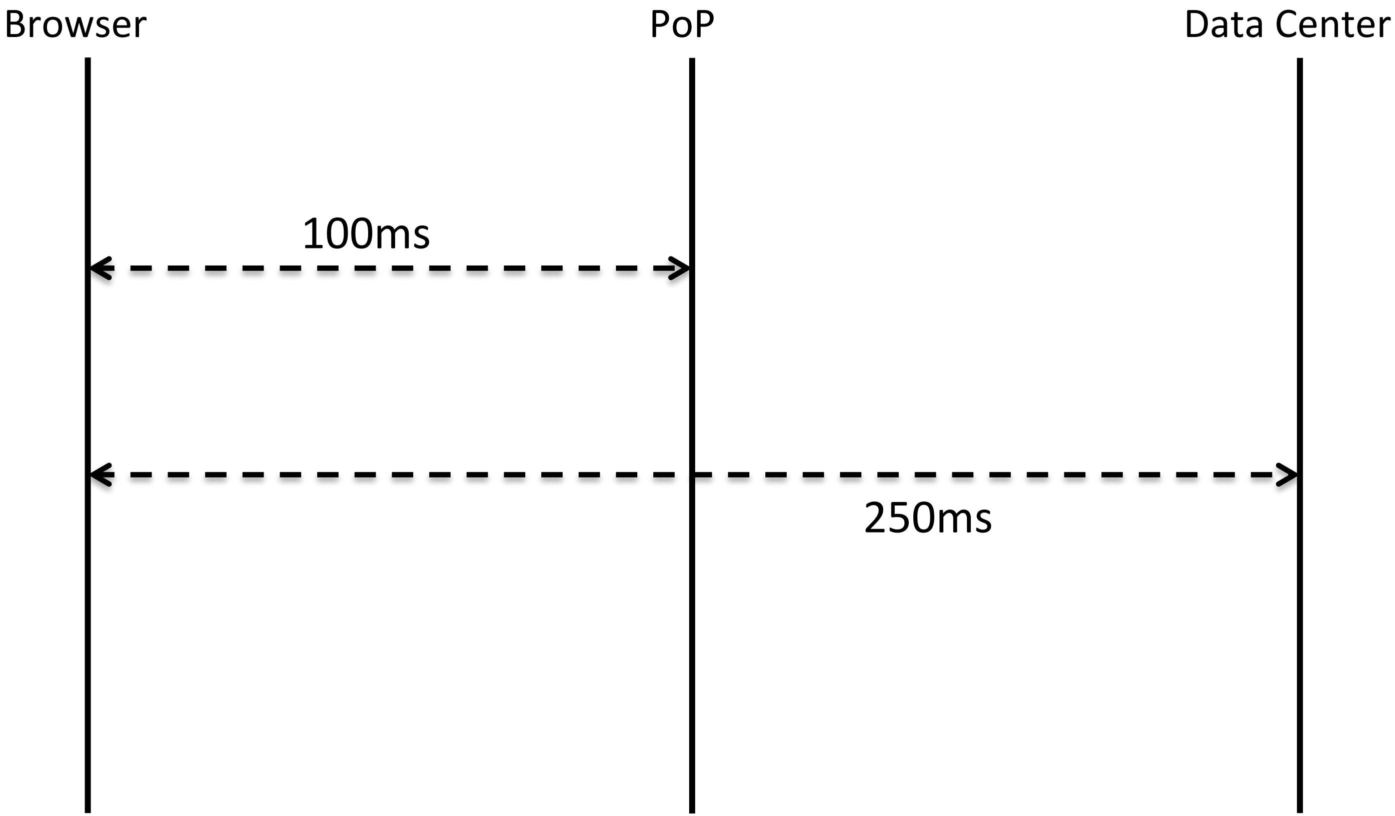


Without PoPs

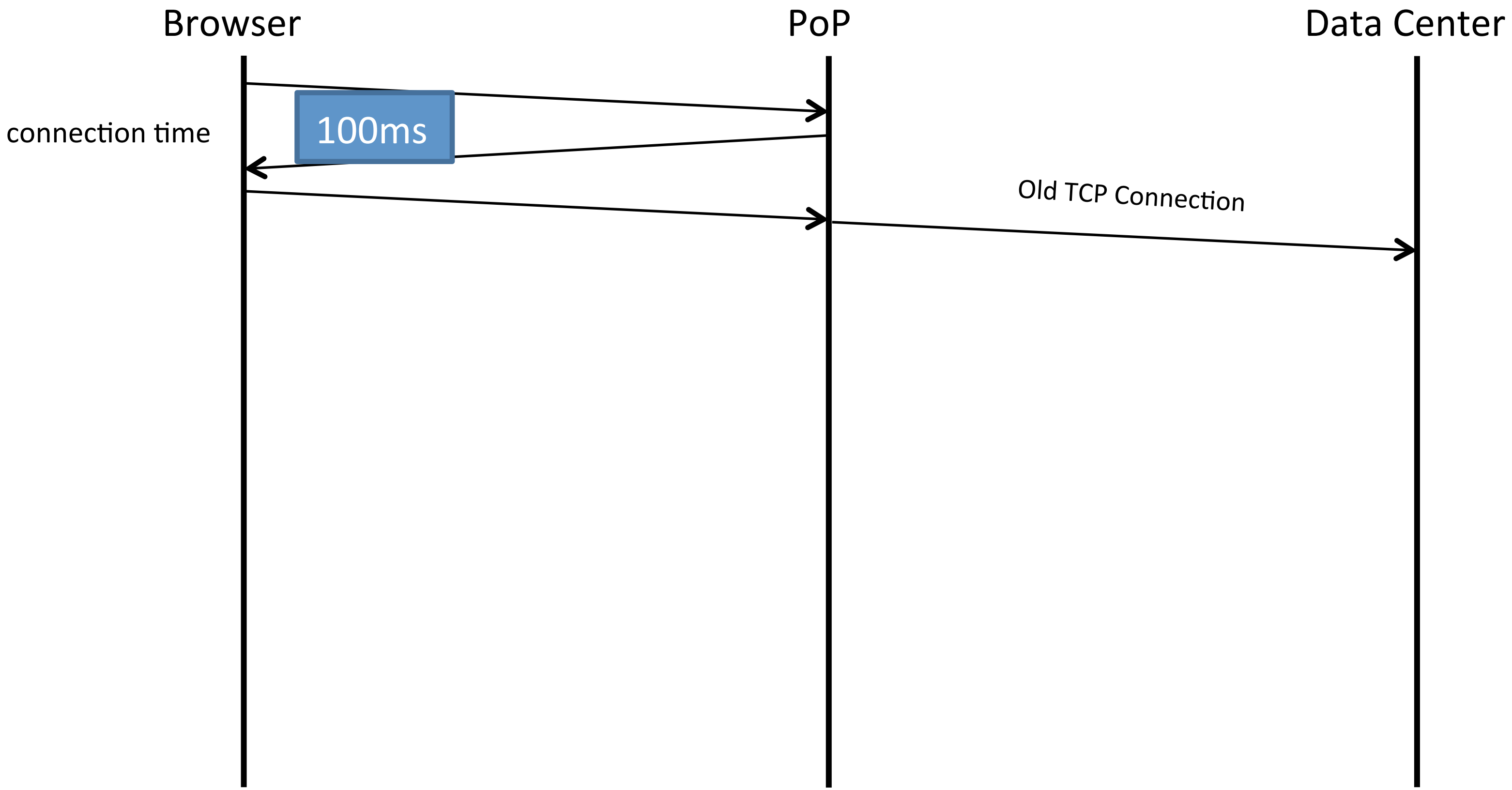
Total = 2000ms



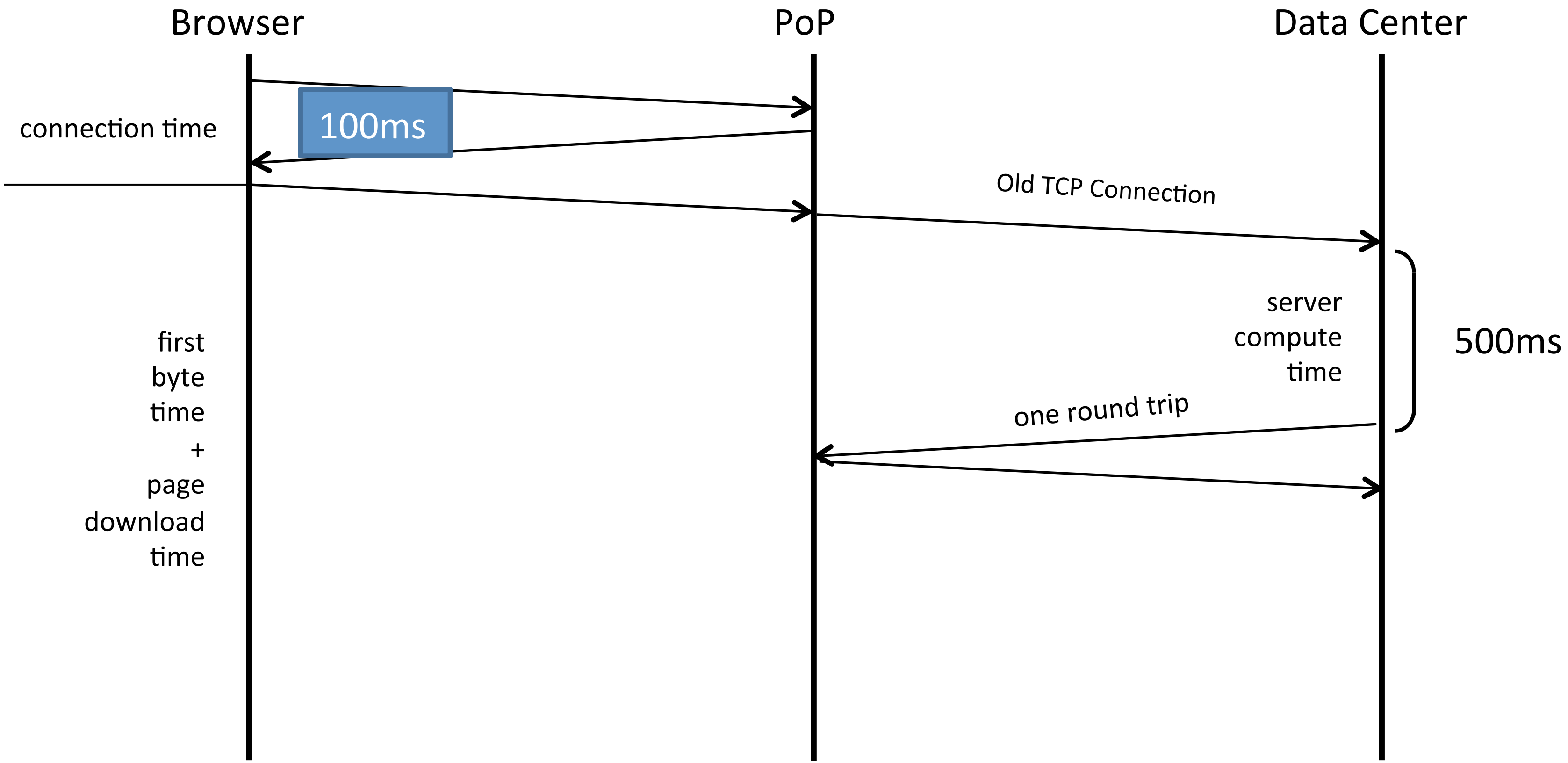
