Columbia University Community Measurement Resources: Routing Experiments for the Cloud Era with the PEERING Testbed & A New Dataset of Packet Captures from a Residential Network

Ethan Katz-Bassett

Ítalo Cunha

Ezri Zhu

Jiangchen Zhu

Tom Koch

Gil Zussman

Shuyue Yu

Ilgar Mammadov

Carson Garland

Columbia University

UFMG

EzriCloud (AS206628)

Updates on two community resources — please use them!

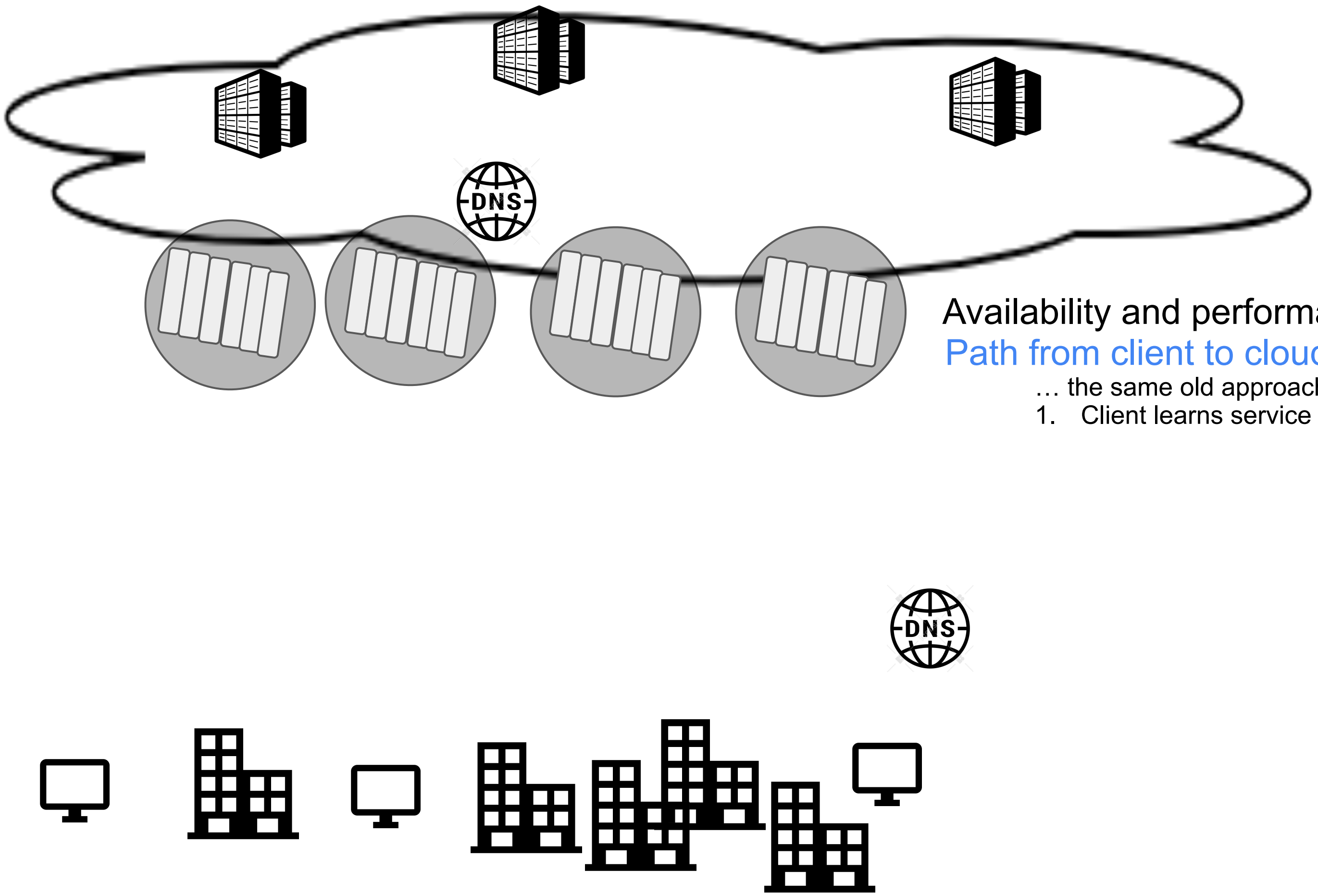
PEERING BGP testbed

- Exchange BGP routes and traffic with thousands of ASes at locations around the world

Residential traffic traces

- Packet traces from ~1000 residences
    - Plan to scale to 8000 units, 24x7

DNS + BGP determine path from client to cloud



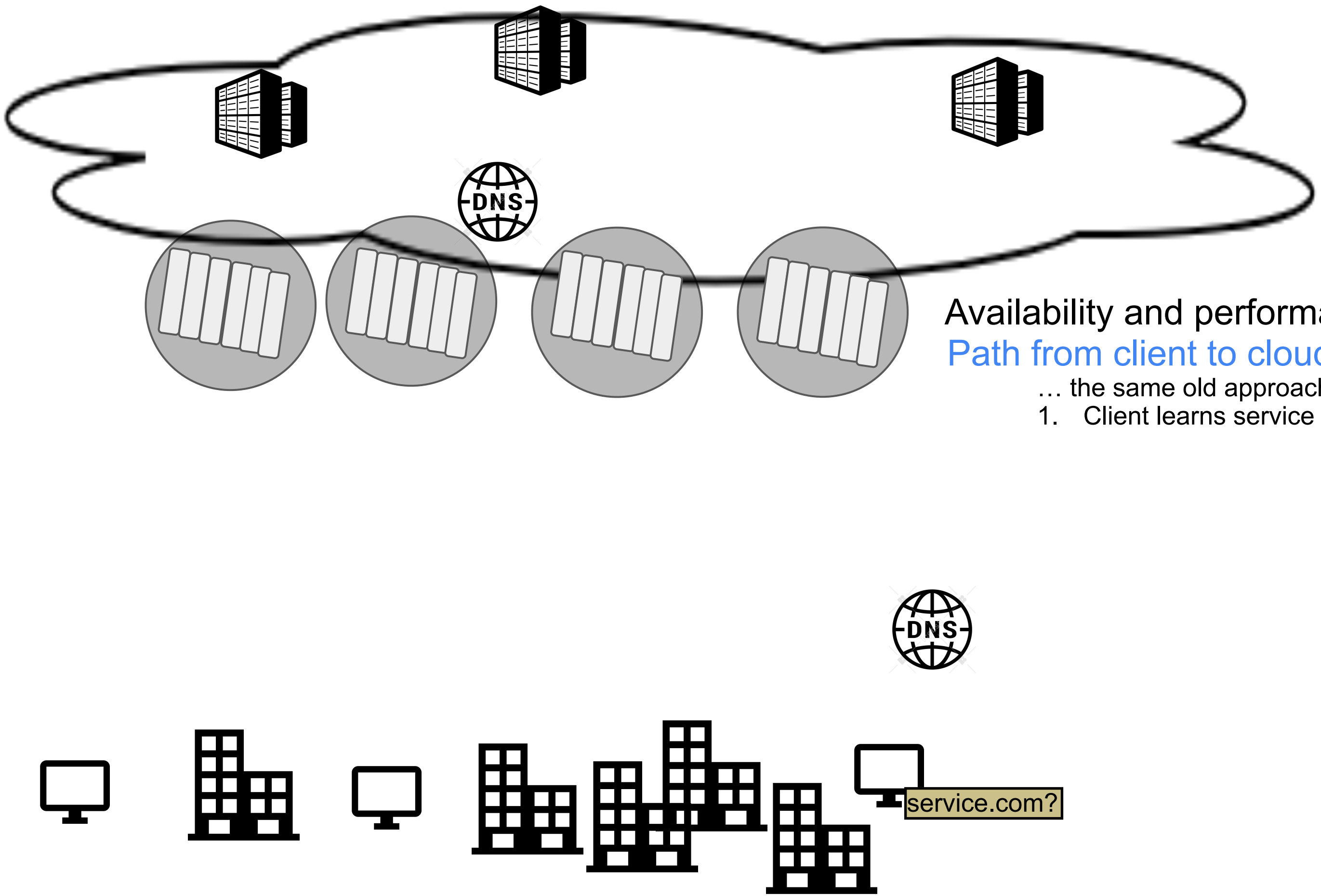
Availability and performance determined by paths:

Path from client to cloud ingress

… the same old approaches for decades:

1. Client learns service IP address from DNS

DNS + BGP determine path from client to cloud



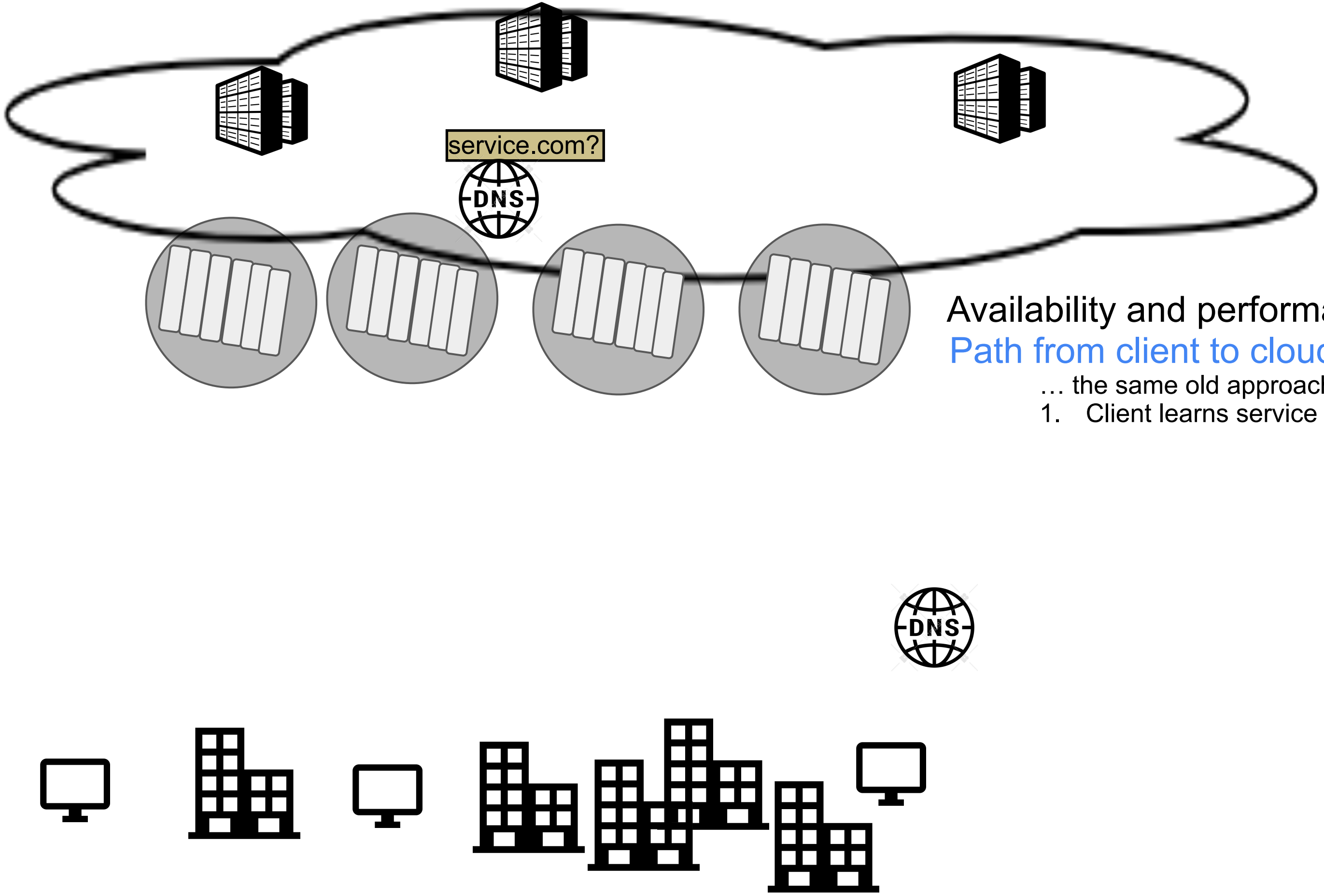
Availability and performance determined by paths:

Path from client to cloud ingress

… the same old approaches for decades:

1. Client learns service IP address from DNS

DNS + BGP determine path from client to cloud



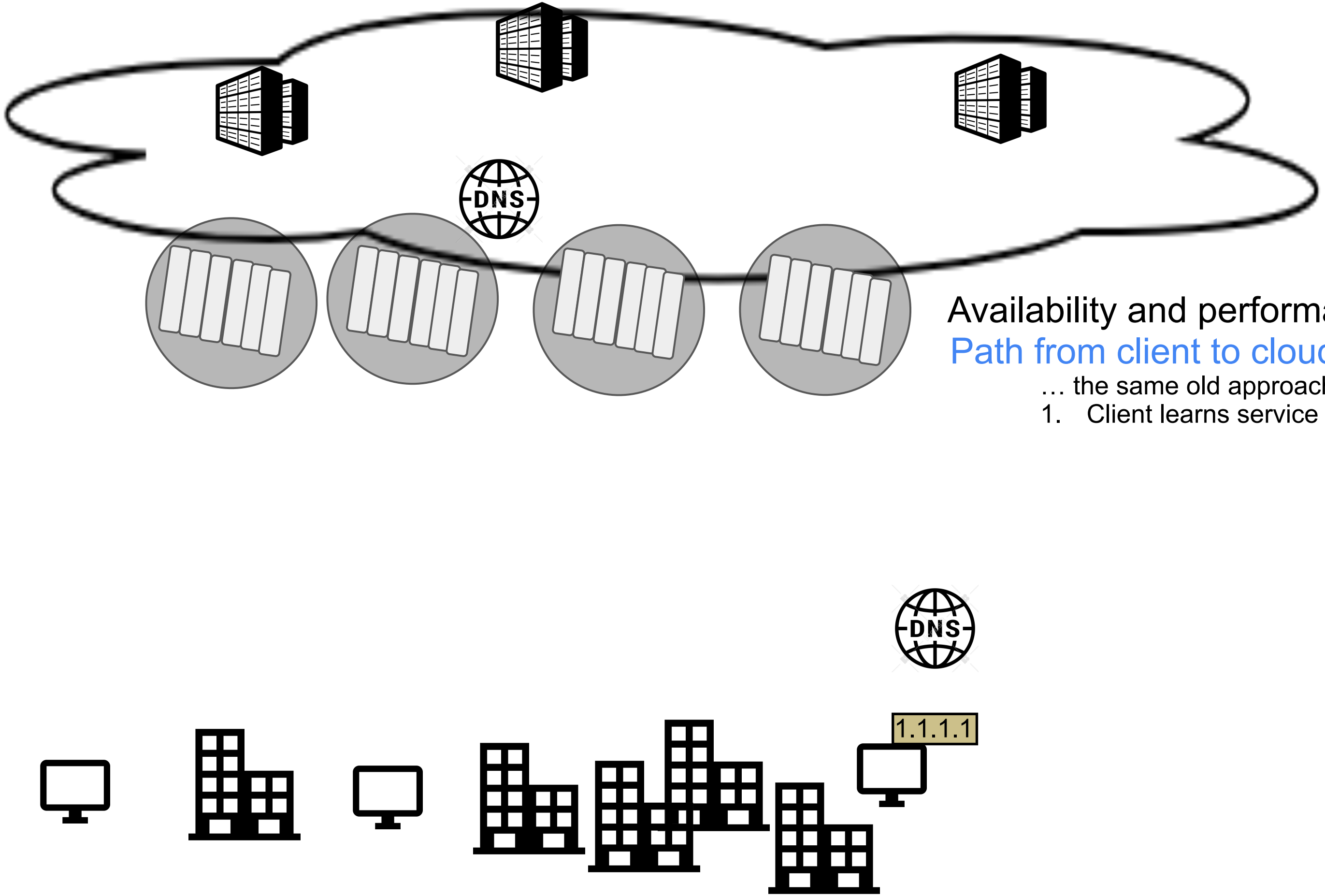
Availability and performance determined by paths:

Path from client to cloud ingress

… the same old approaches for decades:

1. Client learns service IP address from DNS

DNS + BGP determine path from client to cloud



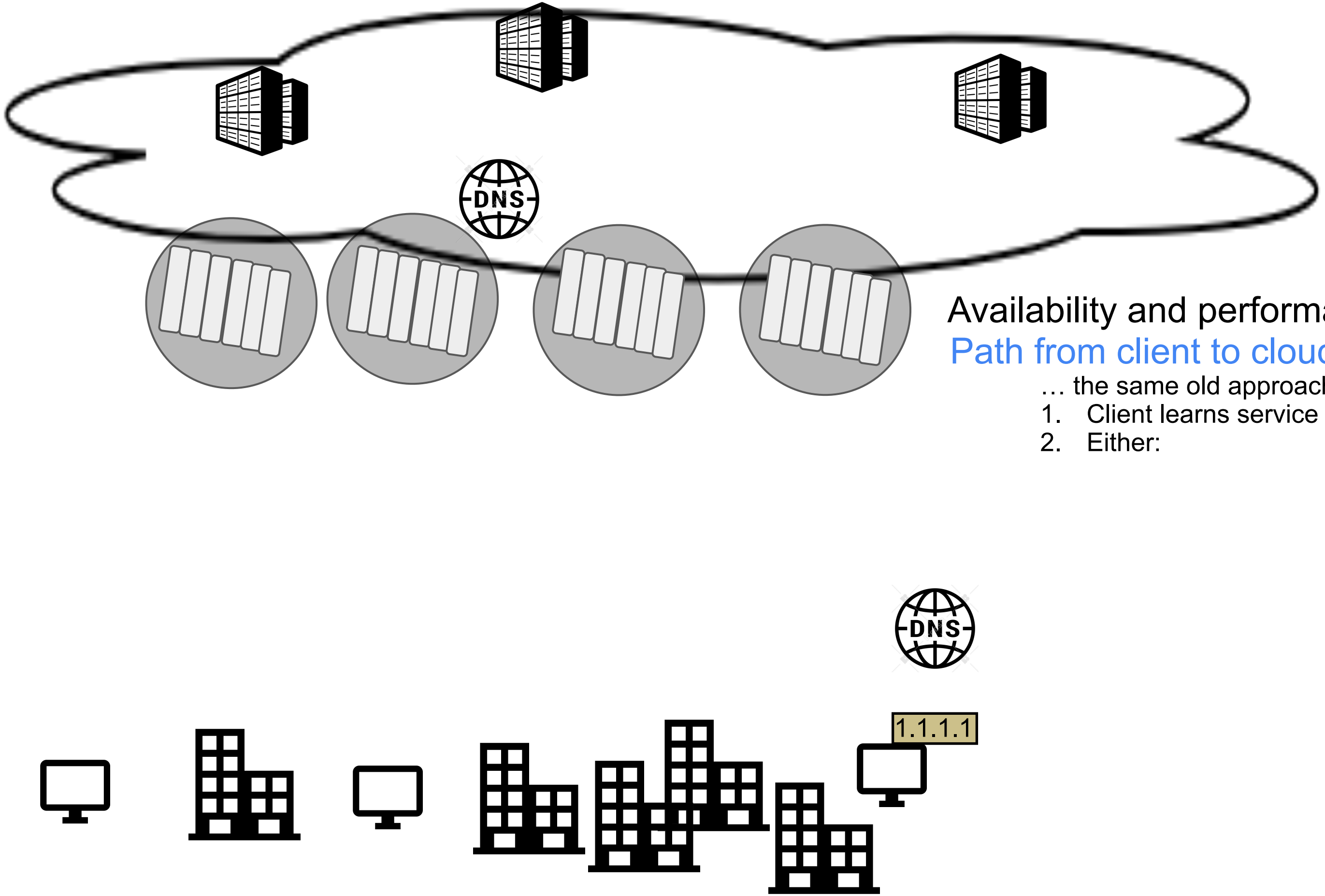
Availability and performance determined by paths:

Path from client to cloud ingress

… the same old approaches for decades:

1. Client learns service IP address from DNS

DNS + BGP determine path from client to cloud



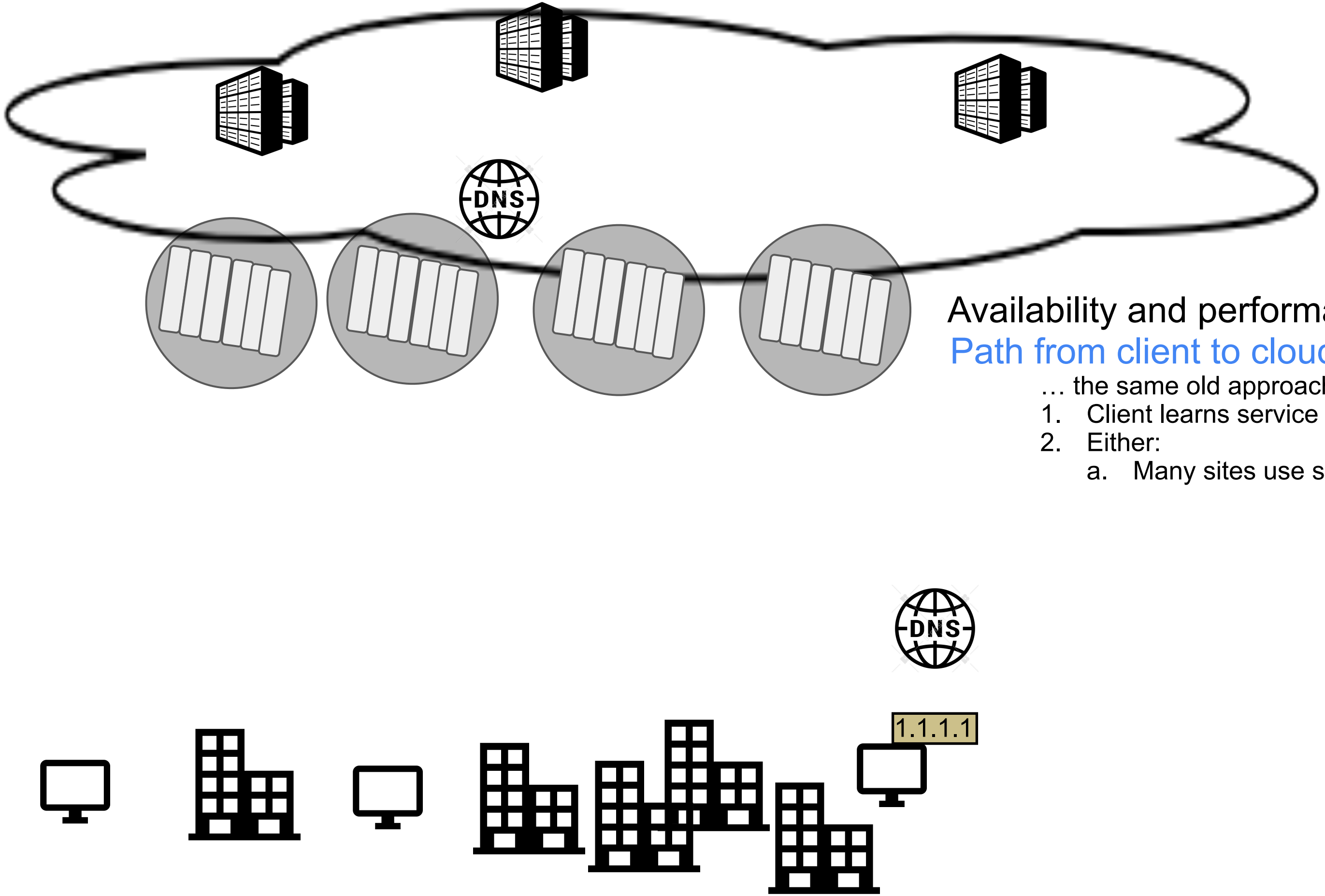
Availability and performance determined by paths:

Path from client to cloud ingress

… the same old approaches for decades:

1. Client learns service IP address from DNS
2. Either:

DNS + BGP determine path from client to cloud



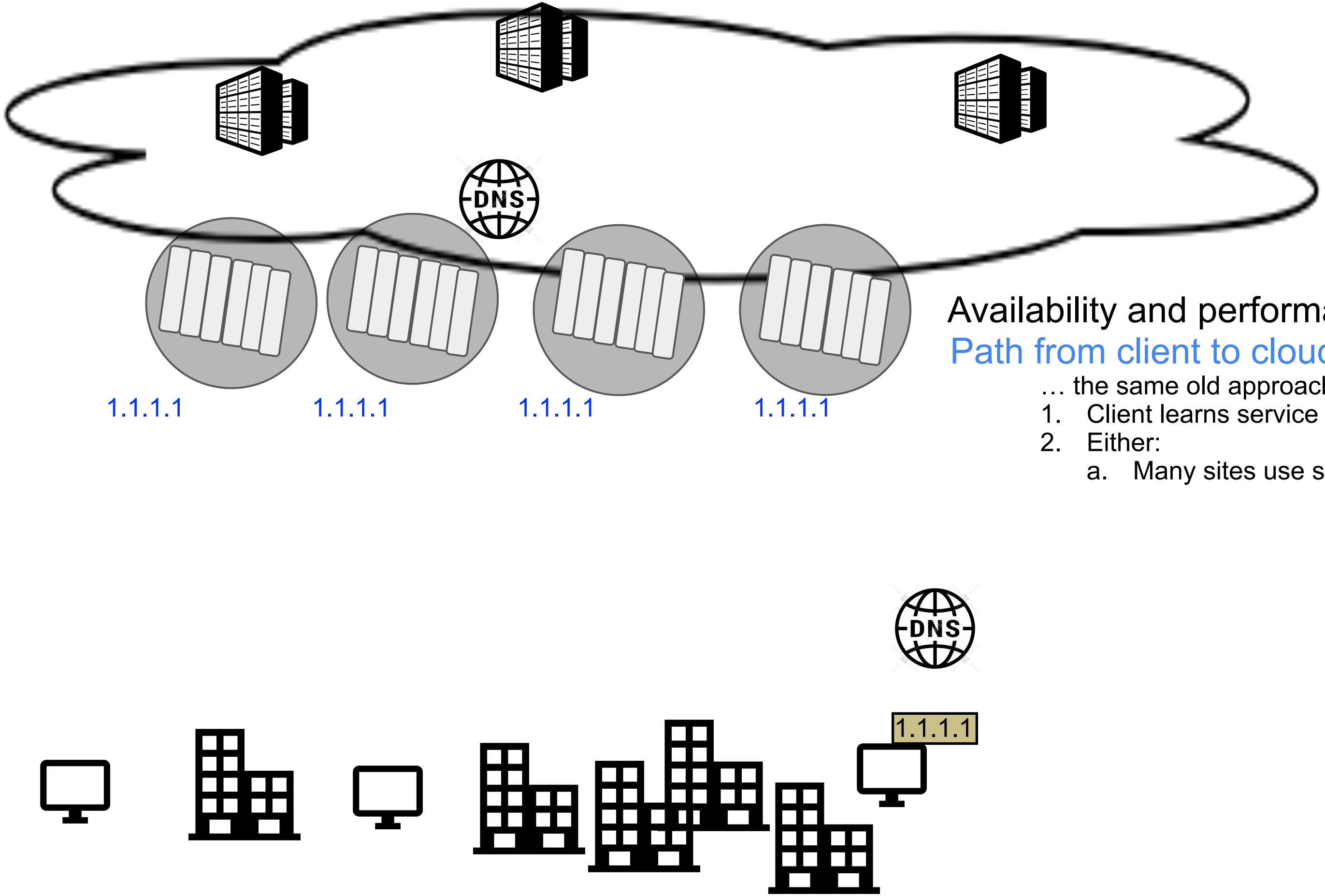
Availability and performance determined by paths:

Path from client to cloud ingress

… the same old approaches for decades:

1. Client learns service IP address from DNS
2. Either:
   a. Many sites use same address (anycast),

DNS + BGP determine path from client to cloud



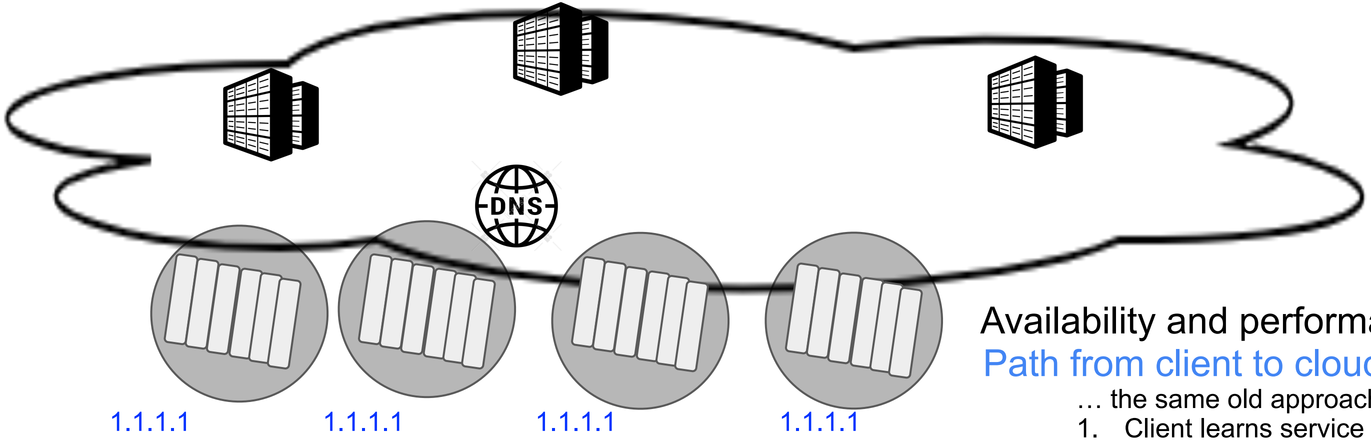
Availability and performance determined by paths:

Path from client to cloud ingress

… the same old approaches for decades:

1. Client learns service IP address from DNS
2. Either:
    a. Many sites use same address (anycast),

DNS + BGP determine path from client to cloud

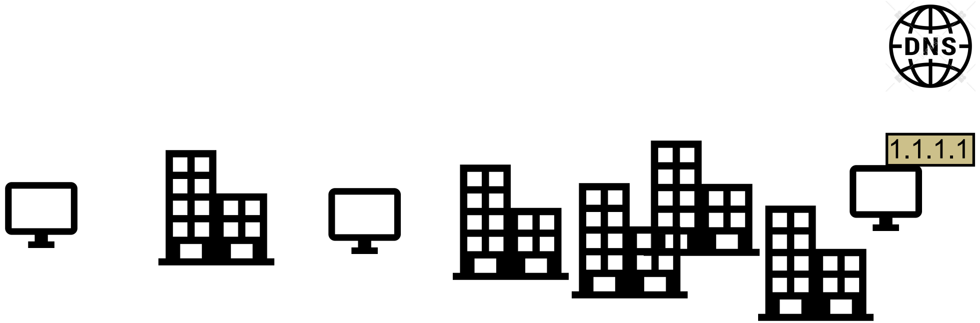


Availability and performance determined by paths:

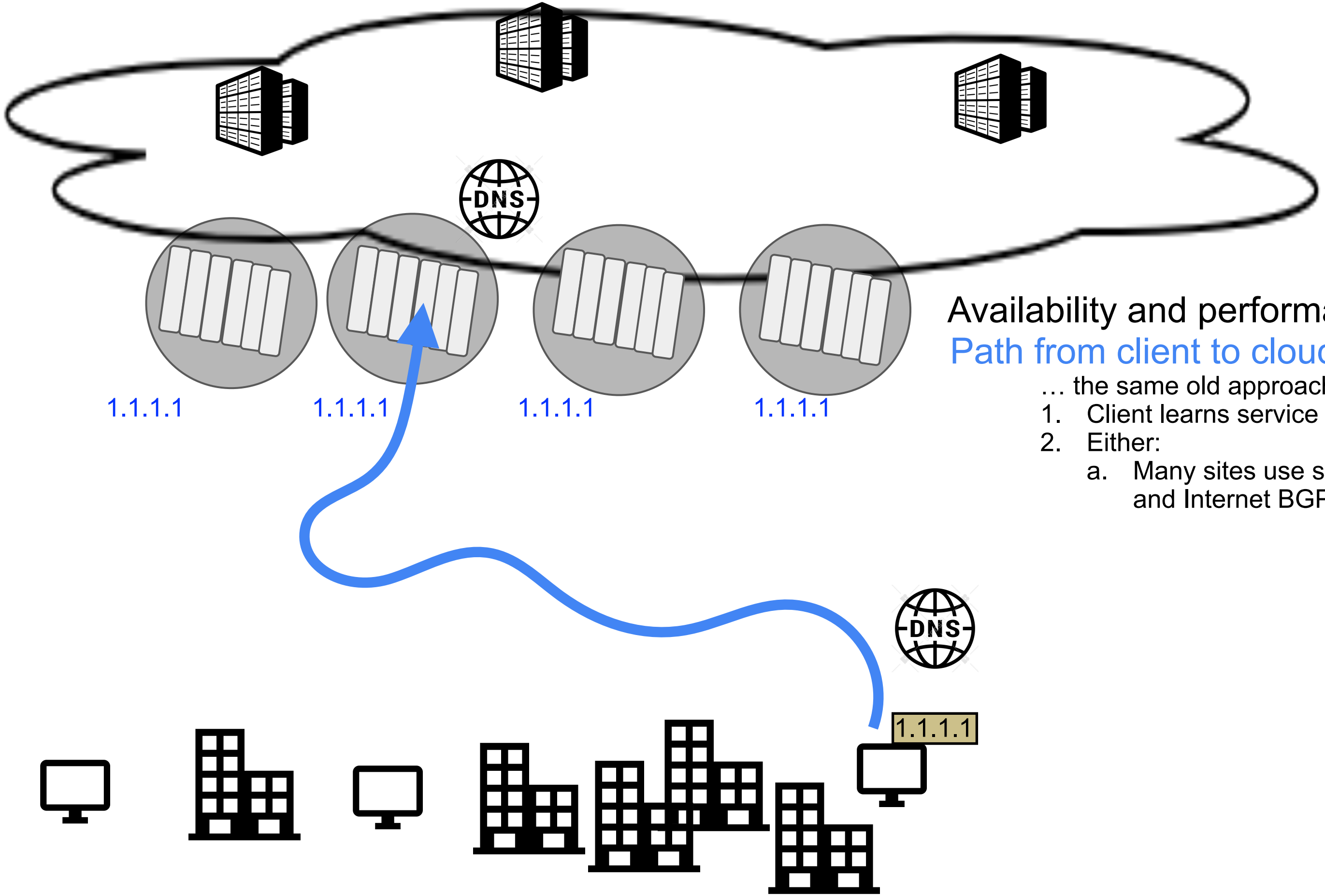
Path from client to cloud ingress

… the same old approaches for decades:

1. Client learns service IP address from DNS
2. Either:
   a. Many sites use same address (anycast),
      and Internet BGP routing picks both site and route to it



DNS + BGP determine path from client to cloud



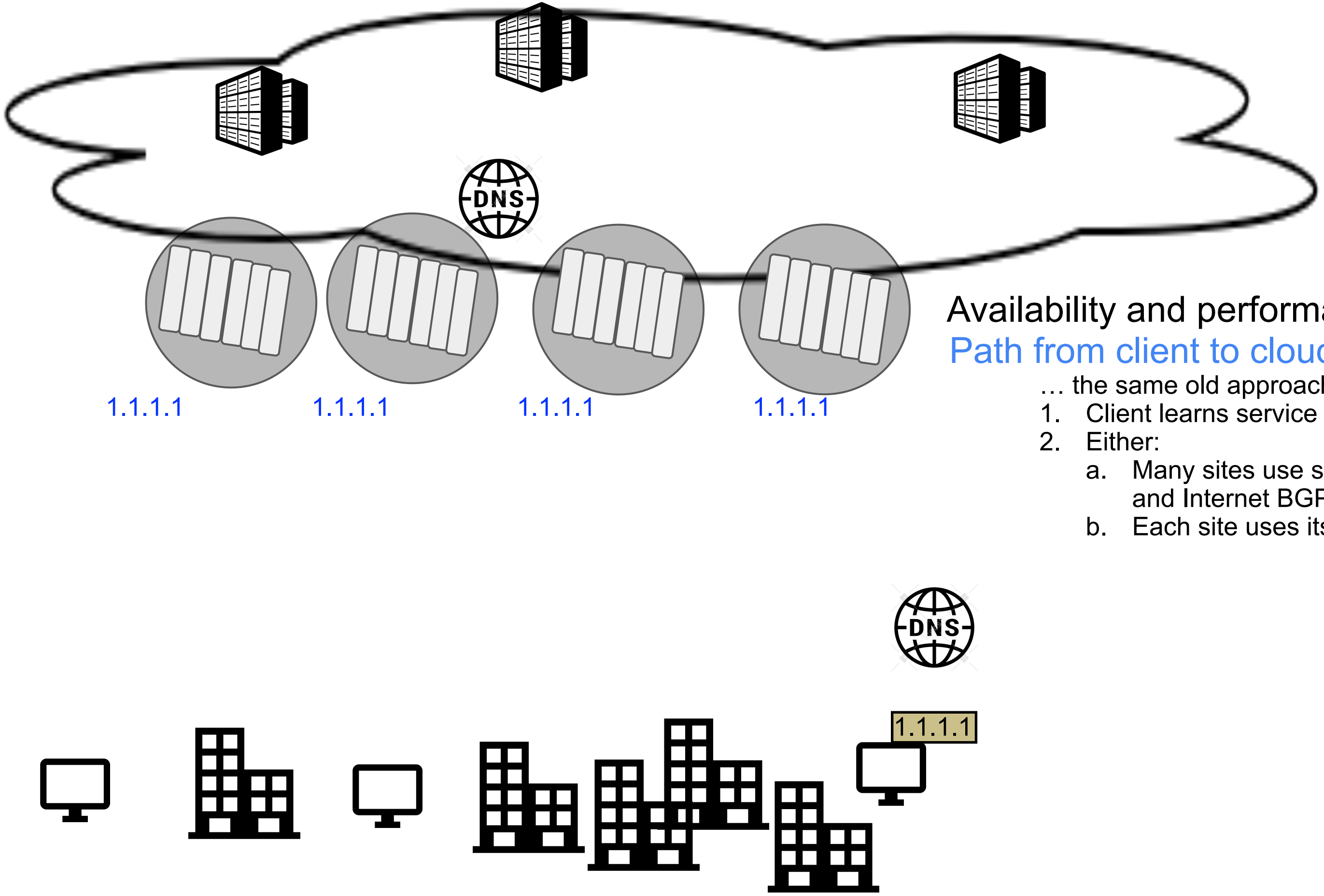
Availability and performance determined by paths:

Path from client to cloud ingress

… the same old approaches for decades:

1. Client learns service IP address from DNS
2. Either:
   a. Many sites use same address (anycast), and Internet BGP routing picks both site and route to it

DNS + BGP determine path from client to cloud



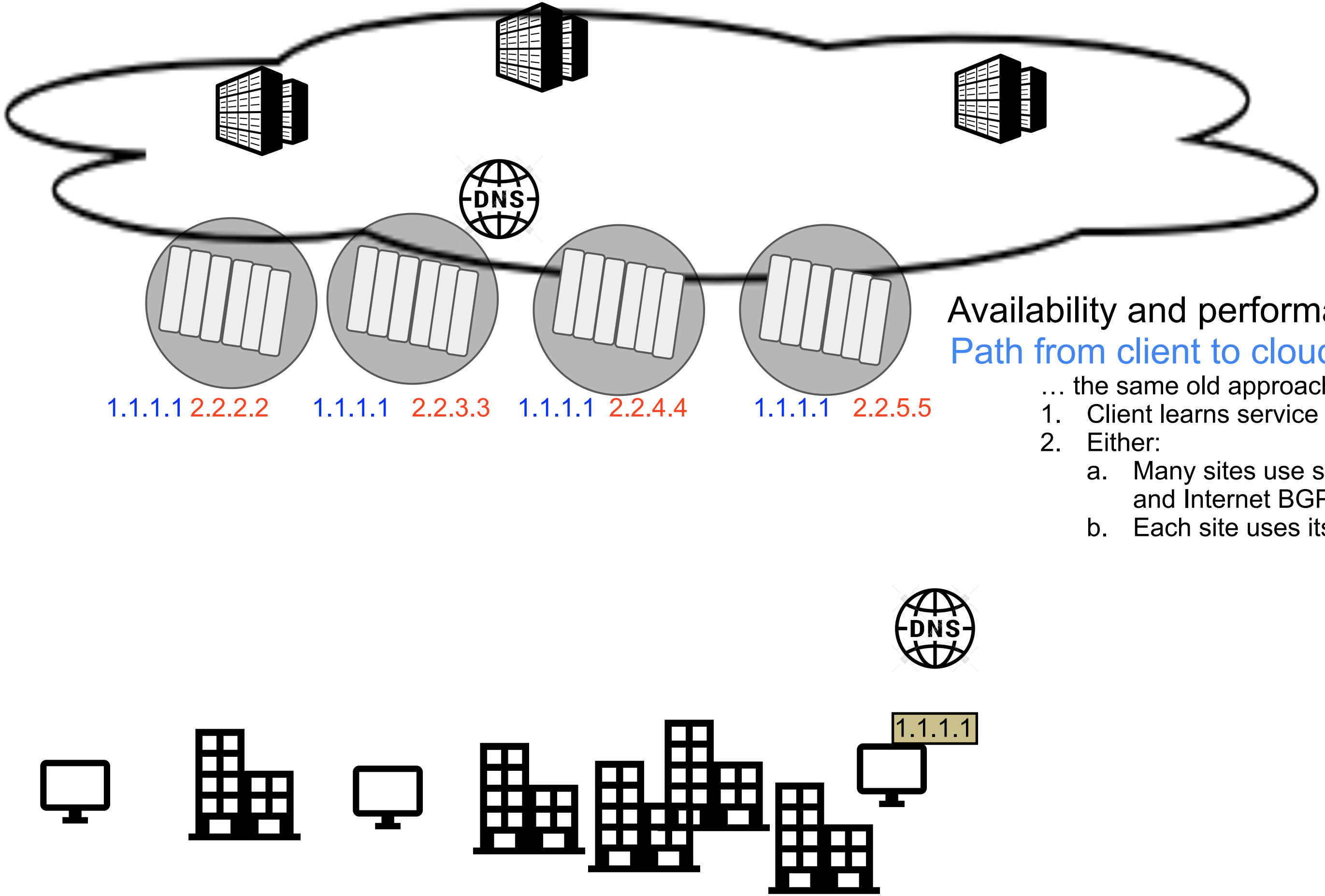
Availability and performance determined by paths:

Path from client to cloud ingress

… the same old approaches for decades:

1. Client learns service IP address from DNS
2. Either:
   a. Many sites use same address (anycast), and Internet BGP routing picks both site and route to it
   b. Each site uses its own address (unicast),