With PoPs



With PoPs


Browser
PoP
Data Center
connection time
100ms
Old TCP Connection

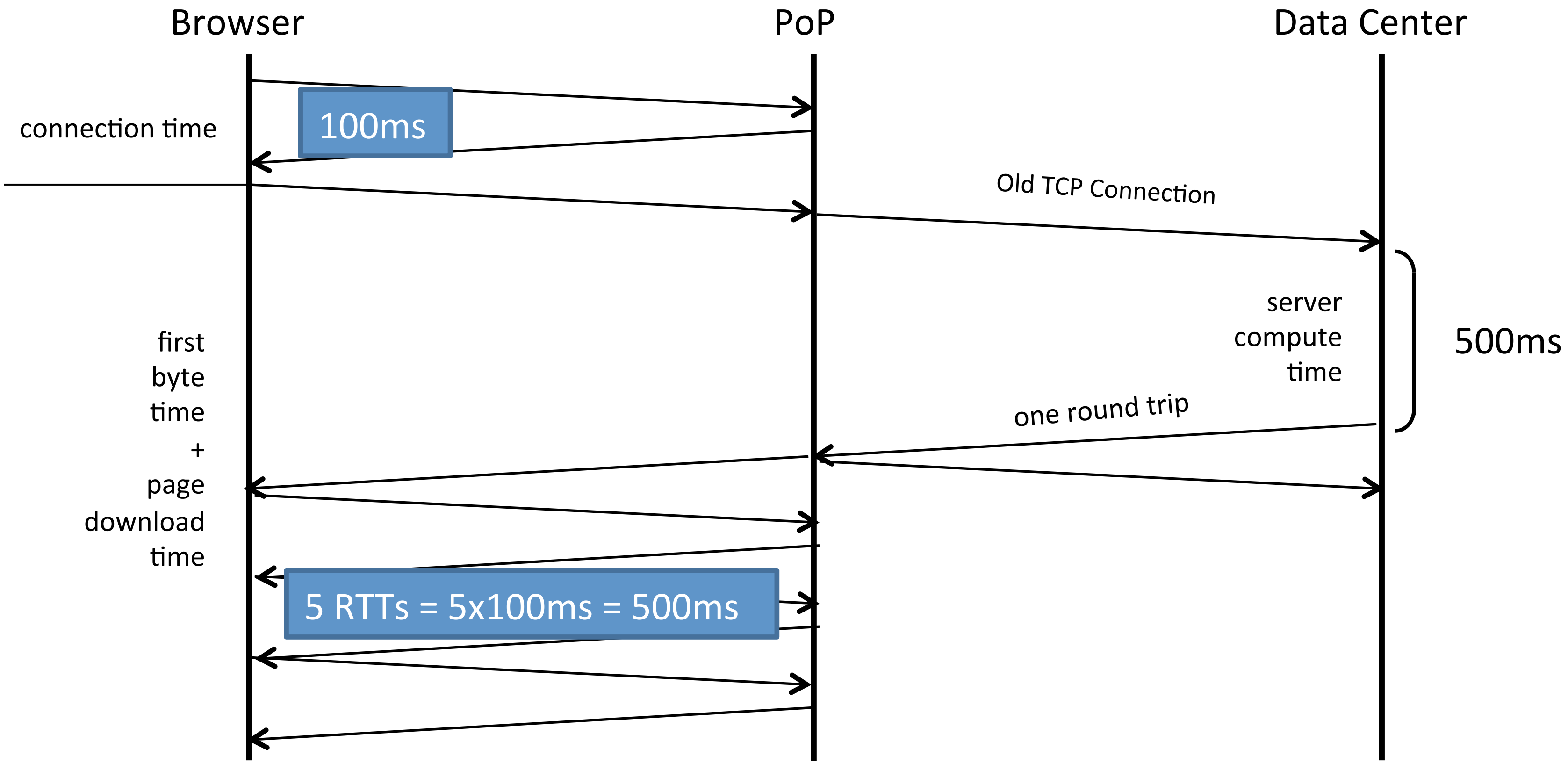
With PoPs


Browser
PoP
Data Center
connection time
100ms
Old TCP Connection
server
compute
time
500ms
one round trip
first
byte
time
+
page
download
time

900 ms gain!

With PoPs

Total = 1100ms



How are users assigned to PoPs?

Through DNS:

IP handed based on user’s resolver country

```
# California
$ dig +short www.linkedin.com
216.52.242.80
```

```
# Spain
$ dig @109.69.8.51 +short www.linkedin.com
91.225.248.80
```

Should India connect to Singapore or Dublin?

How to assure optimal PoPs assignment?

RUM beacons

Fetch a tiny object from each candidate PoP

```
For each pop_name,
1.  Start timer
2.  Fetch {pop_name}.perf.linkedin.com/pop/admin
3.  Stop timer
Send data back to our servers
```

- Millions of agents!
- Analyze data to find “optimal” PoP per country

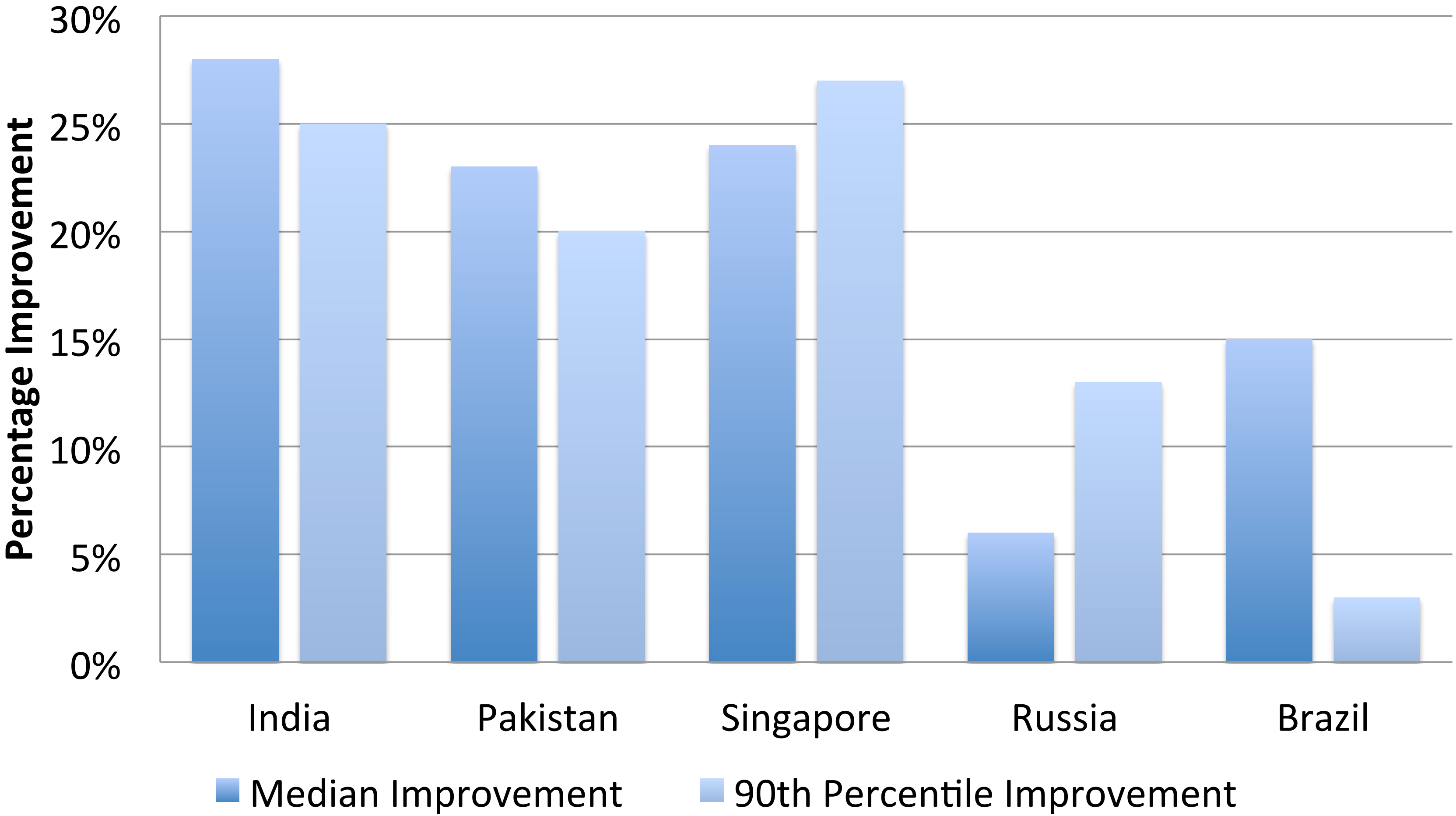
We can assign countries to new PoPs!

Country	PoP	Median Beacon Time(ms)
China	Hong Kong	434
China	Dublin	1216
China	Singapore	515
India	Hong Kong	1368
India	Dublin	1042
India	Singapore	898

We can audit current assignment!