DNS + BGP determine path from client to cloud



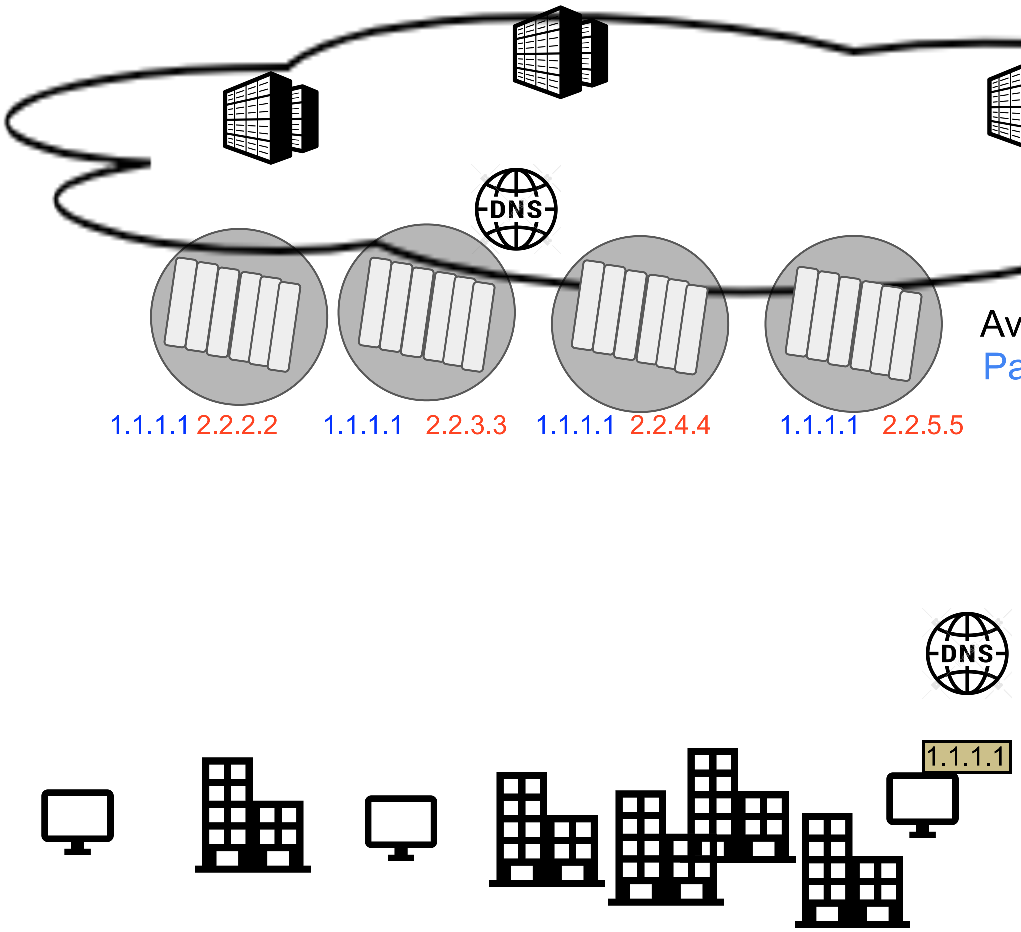
Availability and performance determined by paths:

Path from client to cloud ingress

… the same old approaches for decades:

1. Client learns service IP address from DNS
2. Either:
    a. Many sites use same address (anycast), and Internet BGP routing picks both site and route to it
    b. Each site uses its own address (unicast),

DNS + BGP determine path from client to cloud



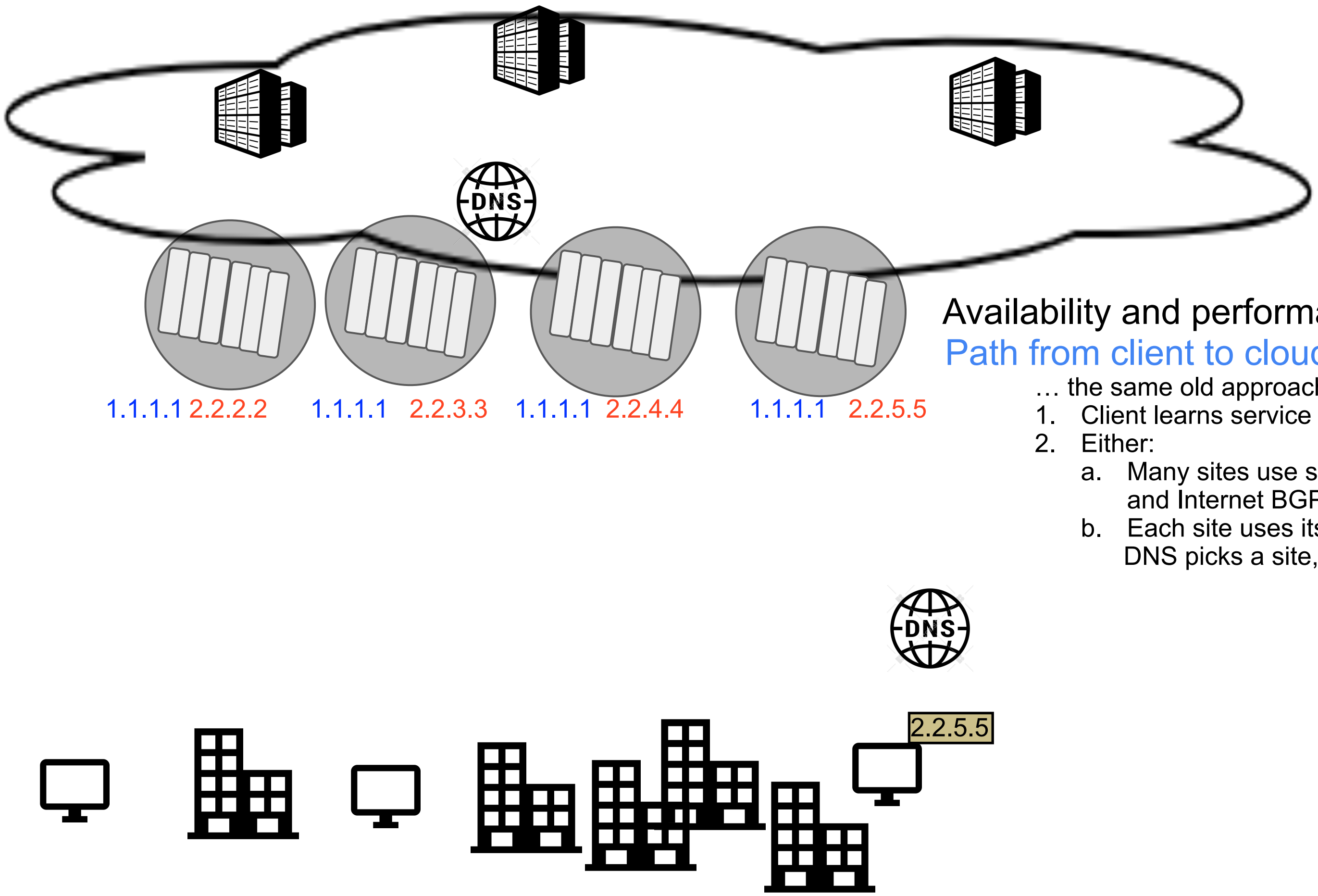
Availability and performance determined by paths:

Path from client to cloud ingress

… the same old approaches for decades:

1. Client learns service IP address from DNS
2. Either:
   a. Many sites use same address (anycast), and Internet BGP routing picks both site and route to it
   b. Each site uses its own address (unicast), DNS picks a site, and BGP picks a route to it

DNS + BGP determine path from client to cloud



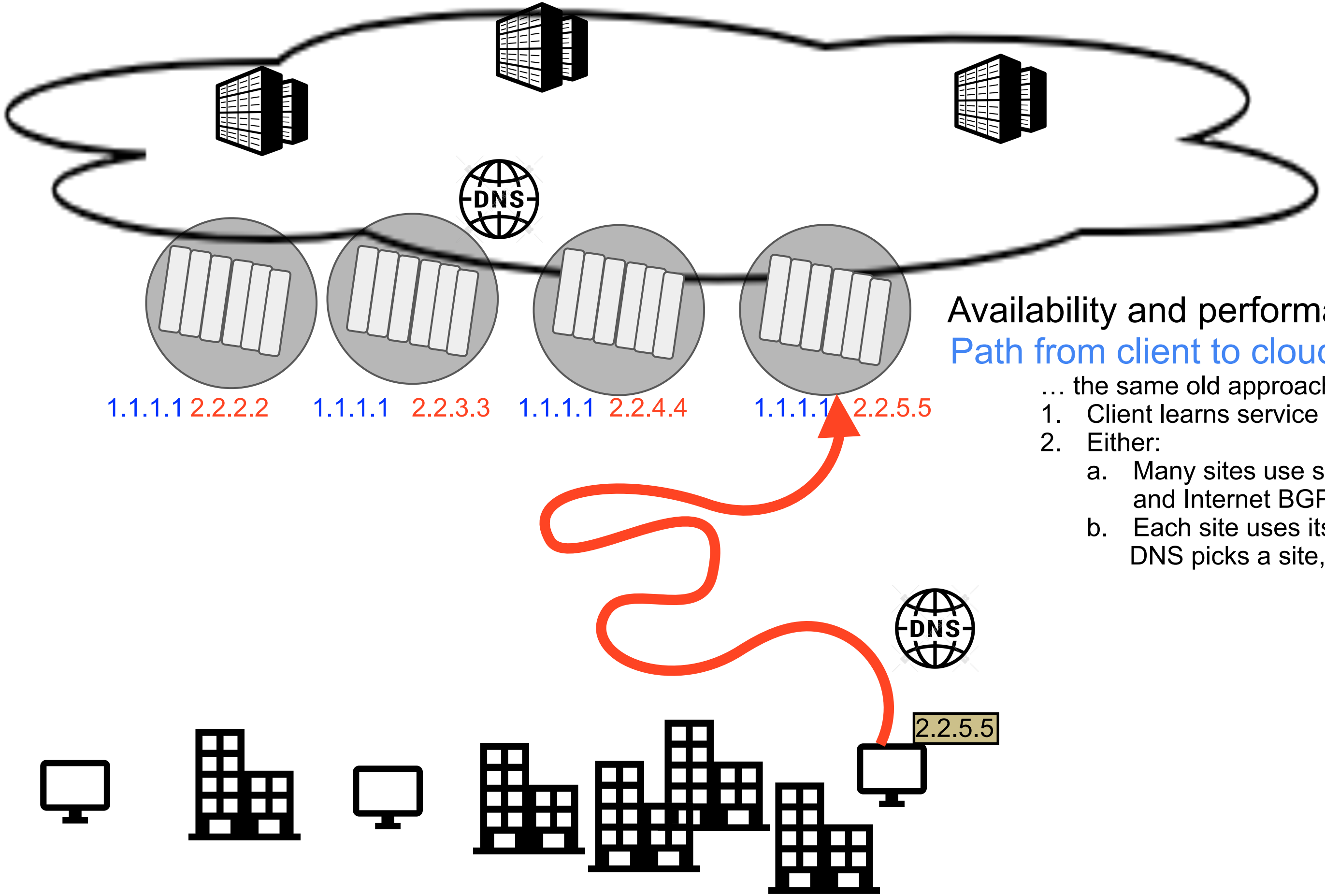
Availability and performance determined by paths:

Path from client to cloud ingress

… the same old approaches for decades:

1. Client learns service IP address from DNS
2. Either:
   a. Many sites use same address (anycast), and Internet BGP routing picks both site and route to it
   b. Each site uses its own address (unicast), DNS picks a site, and BGP picks a route to it

DNS + BGP determine path from client to cloud



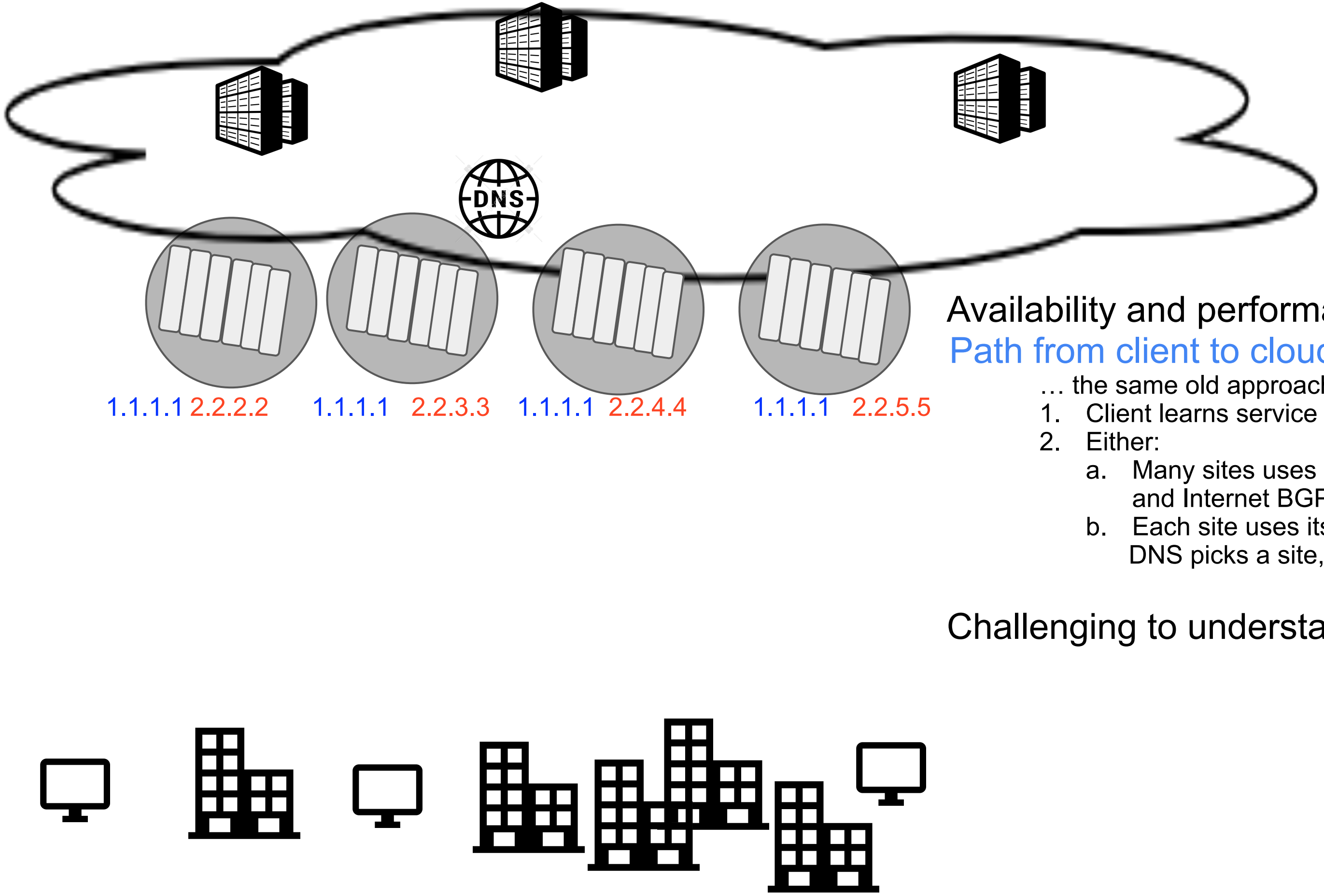
Availability and performance determined by paths:

Path from client to cloud ingress

… the same old approaches for decades:

1. Client learns service IP address from DNS
2. Either:
    a. Many sites use same address (anycast), and Internet BGP routing picks both site and route to it
    b. Each site uses its own address (unicast), DNS picks a site, and BGP picks a route to it

Challenges to improving paths from clients to cloud



Availability and performance determined by paths:

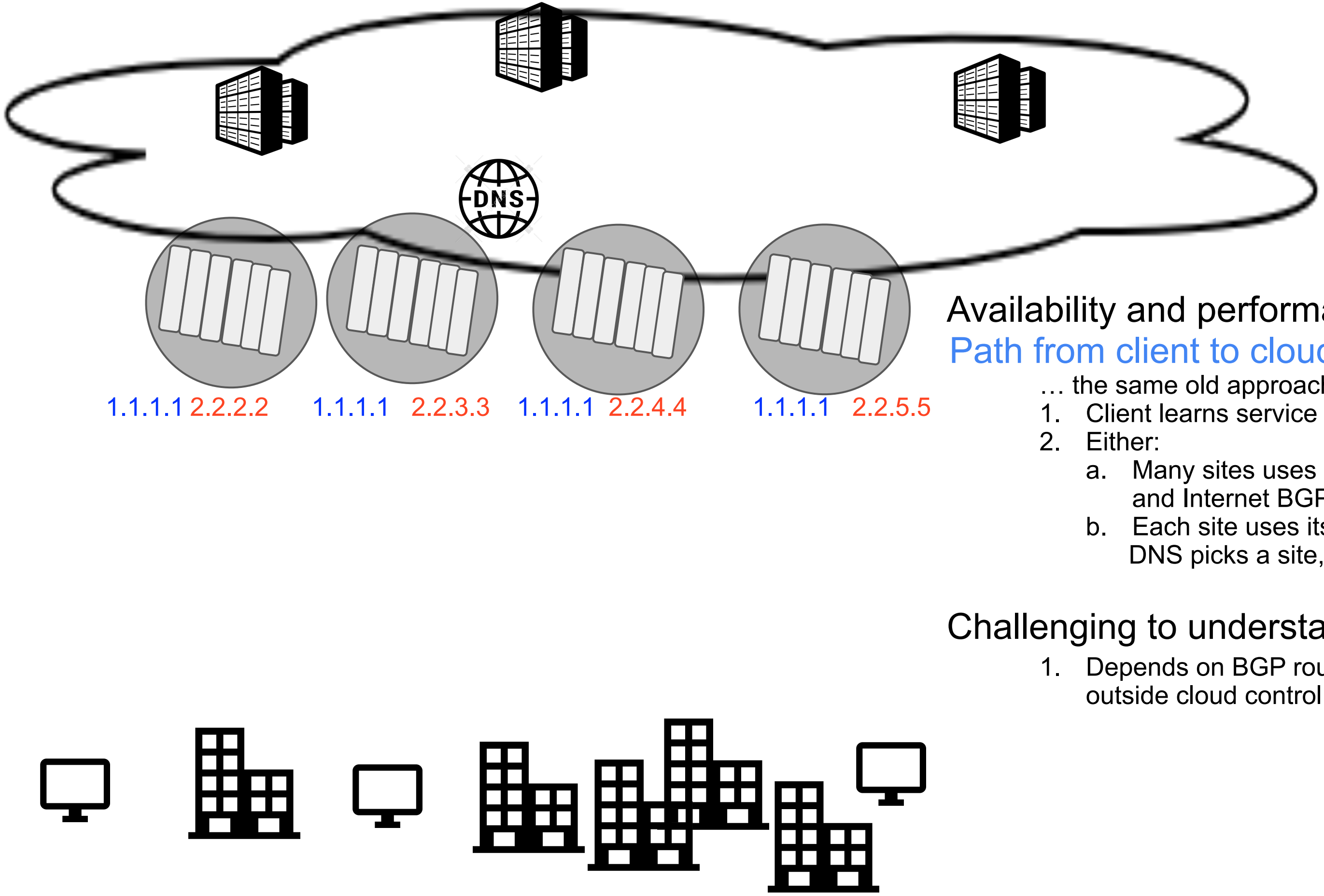
Path from client to cloud ingress

… the same old approaches for decades:

1. Client learns service IP address from DNS
2. Either:
   a. Many sites uses same address (anycast), and Internet BGP routing picks both site and route to it
   b. Each site uses its own address (unicast), DNS picks a site, and BGP picks a route to it

Challenging to understand:

Challenges to improving paths from clients to cloud



Availability and performance determined by paths:

Path from client to cloud ingress

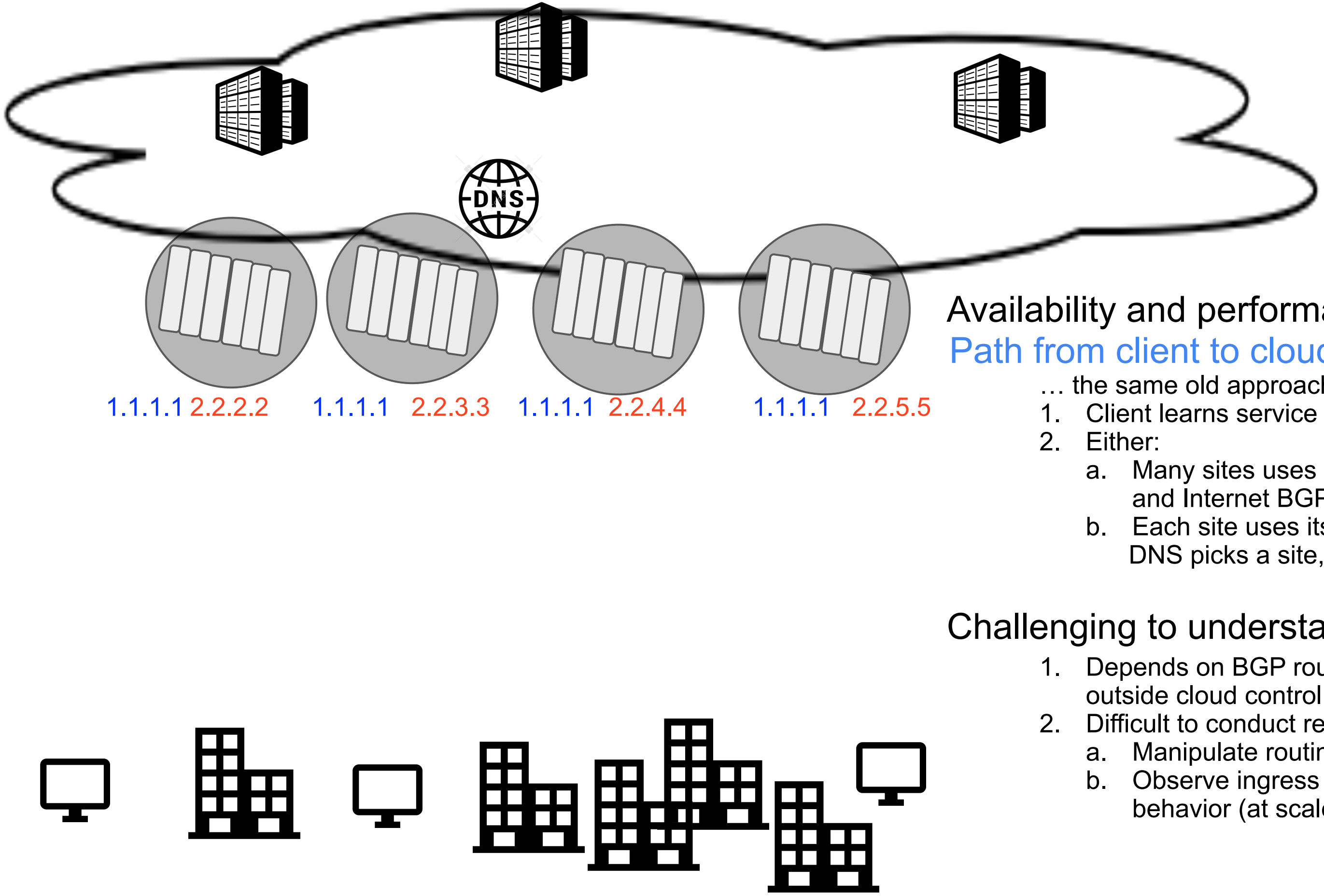
… the same old approaches for decades:

1. Client learns service IP address from DNS
2. Either:
    a. Many sites uses same address (anycast), and Internet BGP routing picks both site and route to it
    b. Each site uses its own address (unicast), DNS picks a site, and BGP picks a route to it

Challenging to understand:

1. Depends on BGP routing policy and DNS caching policy outside cloud control

Challenges to improving paths from clients to cloud



Availability and performance determined by paths:

Path from client to cloud ingress

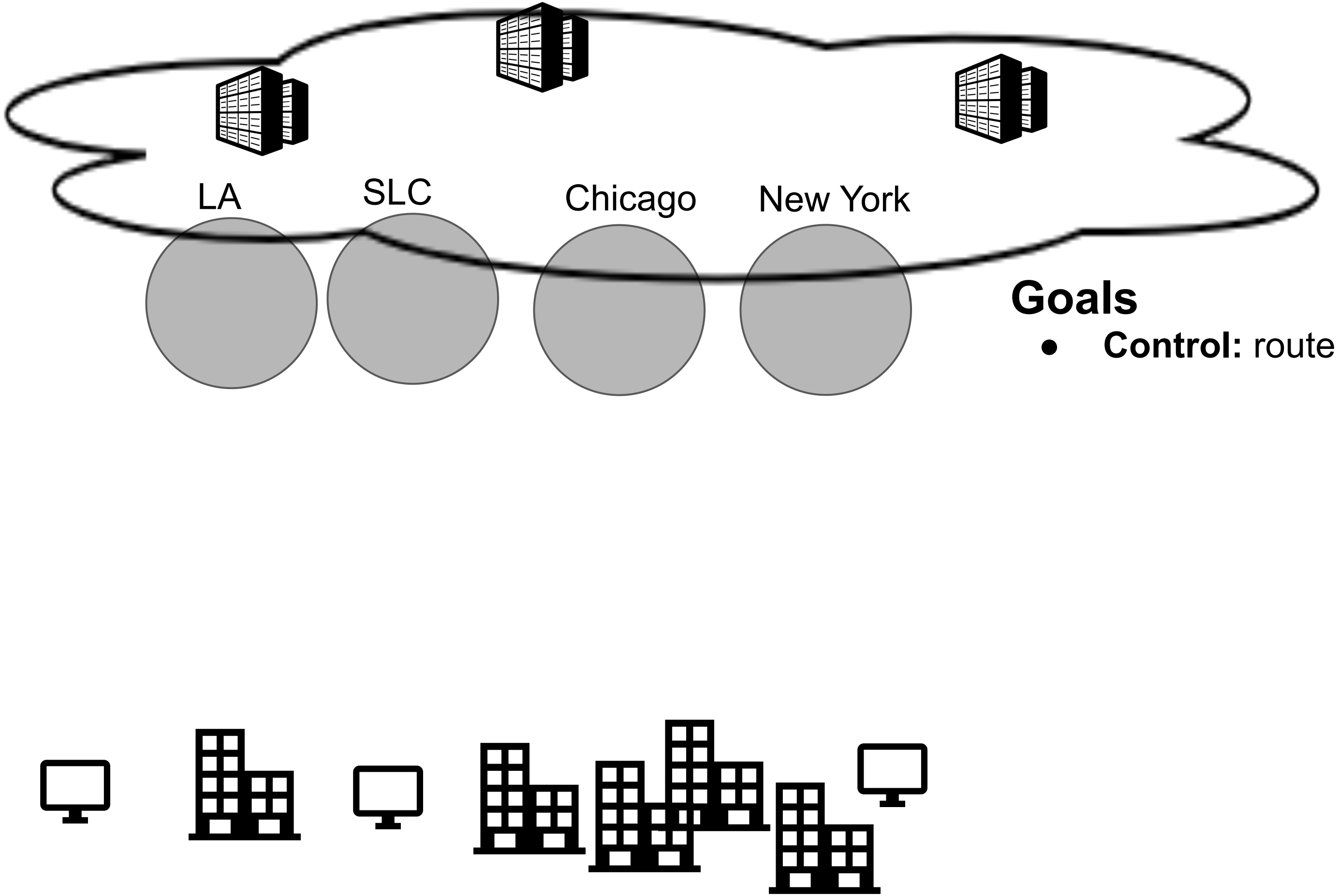
… the same old approaches for decades:

1. Client learns service IP address from DNS
2. Either:
   a. Many sites uses same address (anycast), and Internet BGP routing picks both site and route to it
   b. Each site uses its own address (unicast), DNS picks a site, and BGP picks a route to it

Challenging to understand:

1. Depends on BGP routing policy and DNS caching policy outside cloud control
2. Difficult to conduct research in academia:
   a. Manipulate routing (at cloud scale)
   b. Observe ingress routing decisions and DNS caching behavior (at scale)

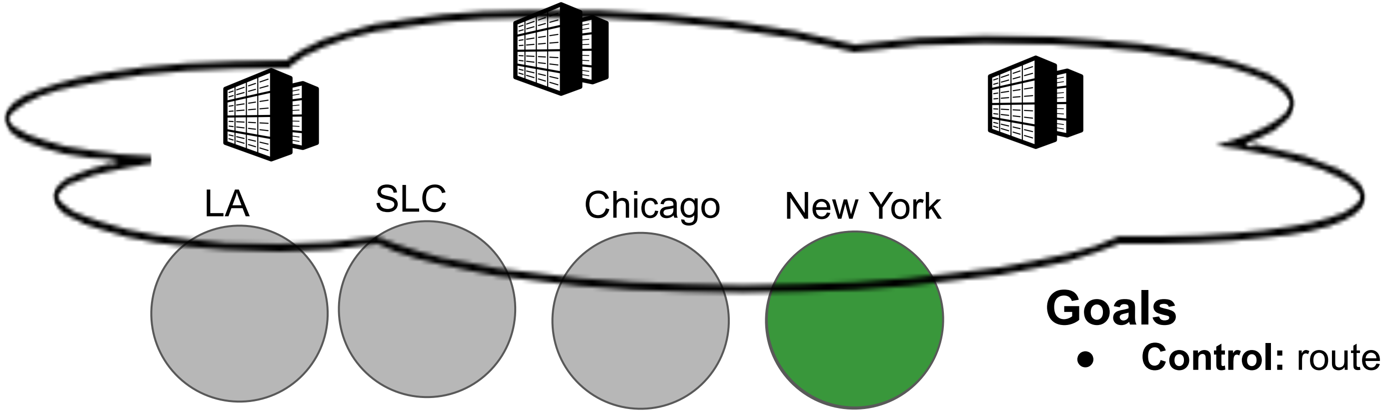
Goals: **Control** and **Availability**



Goals

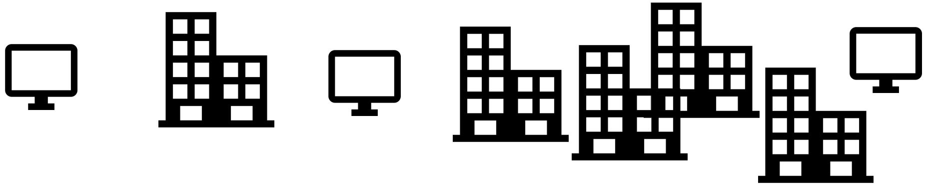
- **Control:** route client to best-performing site

Goals: **Control** and **Availability**

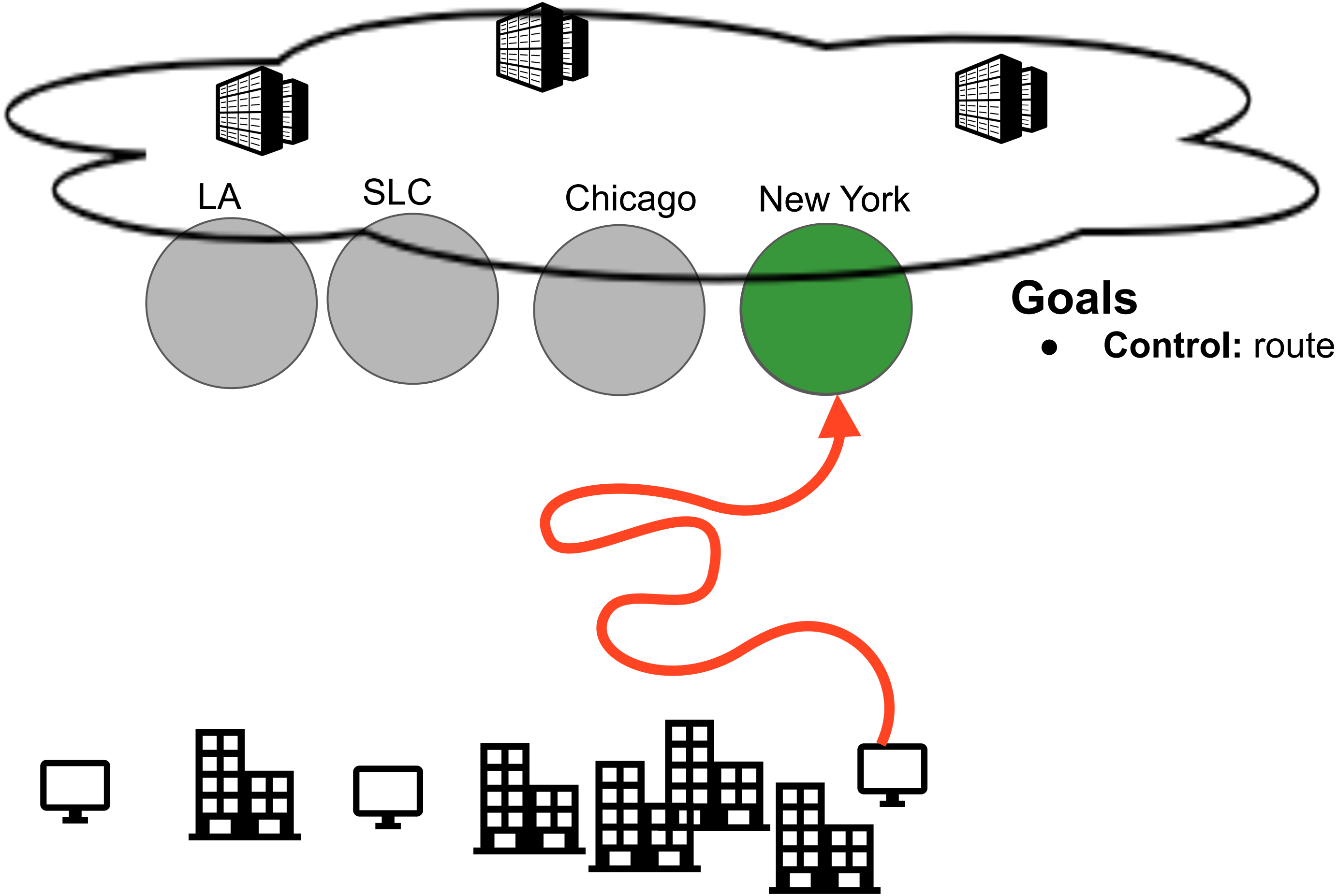


Goals

- **Control:** route client to best-performing site



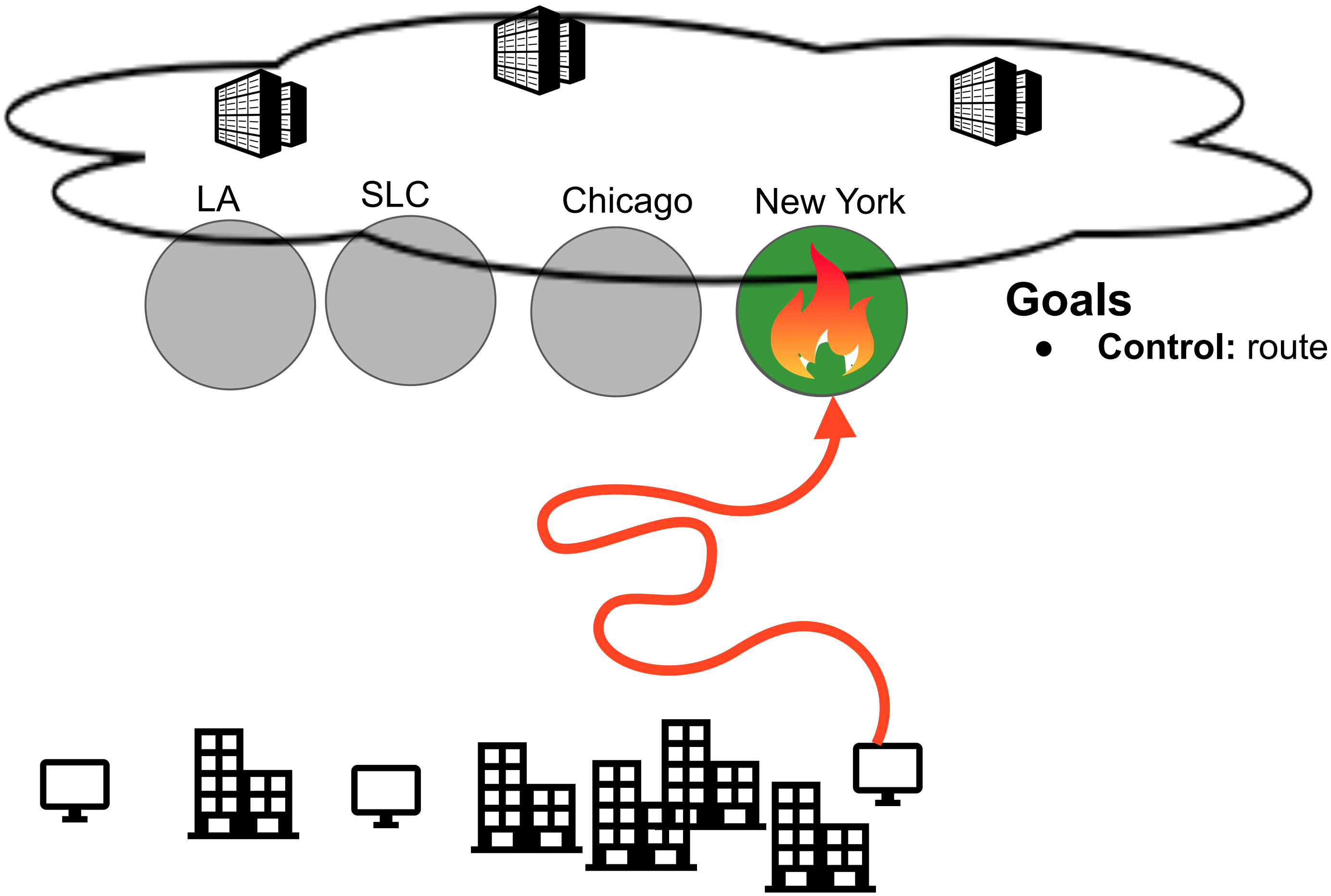
Goals: **Control** and **Availability**



Goals

- **Control:** route client to best-performing site

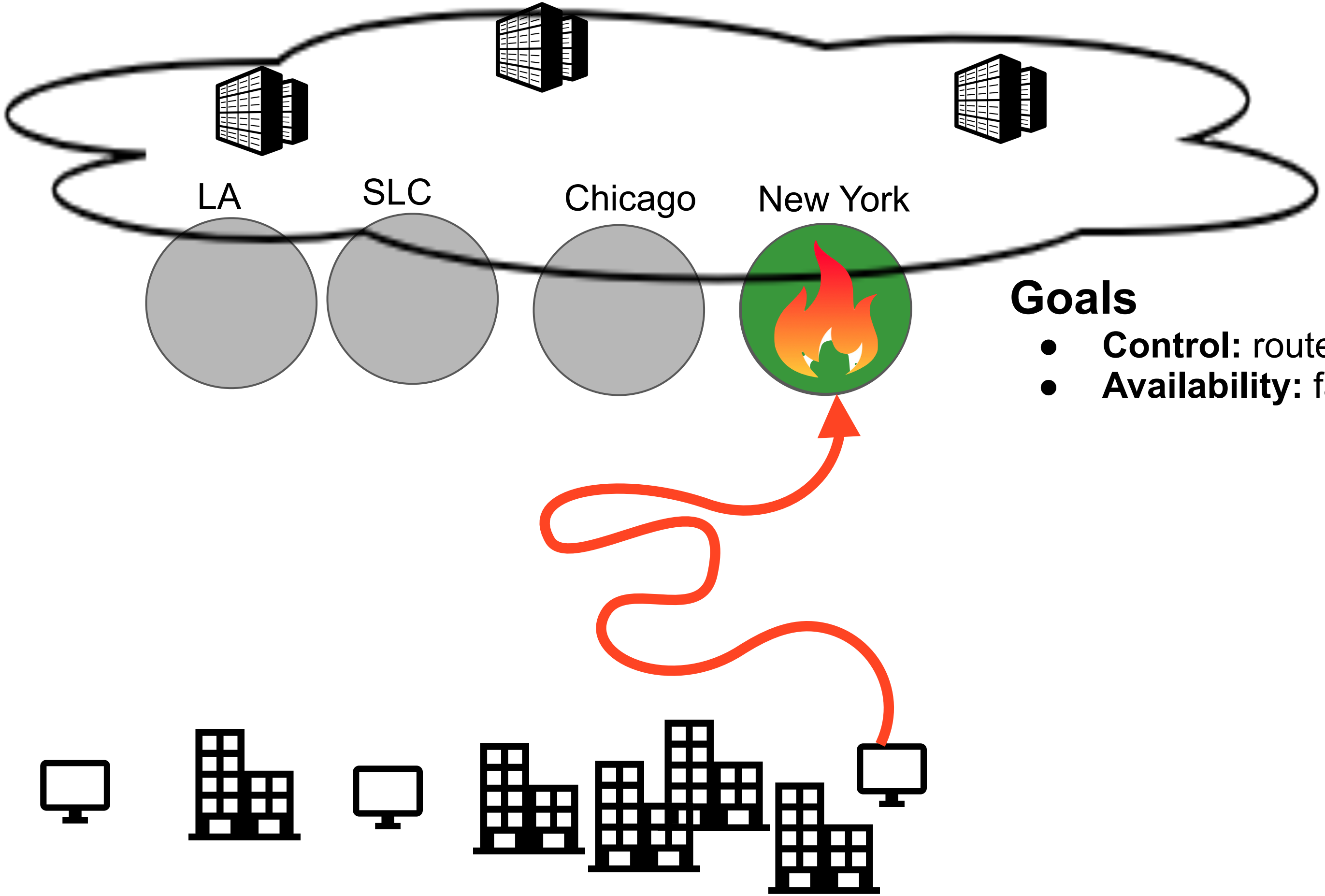
Goals: **Control** and **Availability**



Goals

- **Control:** route client to best-performing site

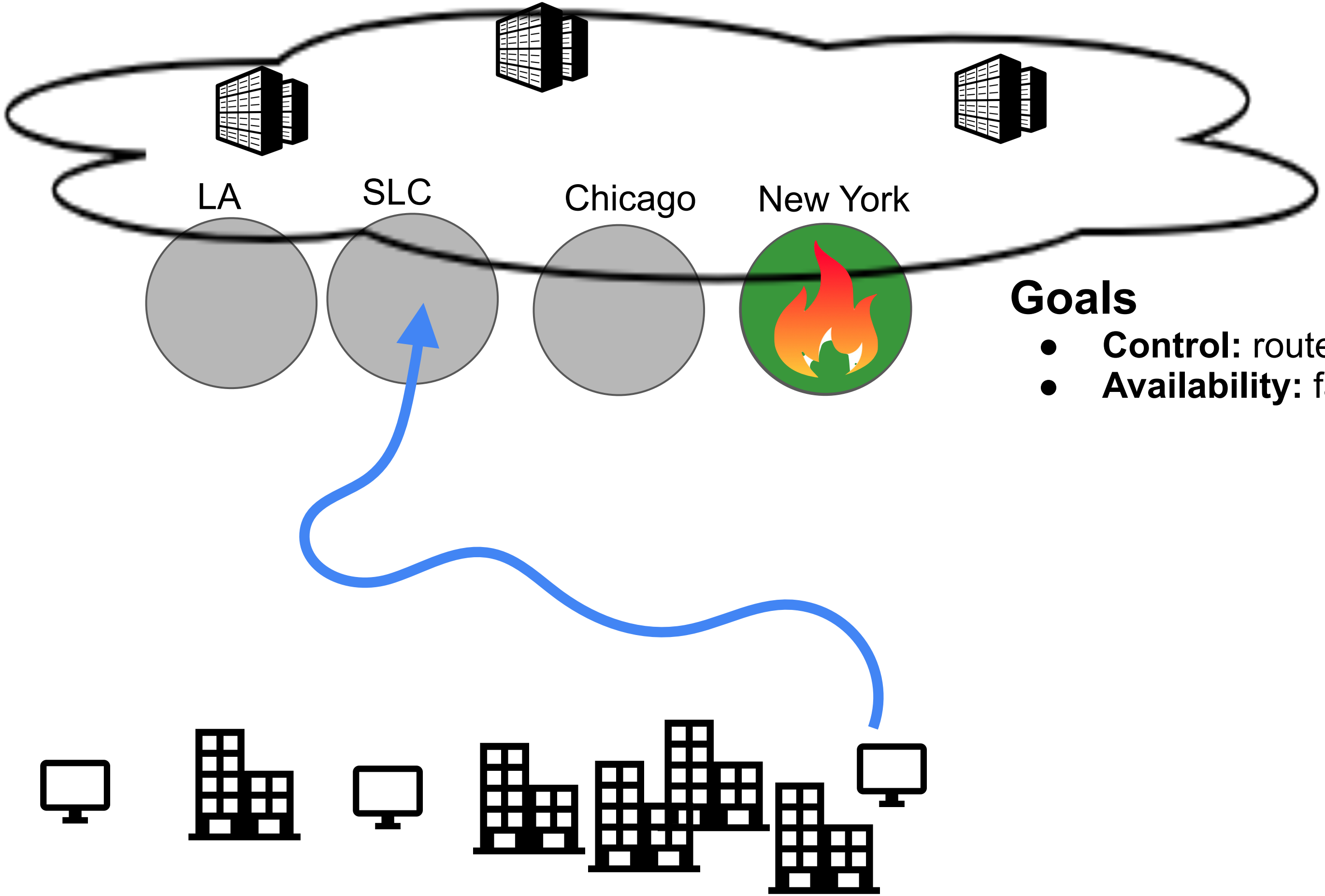
Goals: **Control** and **Availability**



Goals

- **Control:** route client to best-performing site
- **Availability:** fast failover to other sites