Country	Is PoP optimal?	Current PoP	Optimal PoP
India	TRUE	Singapore	Singapore
Pakistan	FALSE	Singapore	Dublin
Spain	TRUE	Dublin	Dublin
Brazil	FALSE	US West Coast	US East Coast
Netherlands	TRUE	Dublin	Dublin
UAE	FALSE	US West Coast	Dublin
Italy	TRUE	Dublin	Dublin
Mexico	TRUE	US West Coast	US West Coast
Russia	FALSE	US West Coast	Dublin

LinkedIn Homepage Download Time Improvement



Success.

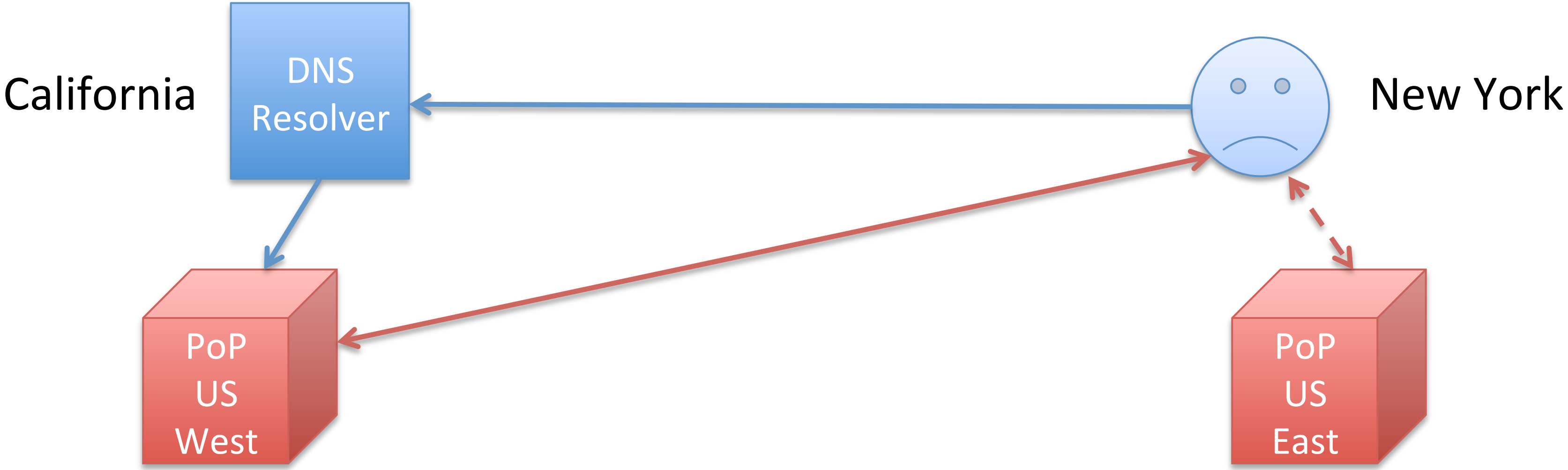
4GIFs.com

Plot Twist:
Assignment far from optimal

- About 31% of US traffic gets assigned to a suboptimal PoP.
  - 45% of East Coast
- About 10% of traffic globally gets assigned to a suboptimal PoP.