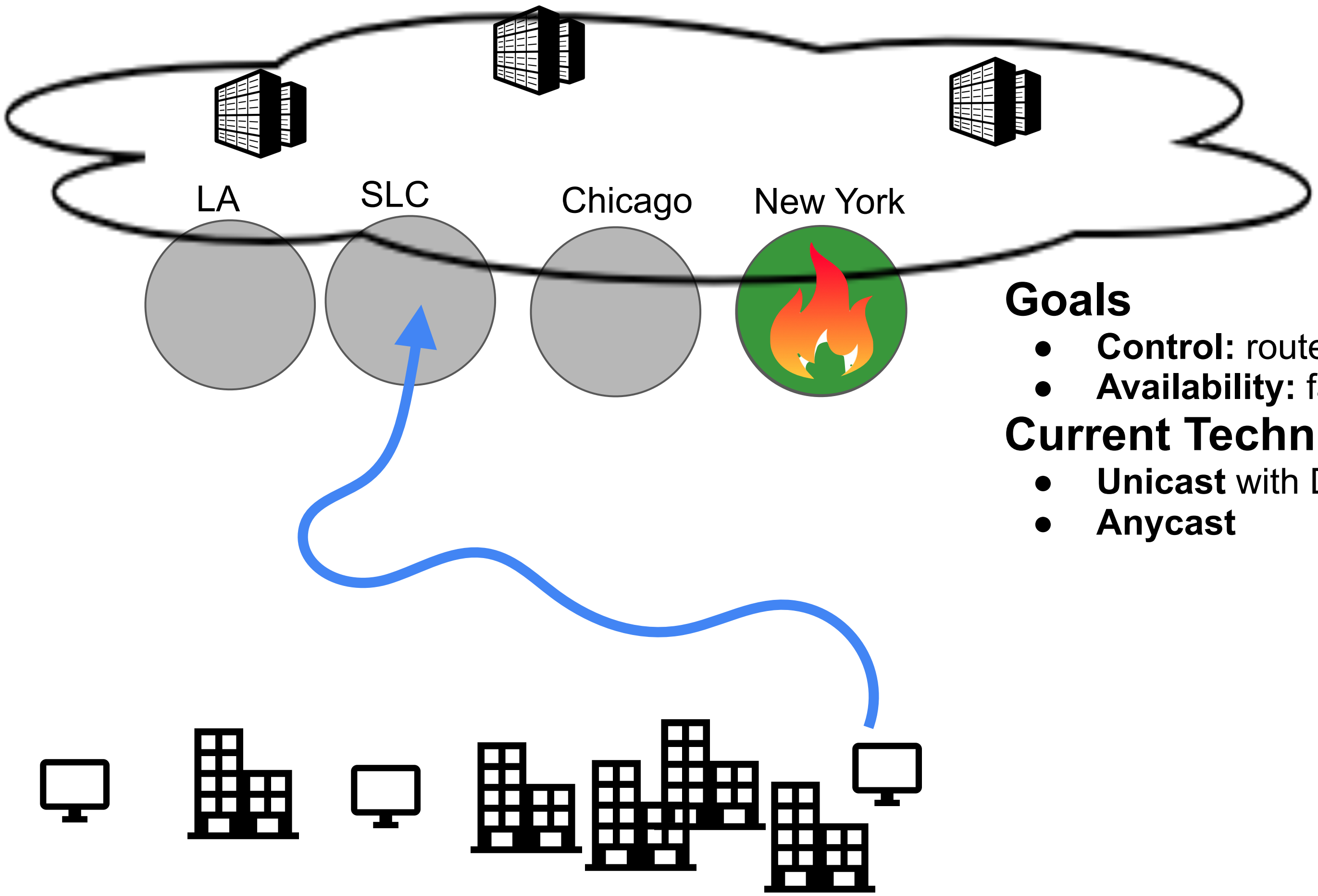
Goals: **Control** and **Availability**



Goals

- **Control:** route client to best-performing site
- **Availability:** fast failover to other sites

Goals: **Control** and **Availability**



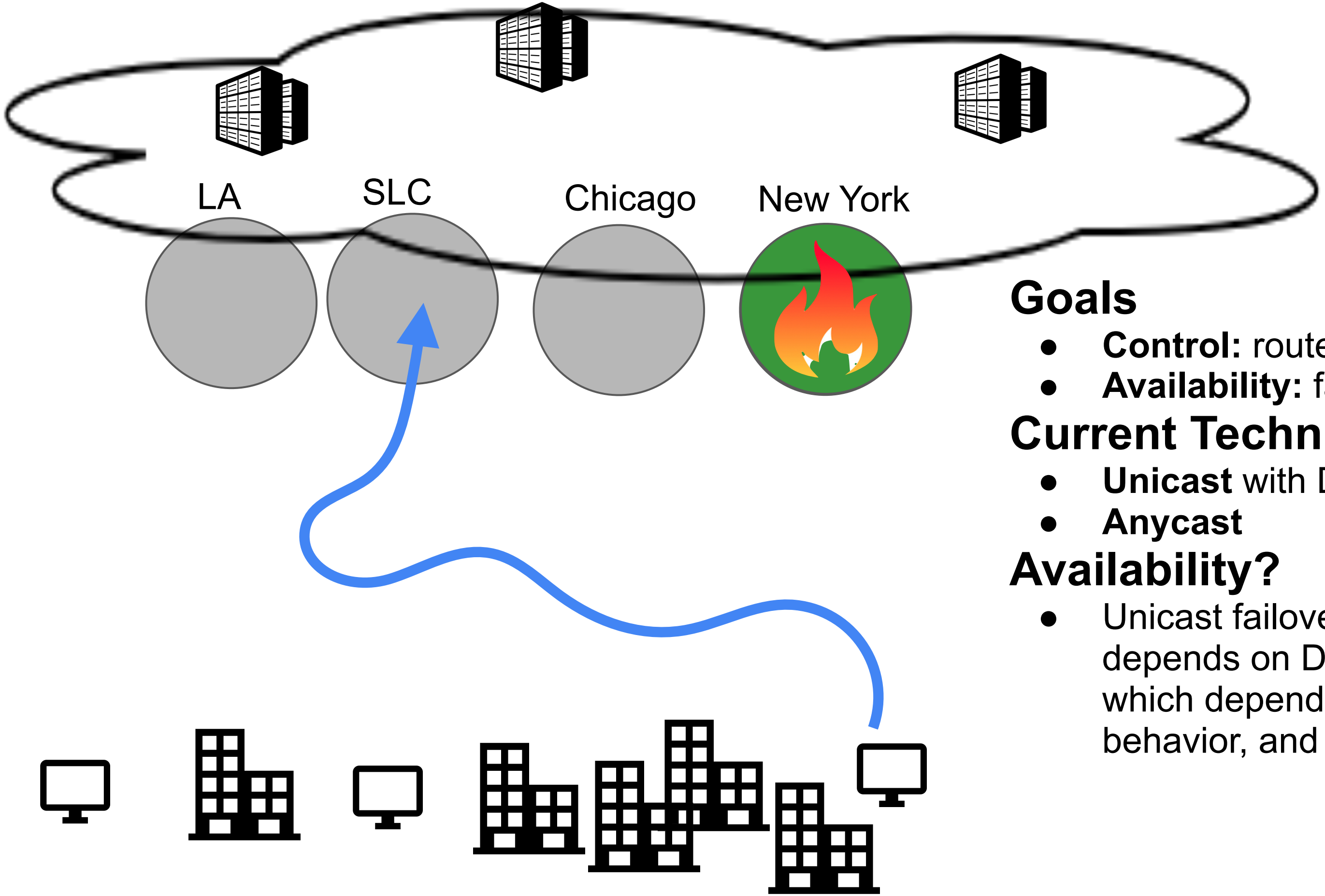
Goals

- **Control:** route client to best-performing site
- **Availability:** fast failover to other sites

Current Techniques

- **Unicast** with DNS redirection
- **Anycast**

Goals: **Control** and **Availability**



Goals

- **Control:** route client to best-performing site
- **Availability:** fast failover to other sites

Current Techniques

- **Unicast** with DNS redirection
- **Anycast**

Availability?

- Unicast failover (and hence availability) depends on DNS caching behavior, which depends on traffic patterns, OS behavior, and application behavior

Updates on two community resources — please use them!

PEERING BGP testbed

- Exchange BGP routes and traffic with thousands of ASes at locations around the world

Residential traffic traces

- Packet traces from ~1000 residences
    - Plan to scale to 8000 units, 24x7

Updates on two community resources — please use them!

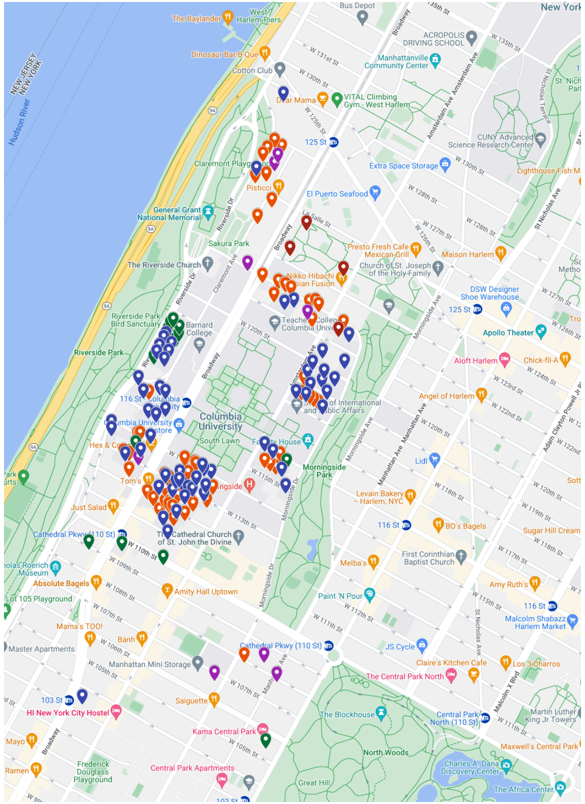
PEERING BGP testbed

- Exchange BGP routes and traffic with thousands of ASes at locations around the world

Residential traffic traces

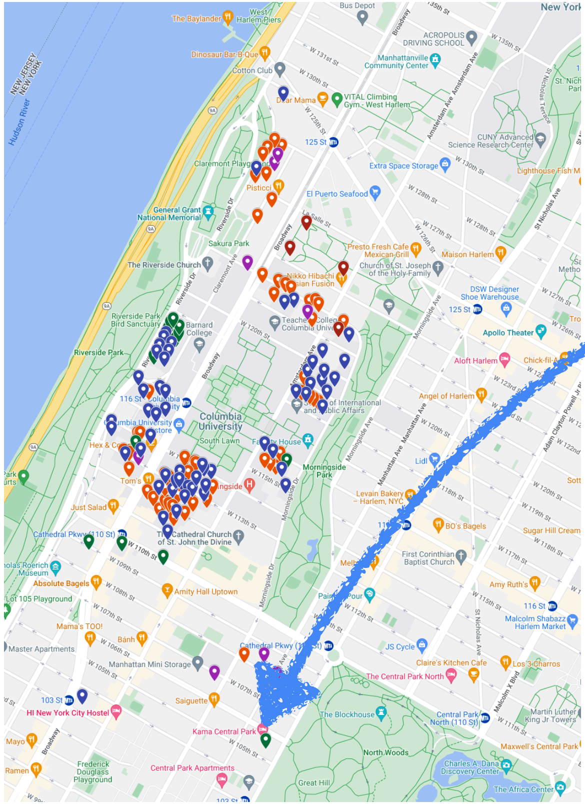
- Packet traces from ~1000 residences
    - Plan to scale to 8000 units, 24x7

Residential traces to understand real-world behavior



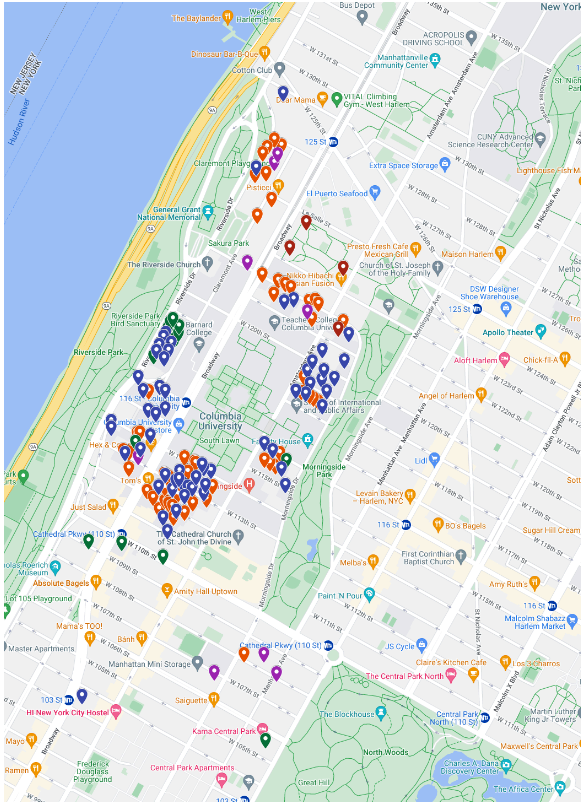
- Columbia operates a residential ISP
  - Largest landlord in NYC
  - 8000 faculty, postdocs, and grad students (and their families) in off-campus apartments (not undergrad dorms)
  - All on Columbia network

Residential traces to understand real-world behavior

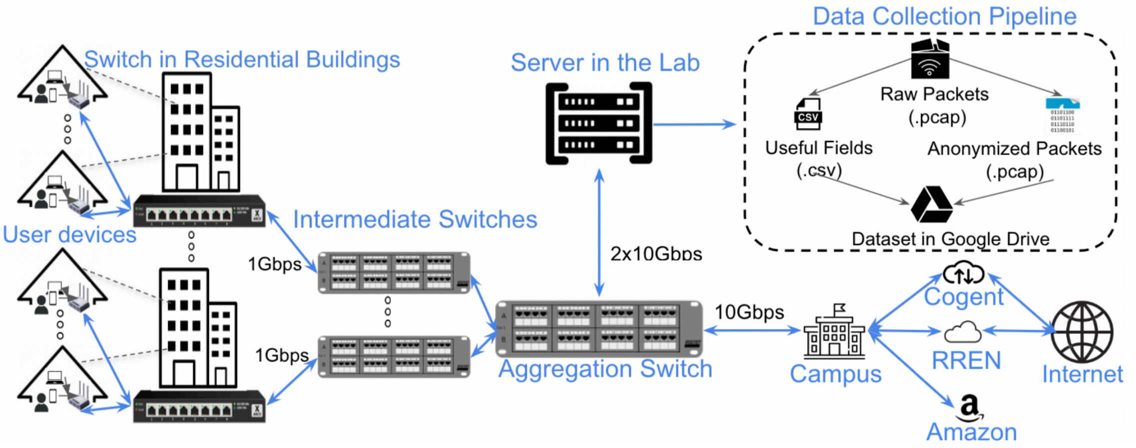


- Columbia operates a residential ISP
  - Largest landlord in NYC
  - 8000 faculty, postdocs, and grad students (and their families) in off-campus apartments (not undergrad dorms)
  - All on Columbia network…(except Ethan's building)

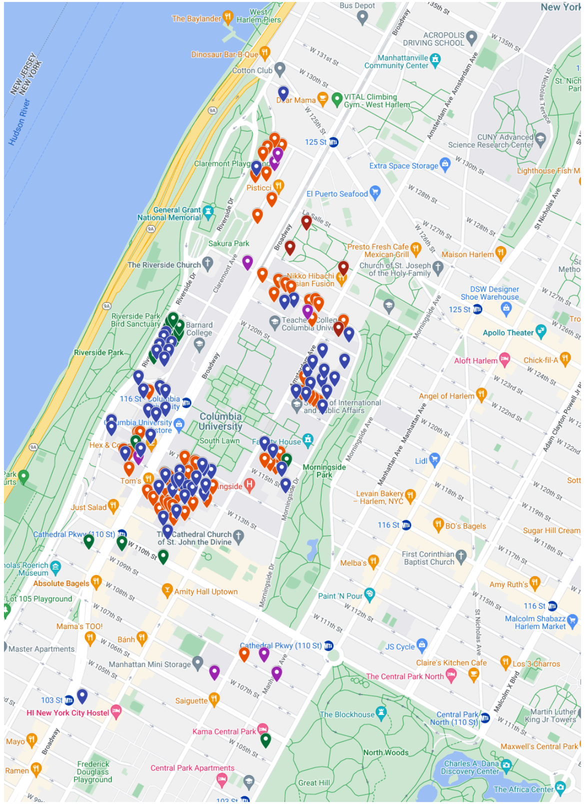
Residential traces to understand real-world behavior



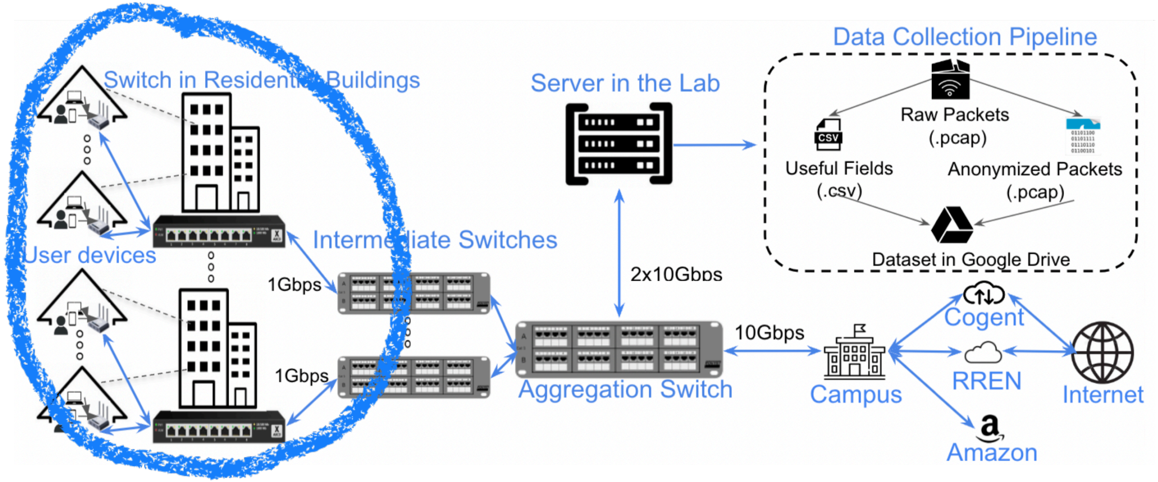
- Columbia operates a residential ISP
  - Largest landlord in NYC
  - 8000 faculty, postdocs, and grad students (and their families) in off-campus apartments (not undergrad dorms)
  - All on Columbia network…(except Ethan's building)



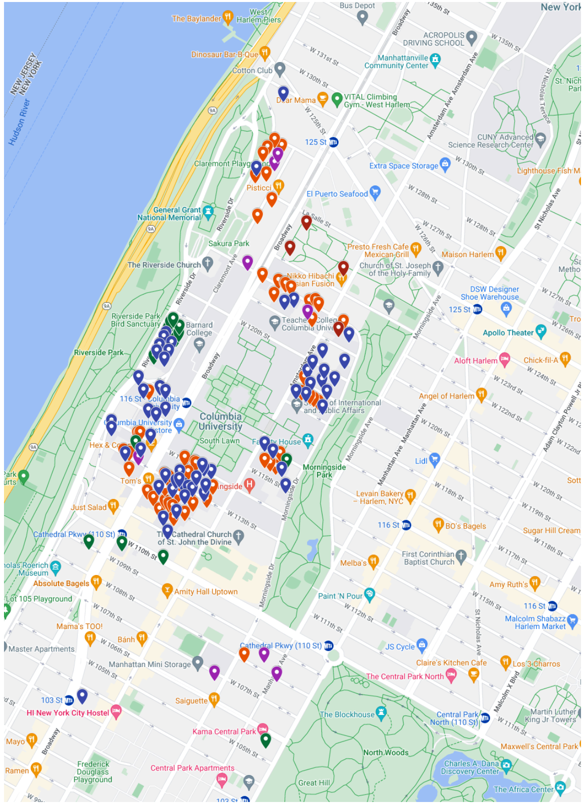
Residential traces to understand real-world behavior



- Columbia operates a residential ISP
  - Largest landlord in NYC
  - 8000 faculty, postdocs, and grad students (and their families) in off-campus apartments (not undergrad dorms)
  - All on Columbia network…(except Ethan's building)

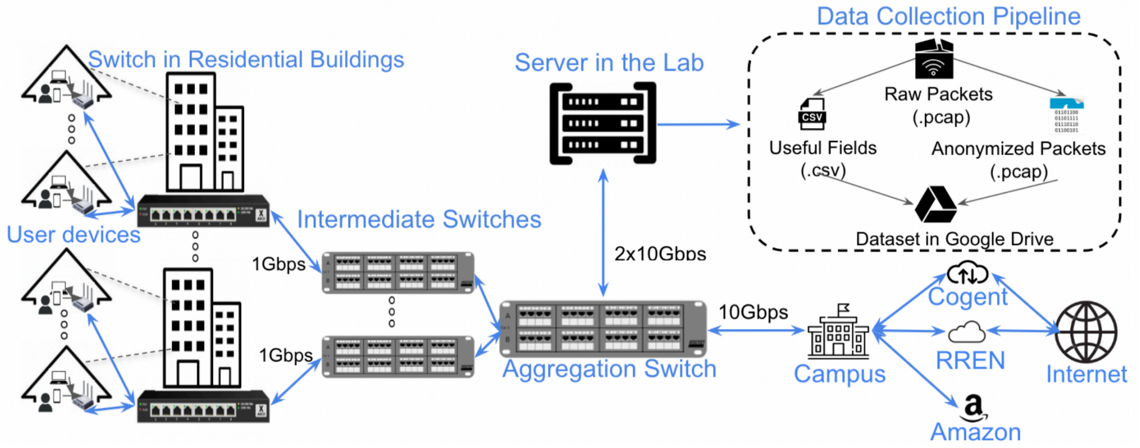


Residential traces to understand real-world behavior

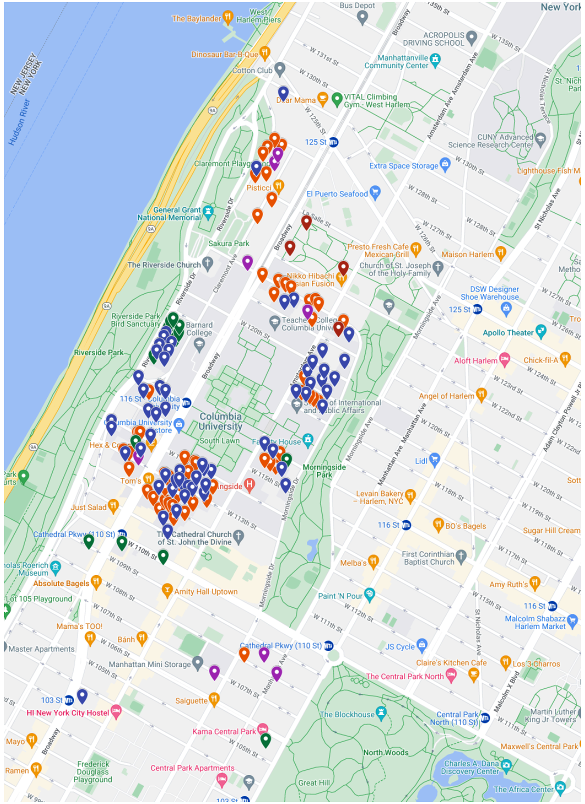


- Columbia operates a residential ISP
  - Largest landlord in NYC
  - 8000 faculty, postdocs, and grad students (and their families) in off-campus apartments (not undergrad dorms)
  - All on Columbia network…(except Ethan's building)

- Traffic mirrored to anonymization and collection pipeline
- Provides view of Internet activity typically invisible to academia

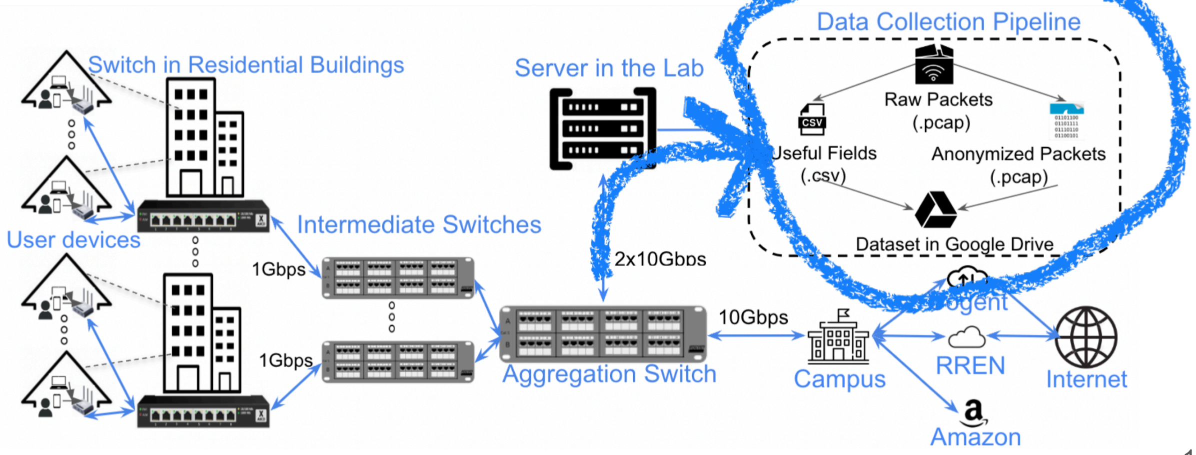


Residential traces to understand real-world behavior



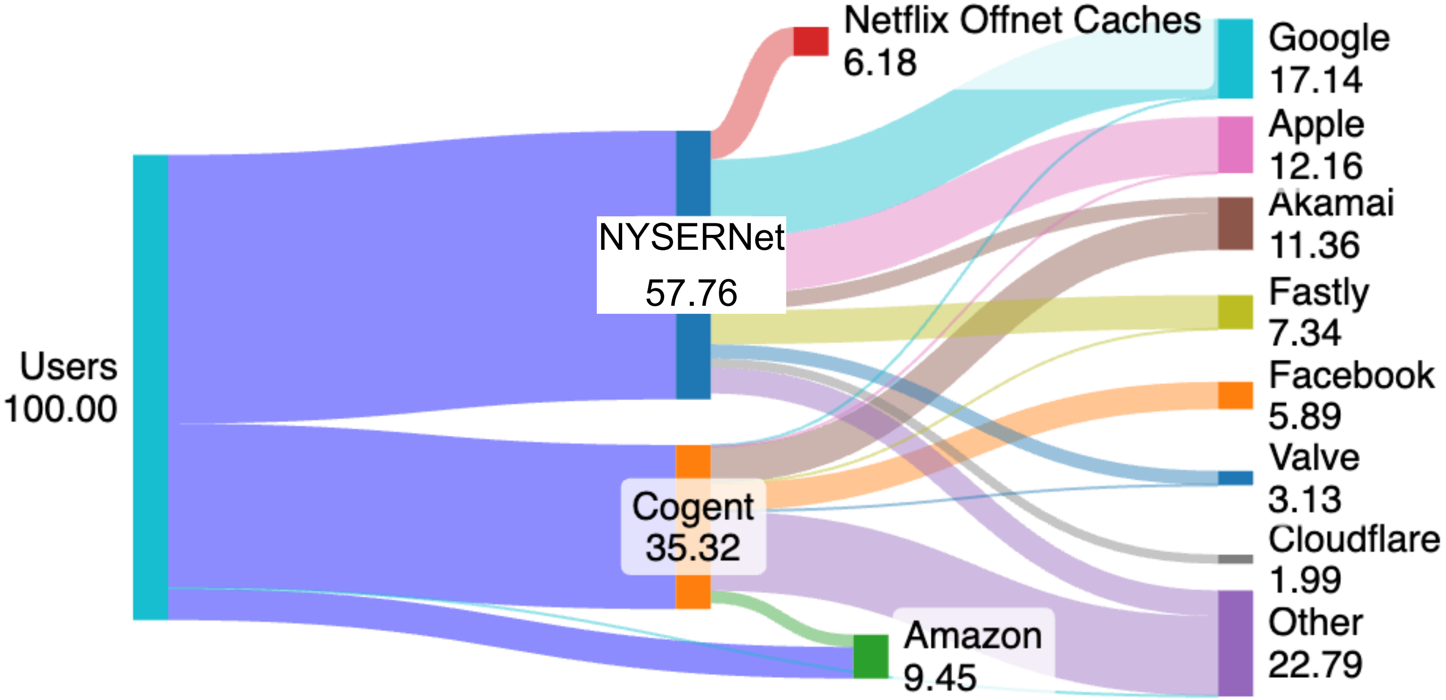
- Columbia operates a residential ISP
  - Largest landlord in NYC
  - 8000 faculty, postdocs, and grad students (and their families) in off-campus apartments (not undergrad dorms)
  - All on Columbia network…(except Ethan’s building)

- Traffic mirrored to anonymization and collection pipeline
- Provides view of Internet activity typically invisible to academia

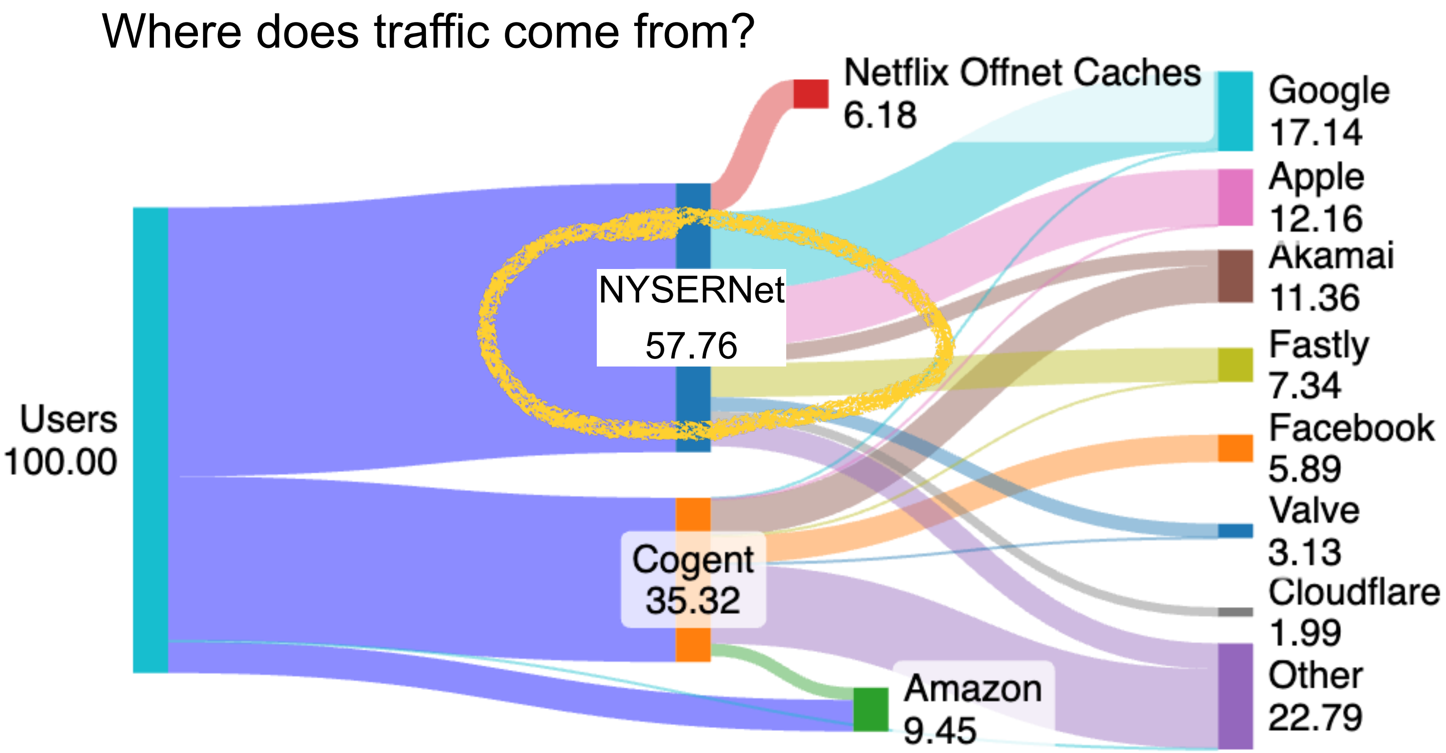


Residential traces to understand real-world behavior

Where does traffic come from?

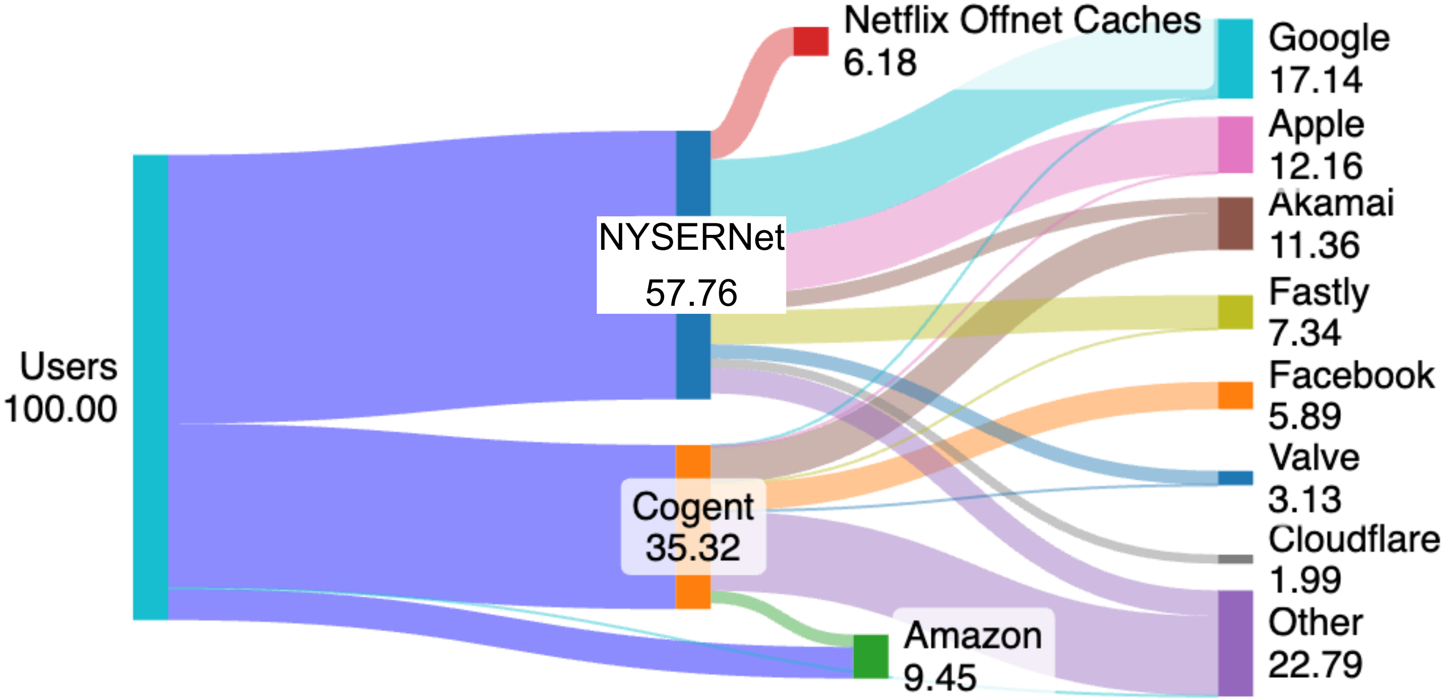


Residential traces to understand real-world behavior

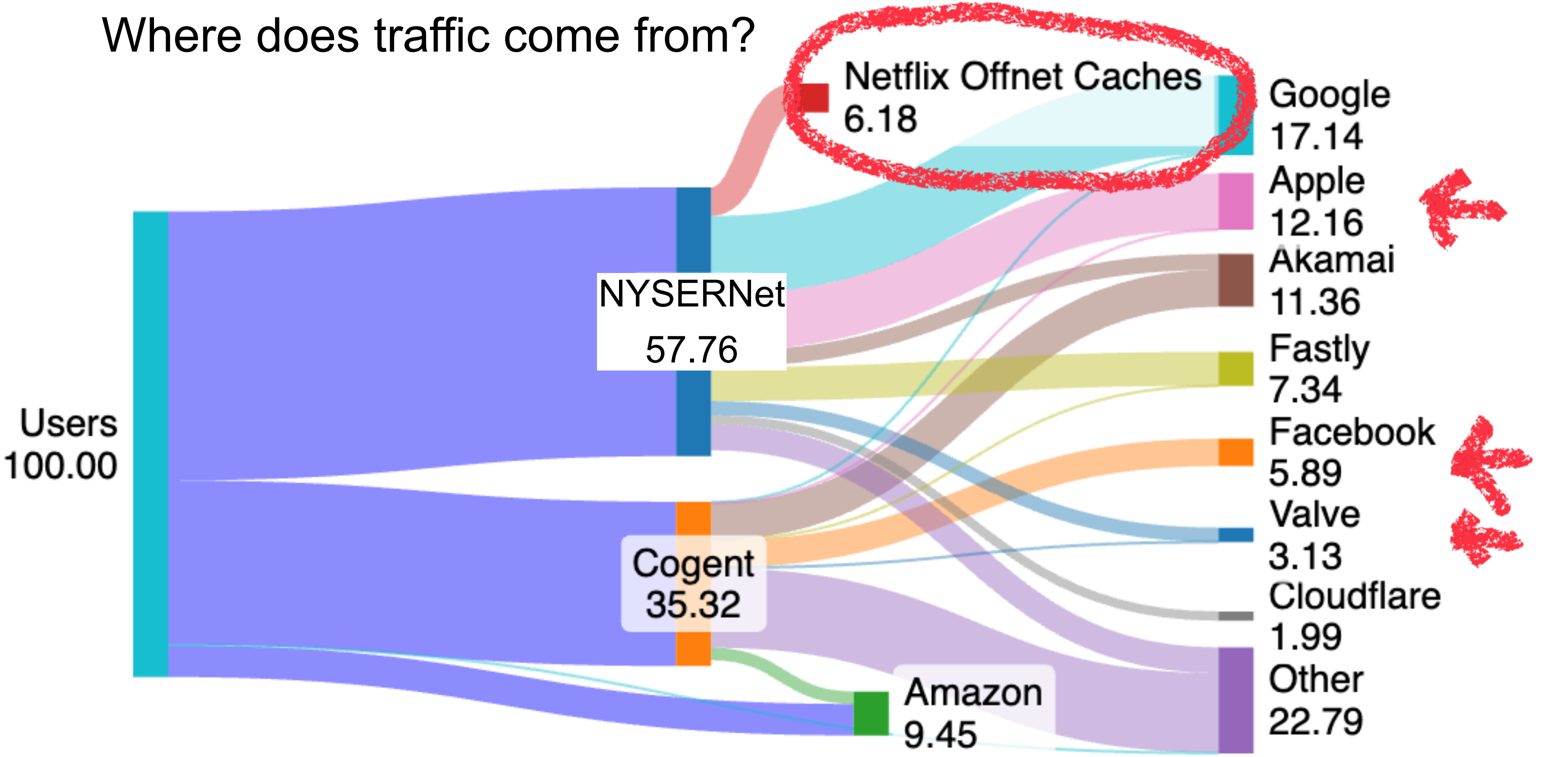


Residential traces to understand real-world behavior

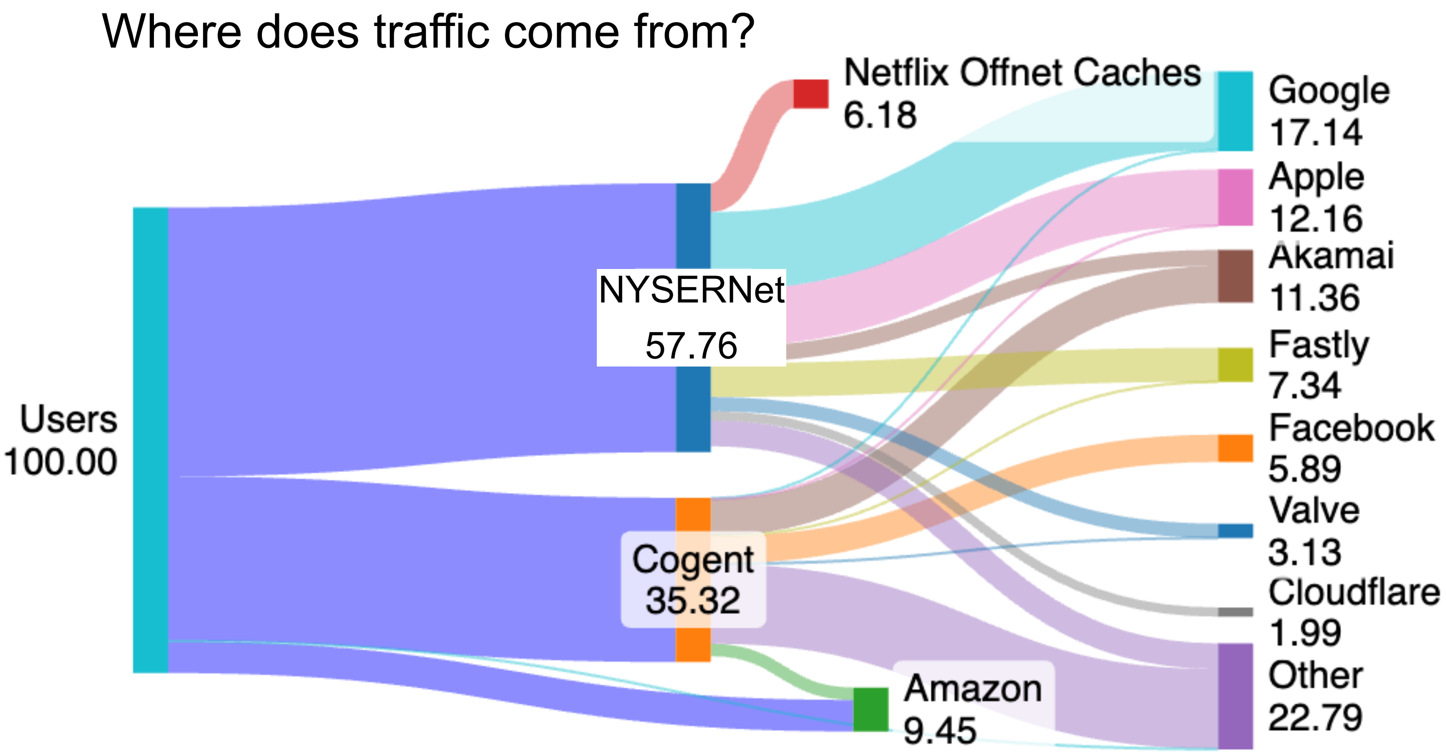
Where does traffic come from?