DNS PoP assignment is suboptimal

- Assignment based on Resolver IP, not Client IP



DNS PoP assignment is suboptimal

- Assignment based on Resolver IP, not Client IP

- Bad *IP to Geo* databases
  - Resolver really in NY, but database says CA

Story so far

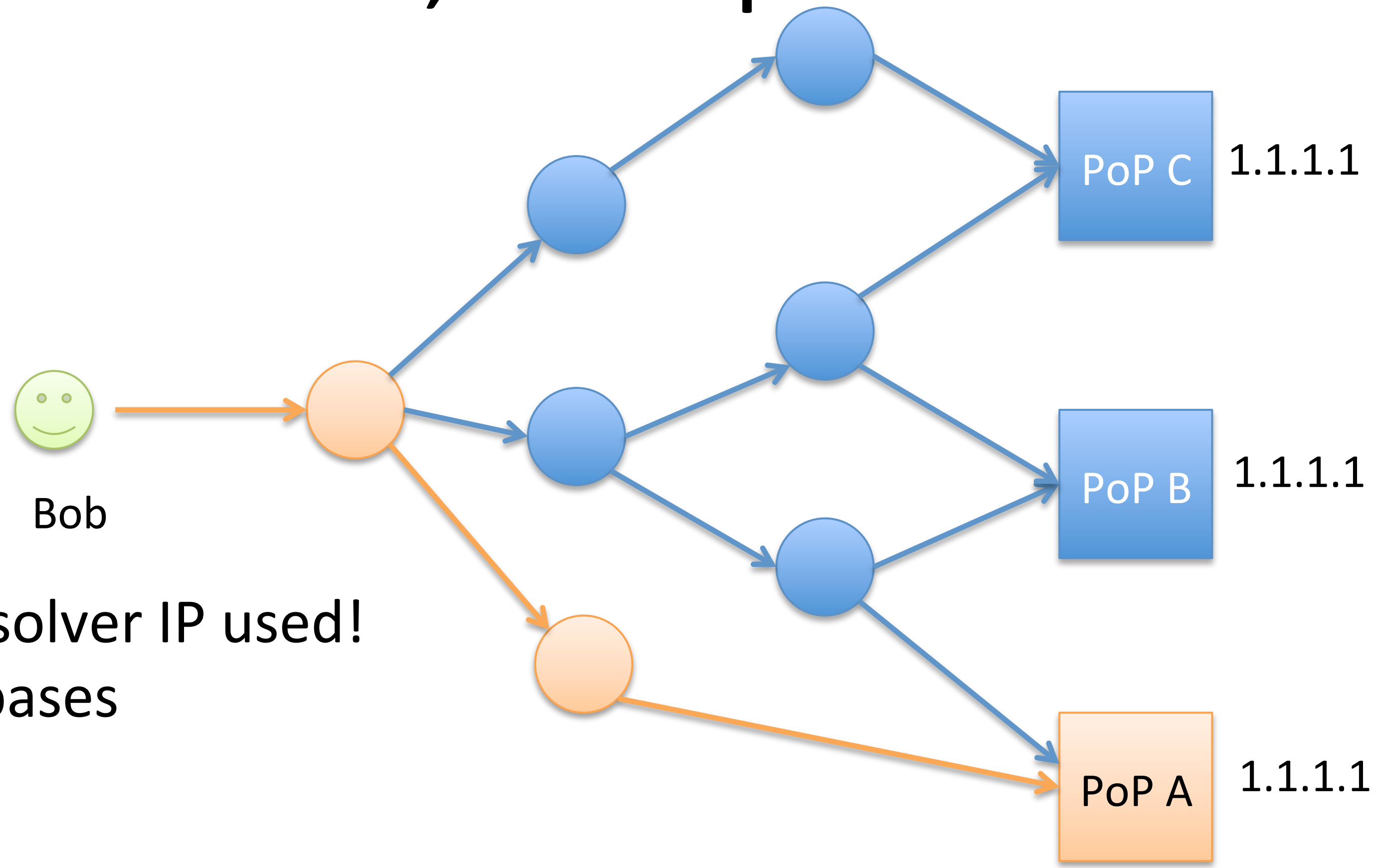
1. We built PoPs
2. …used RUM to assign users to Optimal PoPs
3. …found DNS based assignment is suboptimal

Accurate PoP assignment Problem

- Bug our DNS providers (31% -> 27%)
- Run our own DNS

How about ***Anycast***?

Anycast – One IP, Multiple Servers



✓ Client IP, not Resolver IP used!

✓ No Geo-IP Databases

How does Anycast compare to DNS?

Will anycast send more users to optimal PoP?