Residential traces to understand real-world behavior

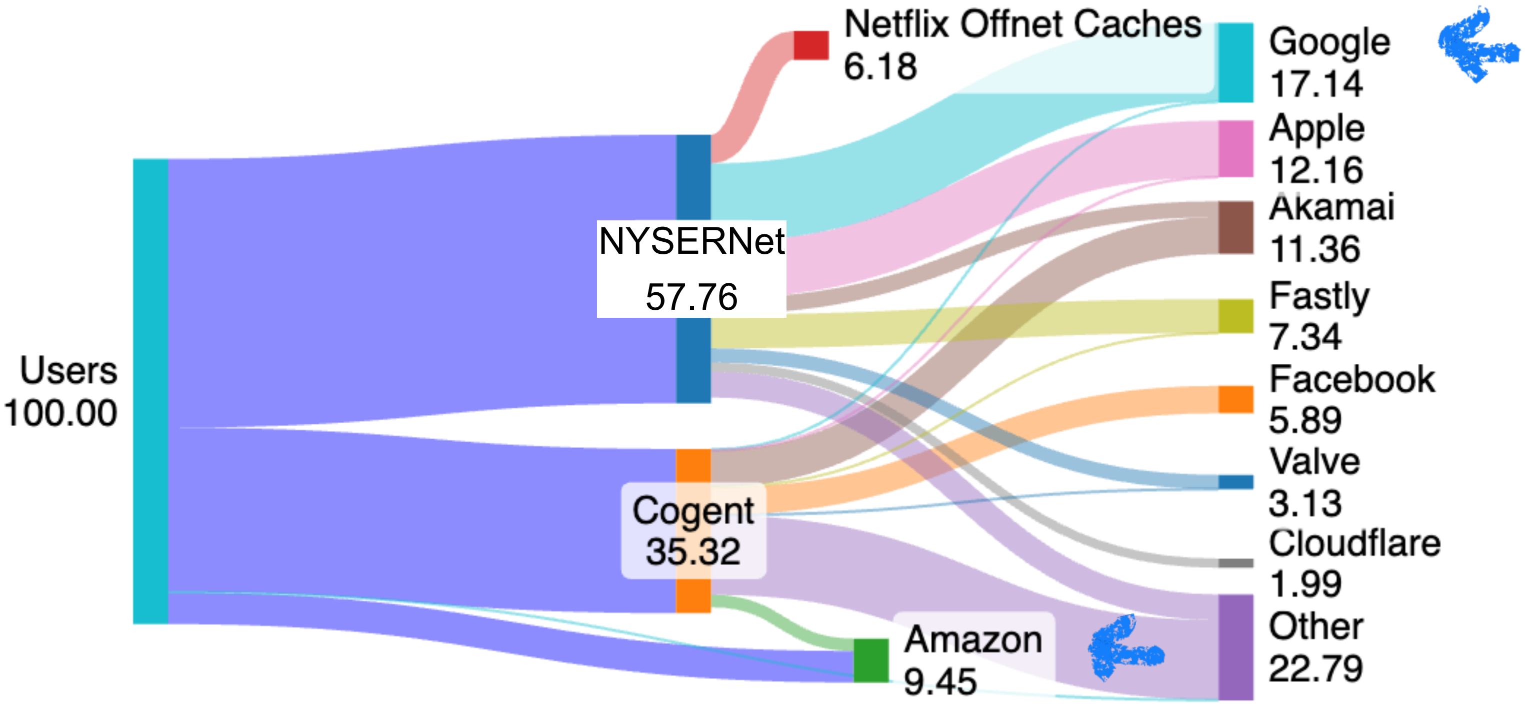


Residential traces to understand real-world behavior

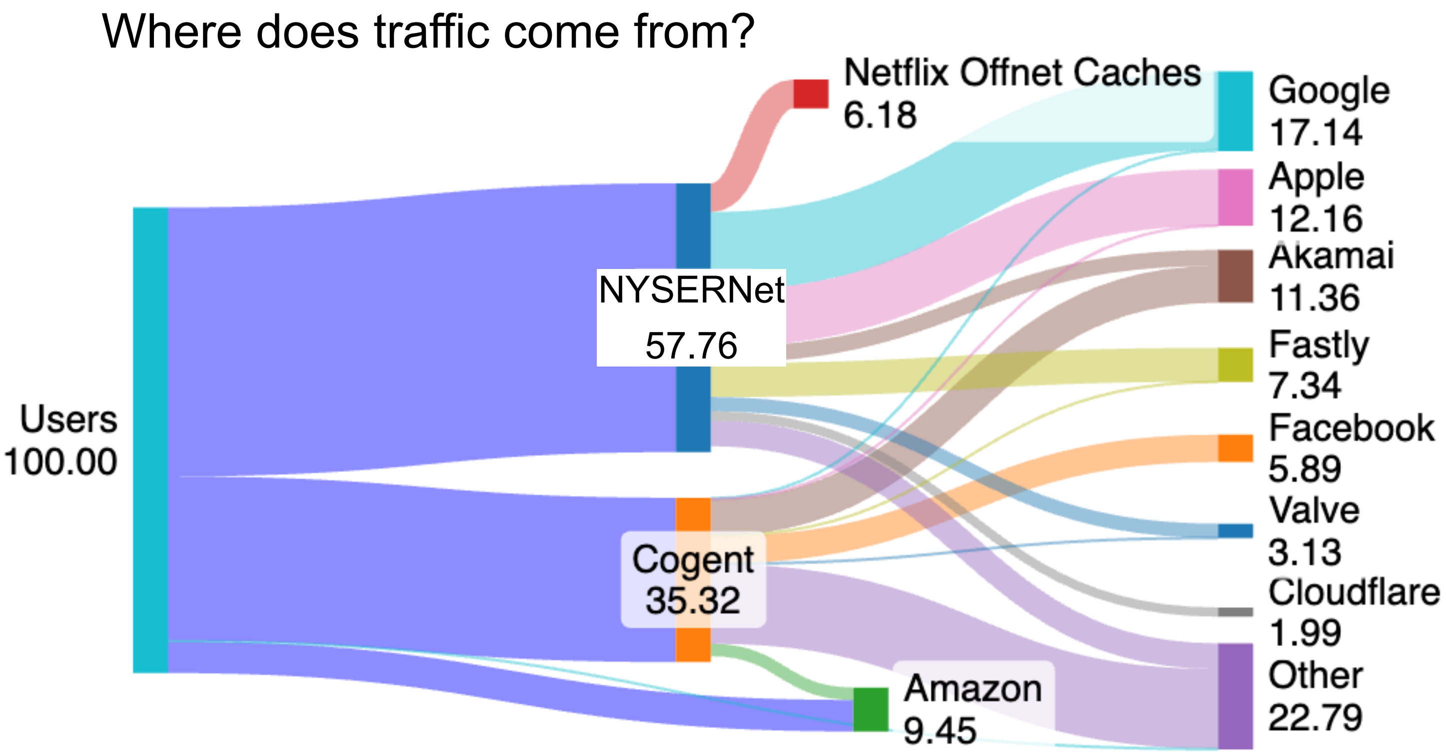


Residential traces to understand real-world behavior

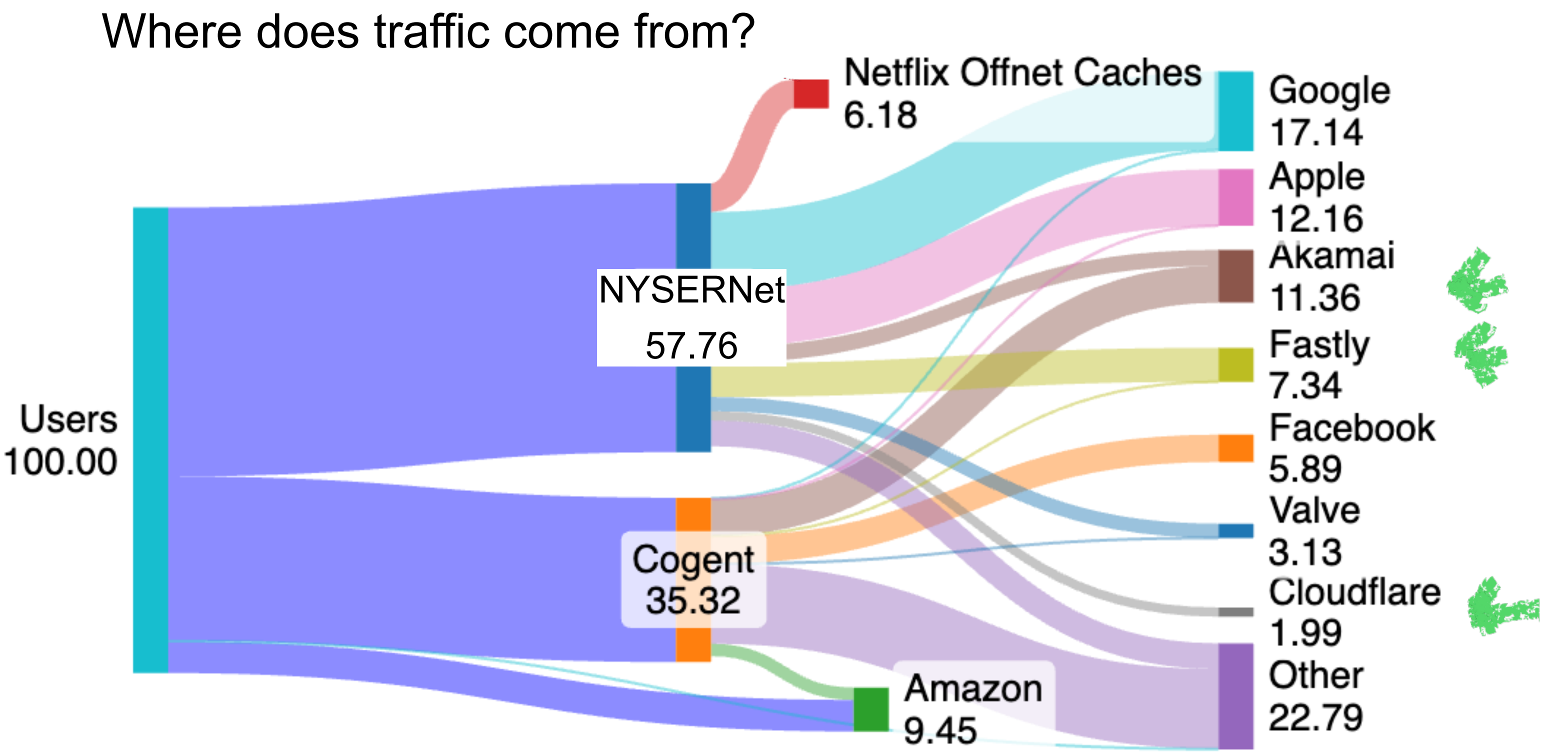
Where does traffic come from?



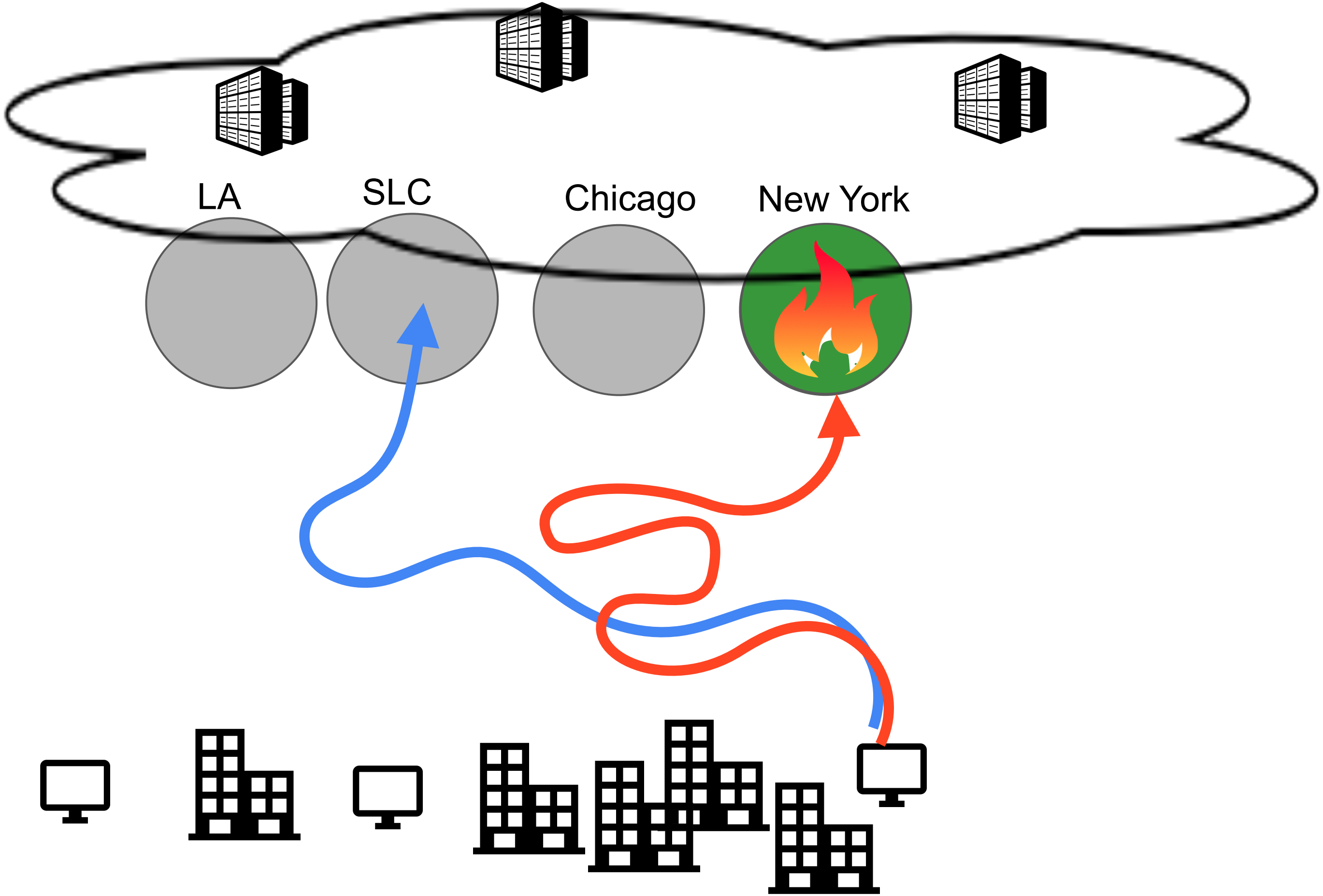
Residential traces to understand real-world behavior



Residential traces to understand real-world behavior

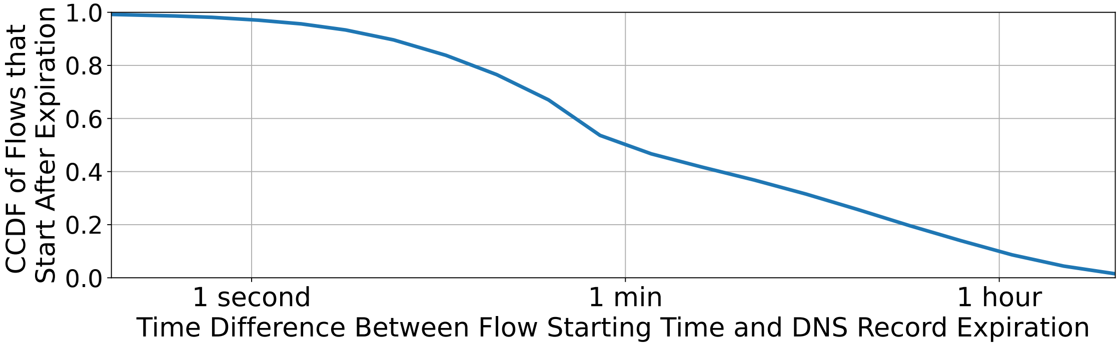


How quickly can DNS fail clients over to a new site?



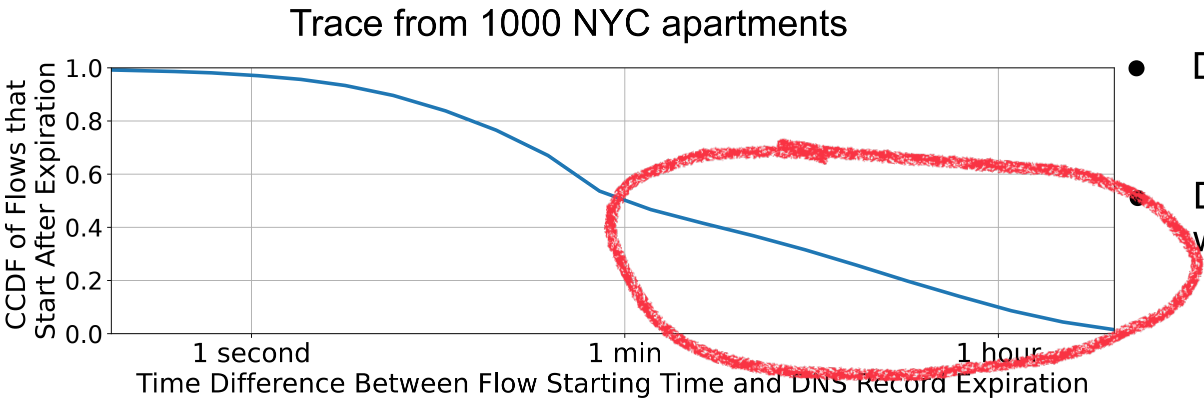
Unicast lacks **availability** in site failure scenarios

Trace from 1000 NYC apartments



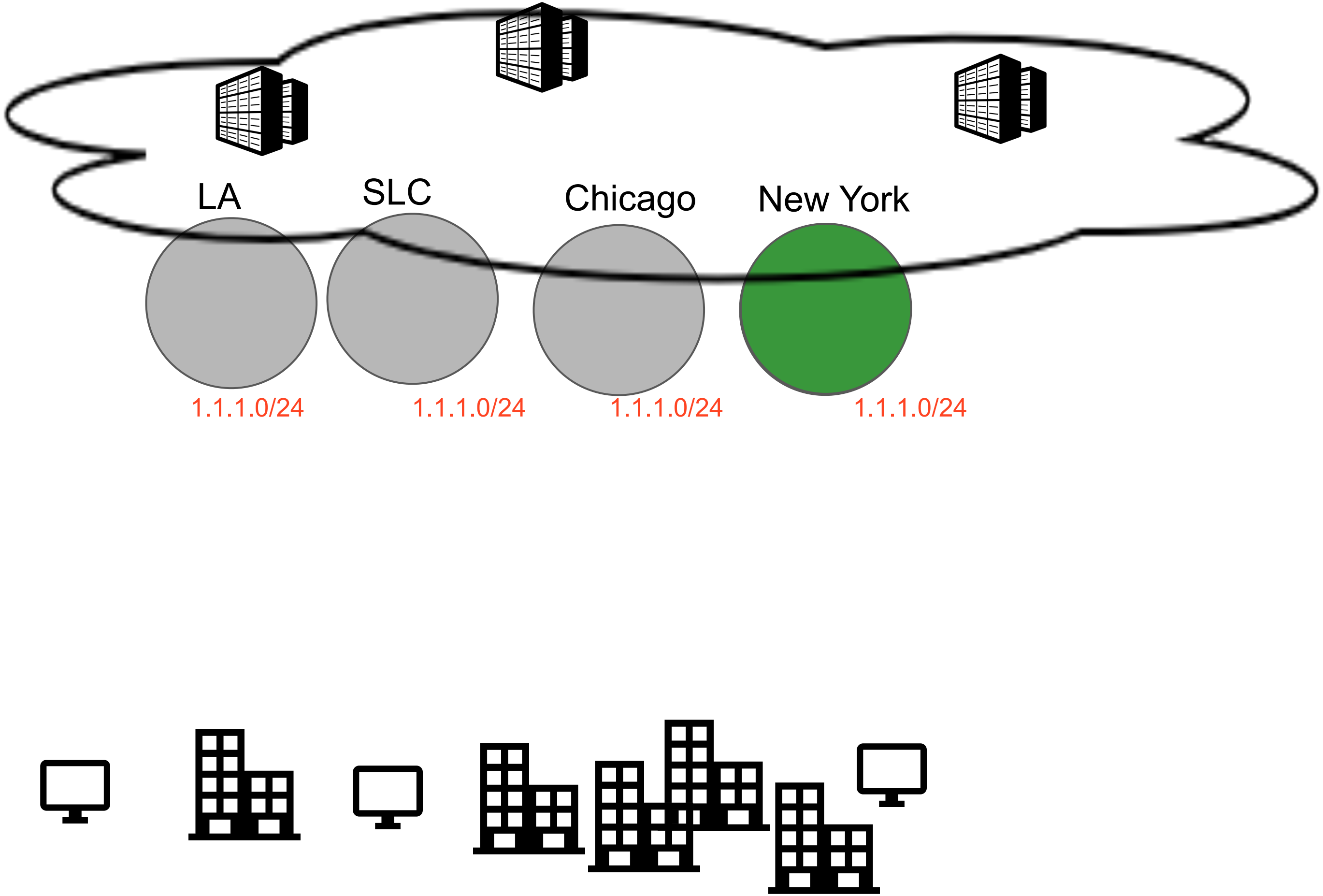
- DNS **controls** client-to-site mapping
- DNS update is slow due to caching, which limits **availability.**
  - Lower DNS TTL increases application latency.
  - TTL is often **violated.**
    13% of flows start after TTL expired
    Of those, 50% start > 1 min. later