➢ Lets test it!

RUM to rescue

For each PoP:

1. Announce same anycast IP (108.174.13.10)
2. Configure a domain `ac.perf.linkedin.com` to point to 108.174.13.10

RUM to rescue

For each page view:

1. RUM downloads a tiny object :
   `ac.perf.linkedin.com/pop/admin`
2. Read `X-Li-Pop` response header to record which PoP served the object
3. Send this back to LinkedIn with RUM data

Data:

1. For each user, the anycast PoP
2. For each user, the optimal PoP (from pop beacons)

Results ☺

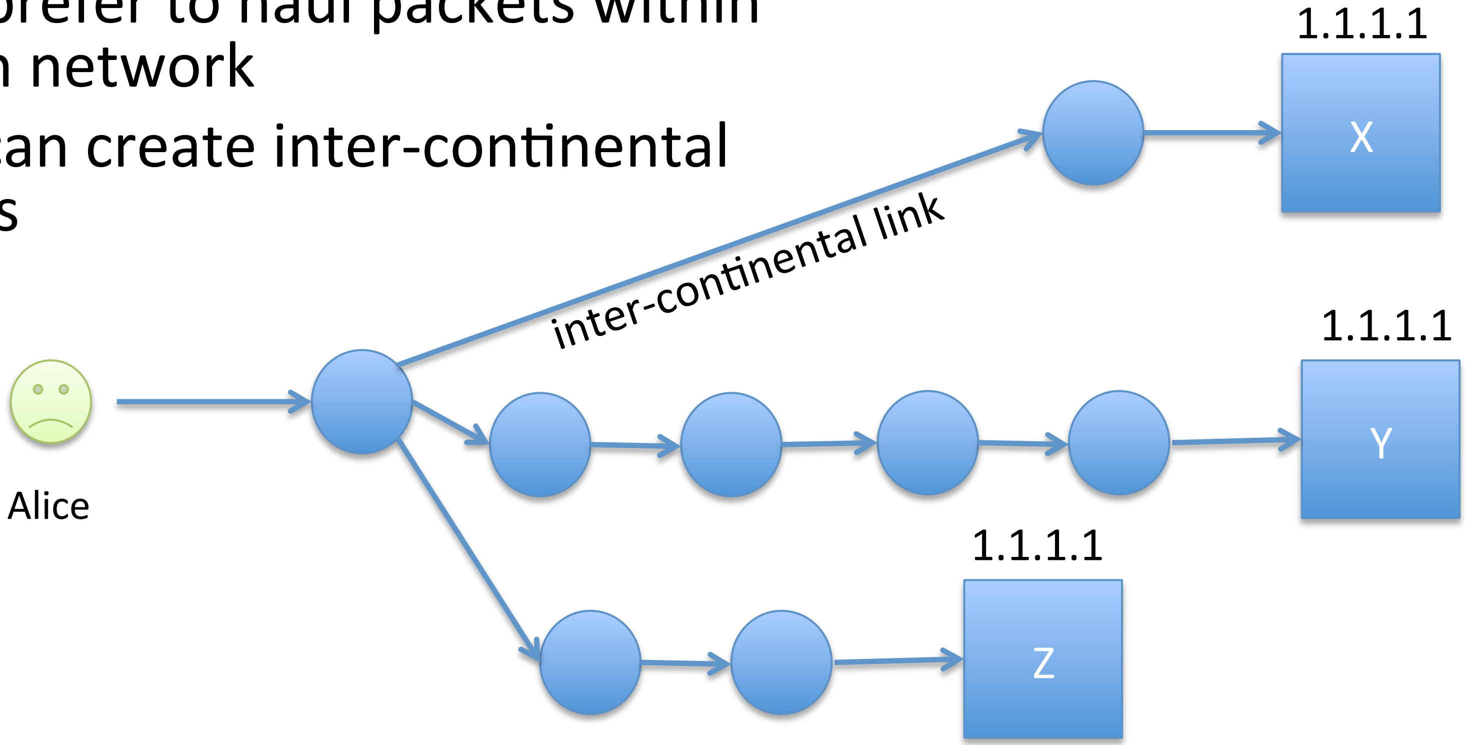
Region or Country	DNS % Optimal Assignment	Anycast % Optimal Assignment
Illinois	70	90
Florida	73	95
Georgia	75	93
Pennsylvania	85	95

Results ☹

Region or Country	DNS % Optimal Assignment	Anycast % Optimal Assignment
Arizona	60	39
Brazil	88	33
New York	77	74

Fewer hops != Lower Latency

- Carriers prefer to haul packets within their own network
- Peering can create inter-continental short cuts



Maybe DNS wasn’t so bad

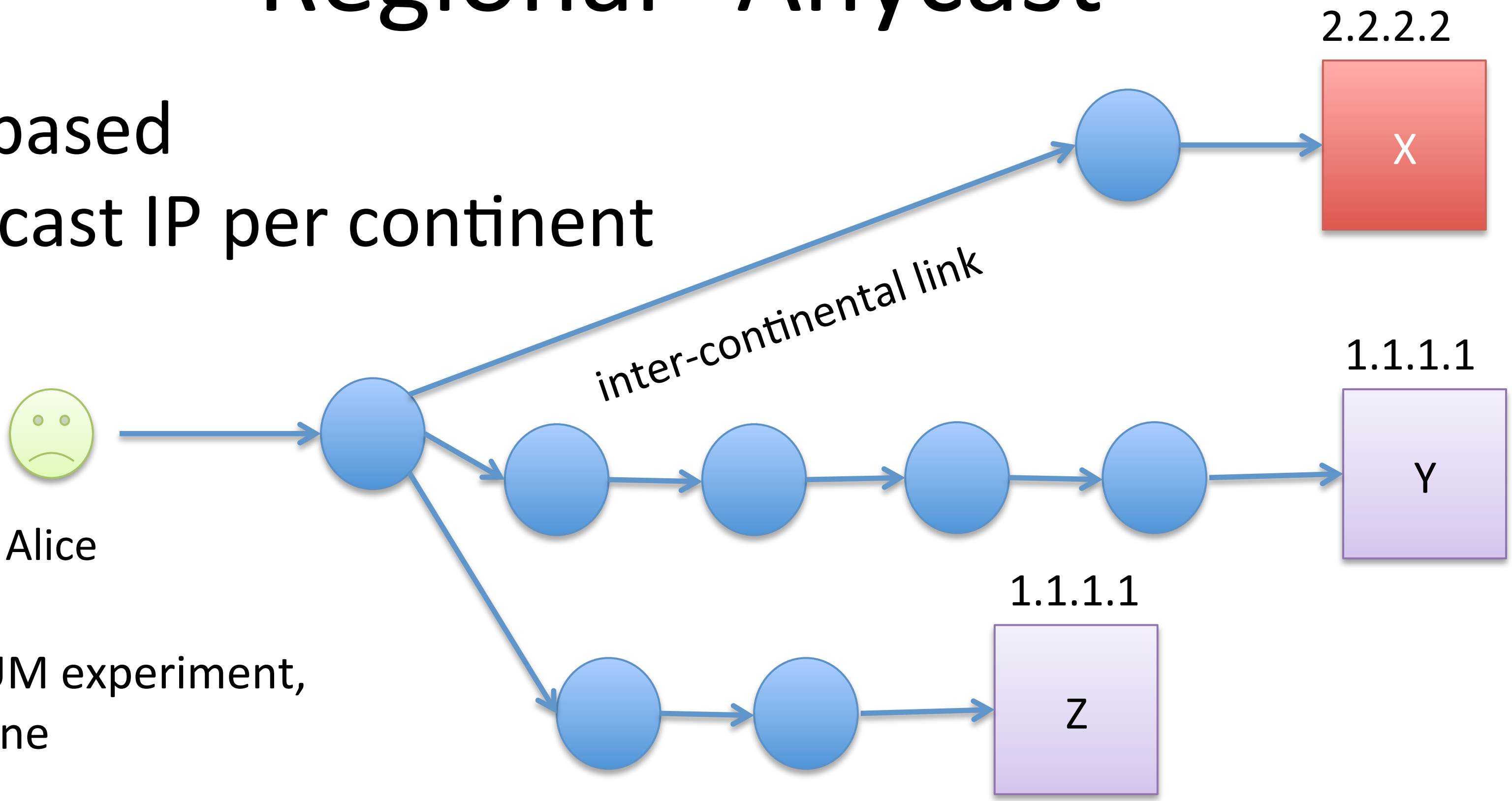
Continent-level assignments 👍

City / State level assignments 👎

“Regional” Anycast

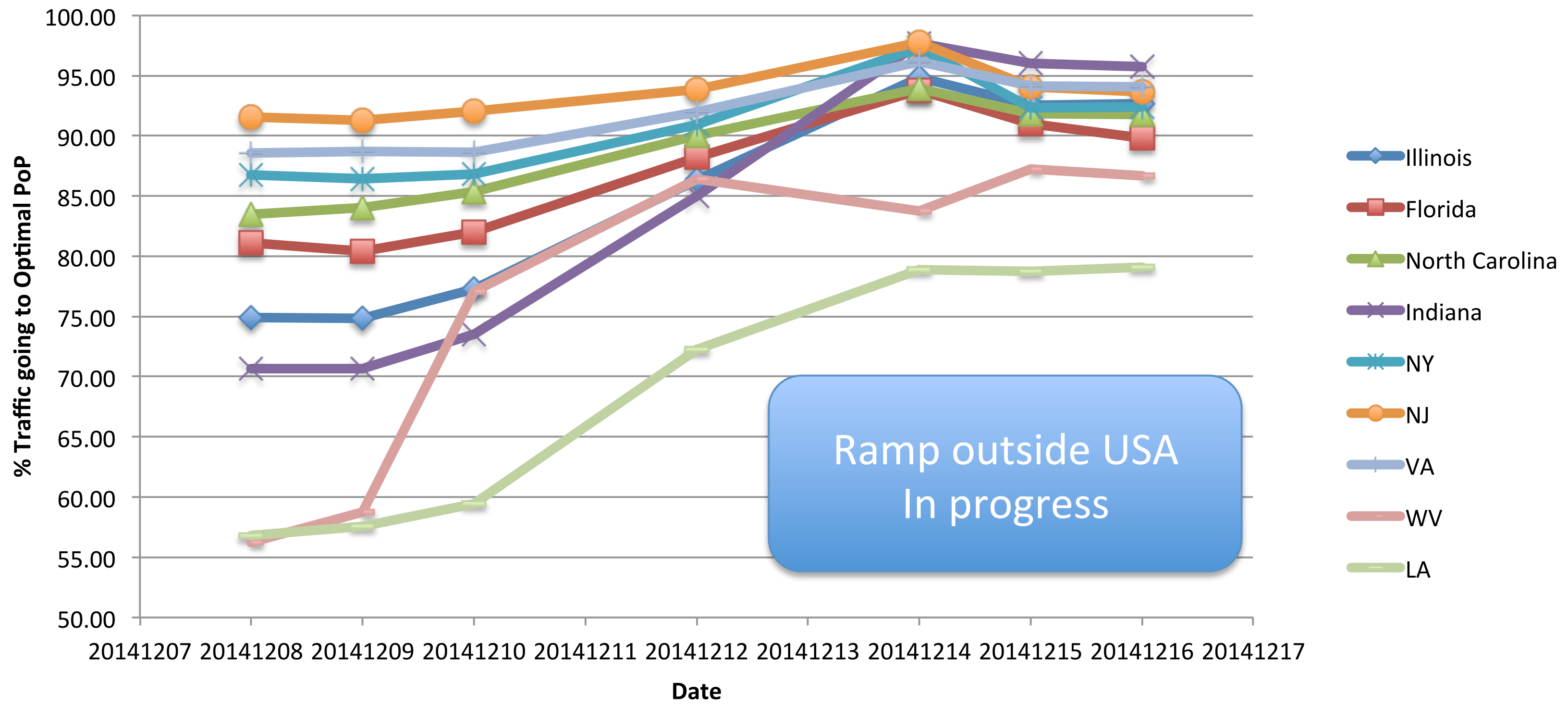
DNS-based

1 anycast IP per continent



Ran a RUM experiment, all was fine

USA Ramp Results



Ramp outside USA
In progress

Story so far

1. We built PoPs
2. …used RUM to assign users to Optimal PoPs
3. …found DNS based assignment is suboptimal
4. …evaluated Anycast as a solution using RUM
5. …now using Anycast to assign users to PoPs

Next play:

- Build more PoPs!

Story: The End

Learnings

- Clients are your measurement agents
- Trust, but verify
- You can have a bigger impact if you collaborate

Next Play

- Keep evaluating Anycast
- Keep building new PoPs