Unicast lacks **availability** in site failure scenarios

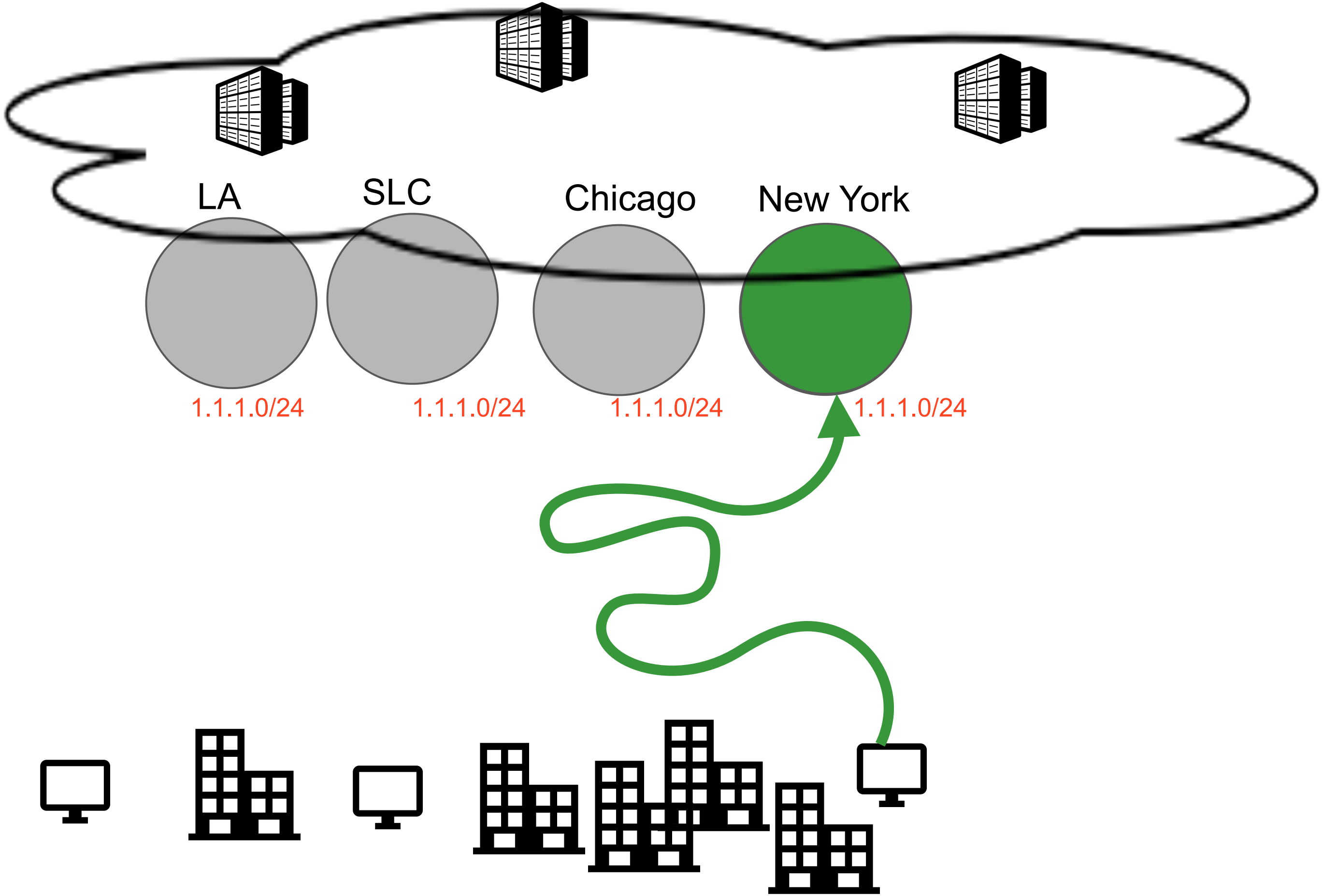


- DNS **controls** client-to-site mapping
- DNS update is slow due to caching, which limits **availability.**
  - Lower DNS TTL increases application latency.
  - TTL is often **violated.**
    13% of flows start after TTL expired
    Of those, 50% start > 1 min. later

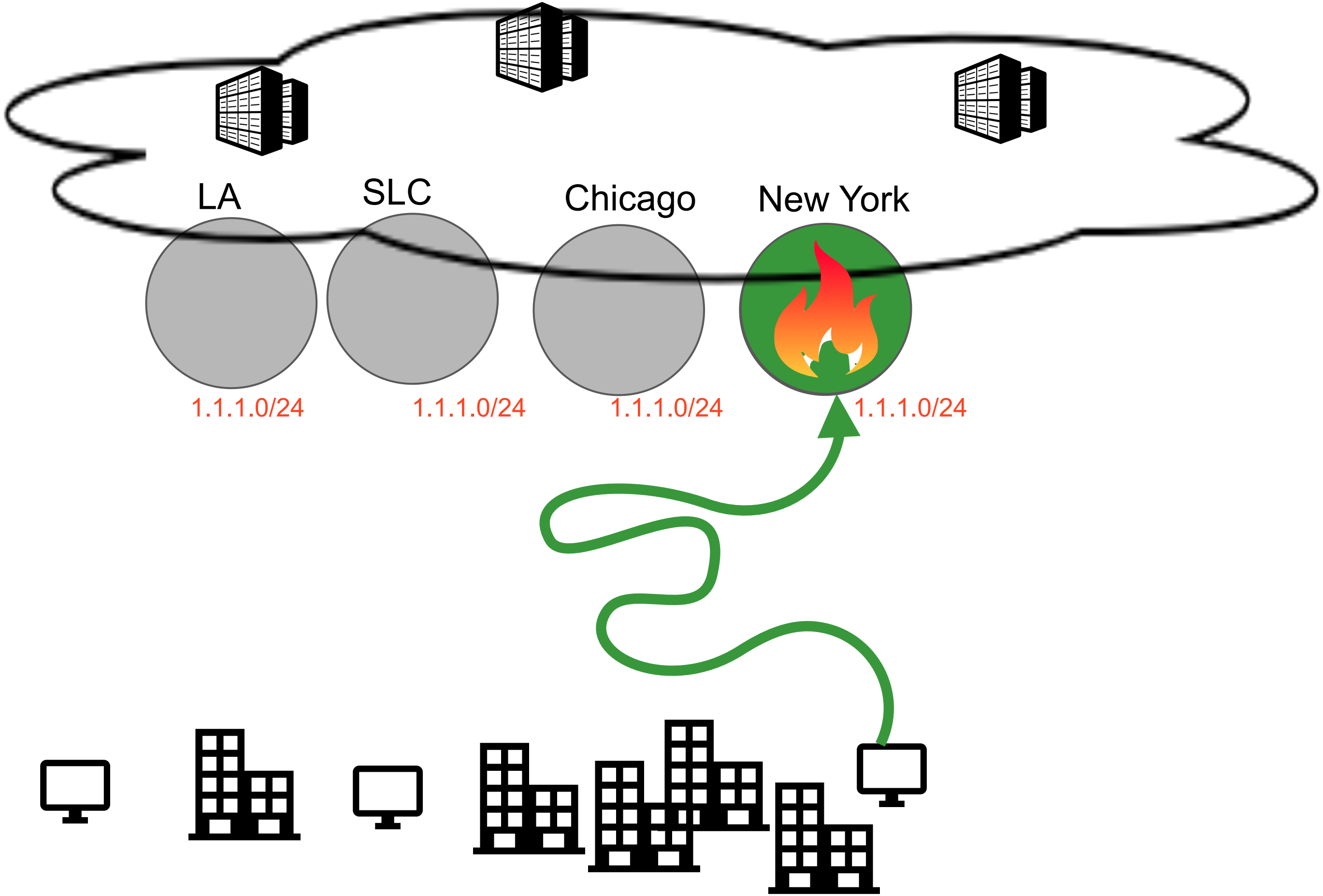
Anycast supports **fast failover** for **high availability**



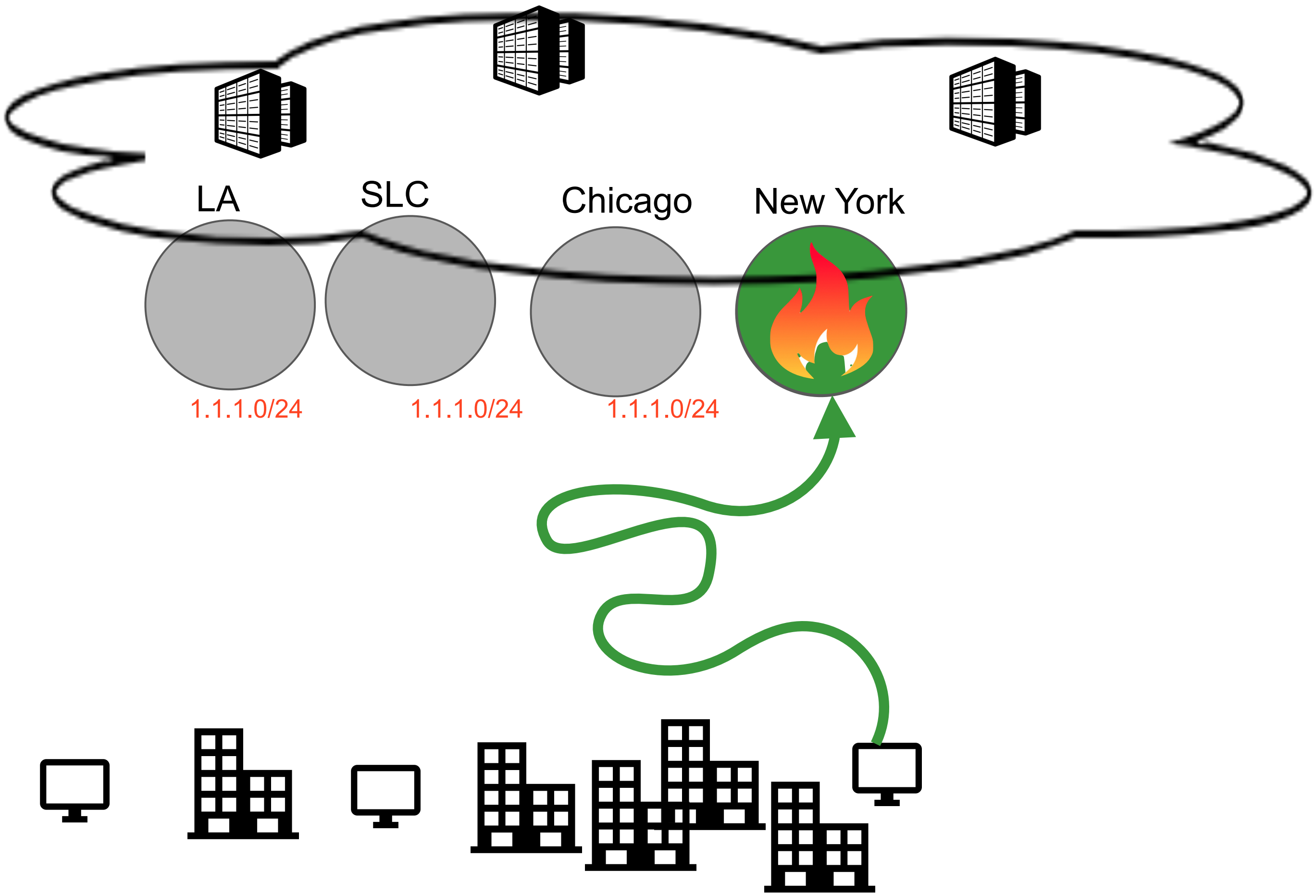
Anycast supports **fast failover** for **high availability**



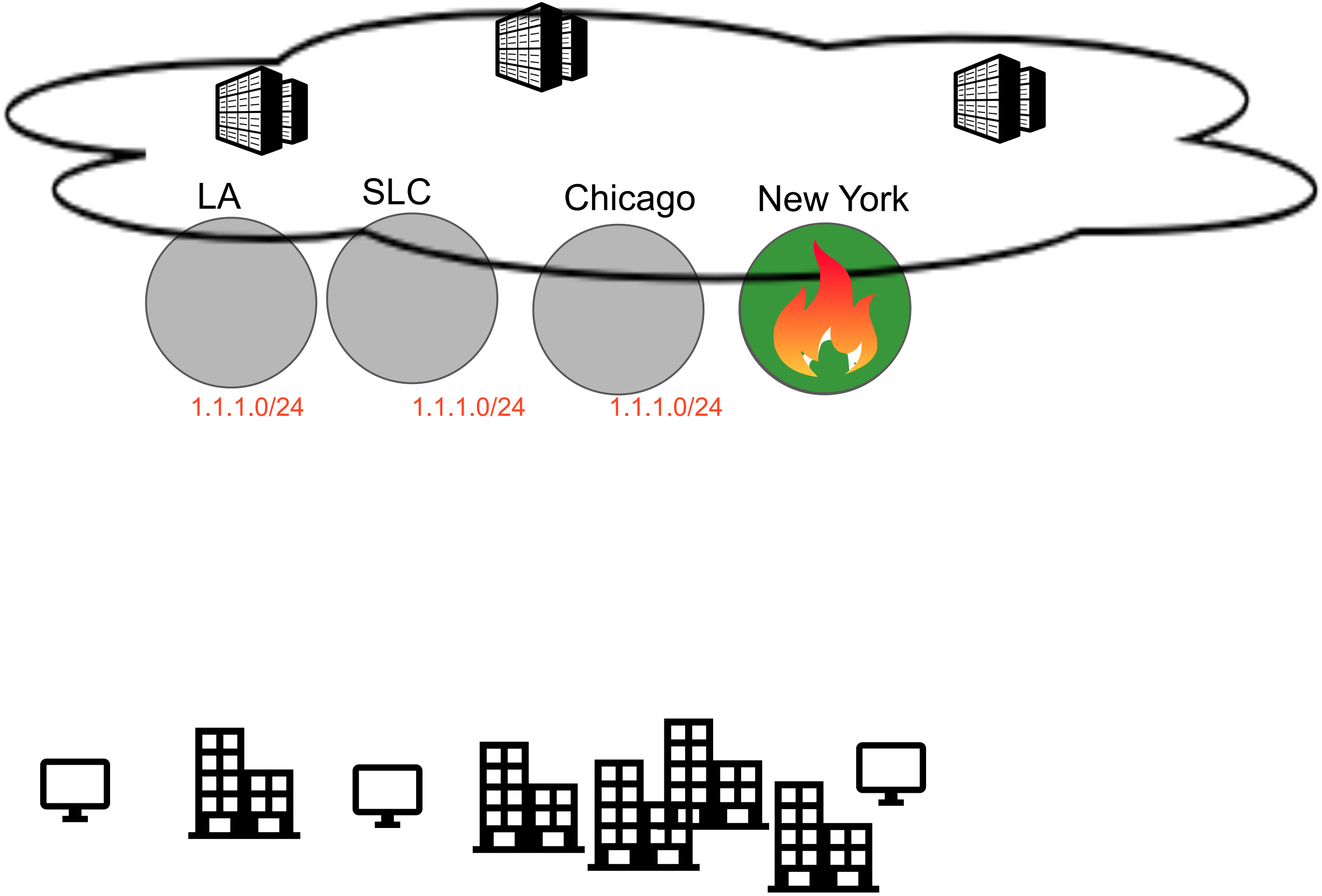
Anycast supports **fast failover** for **high availability**



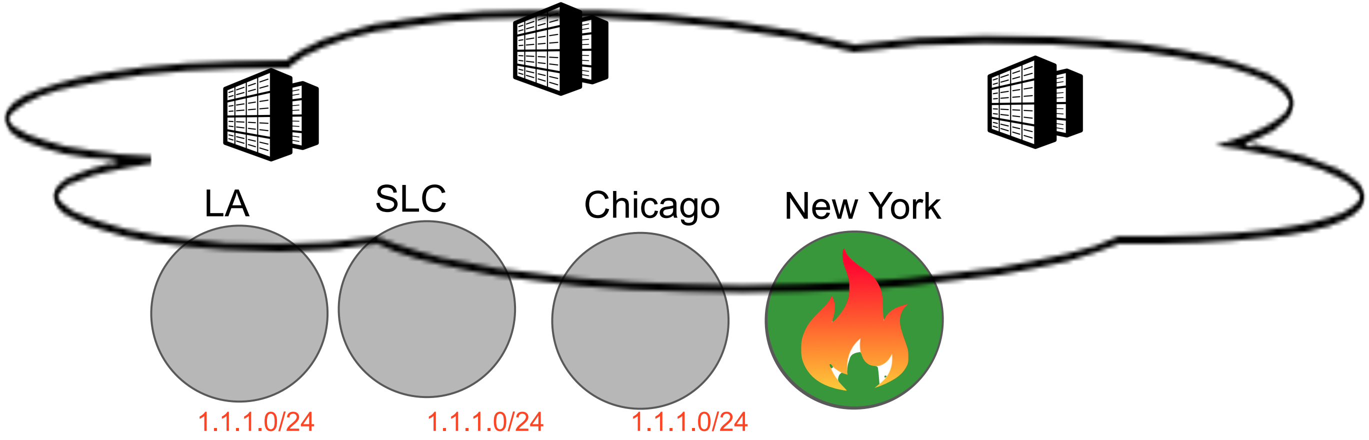
Anycast supports **fast failover** for **high availability**



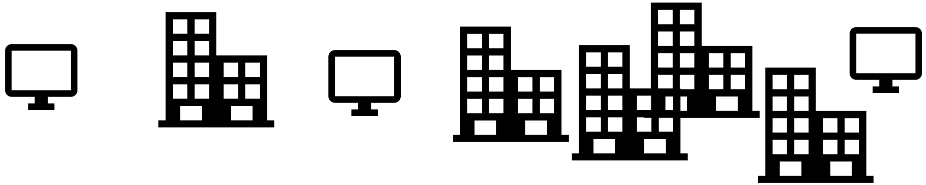
Anycast supports **fast failover** for **high availability**



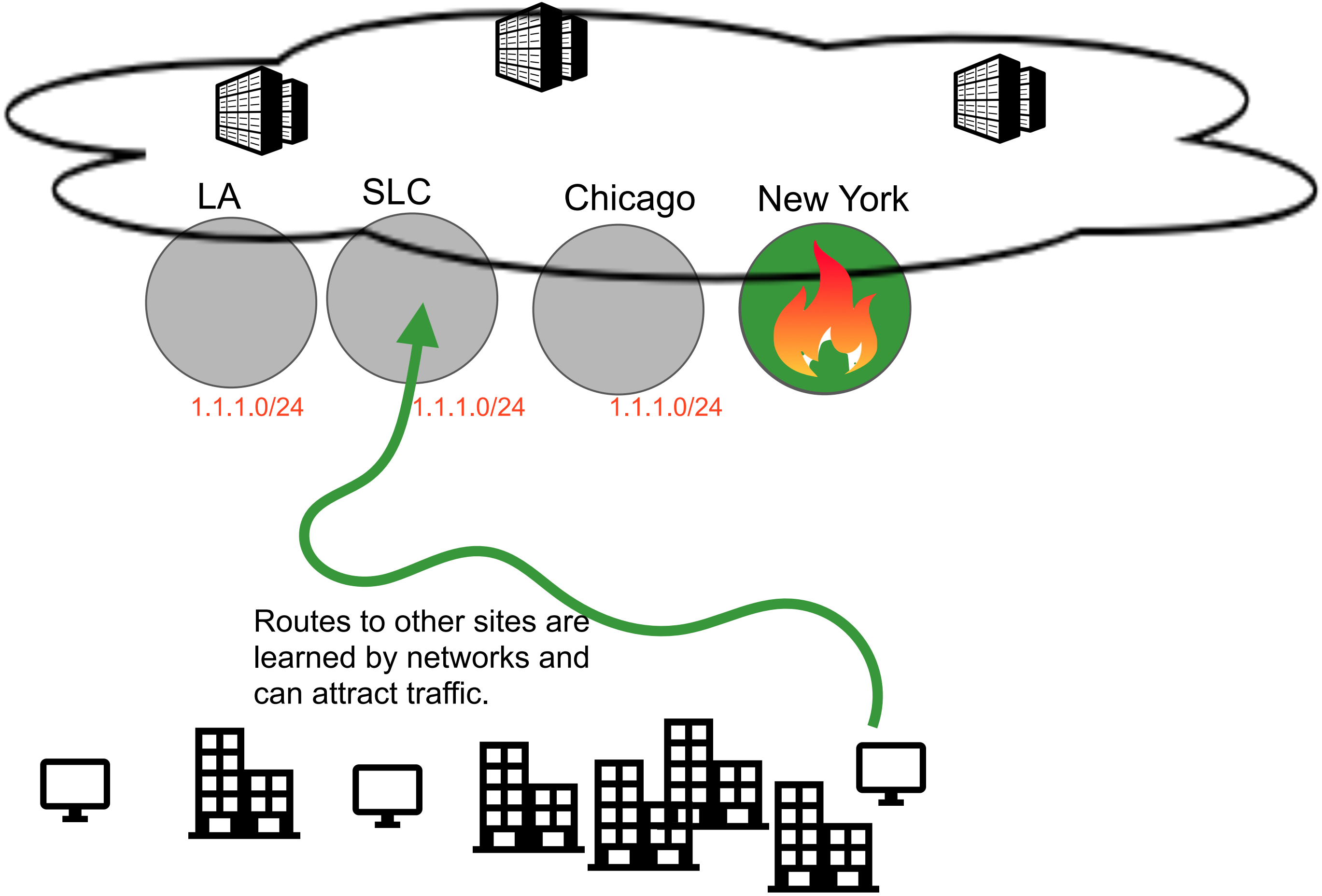
Anycast supports **fast failover** for **high availability**



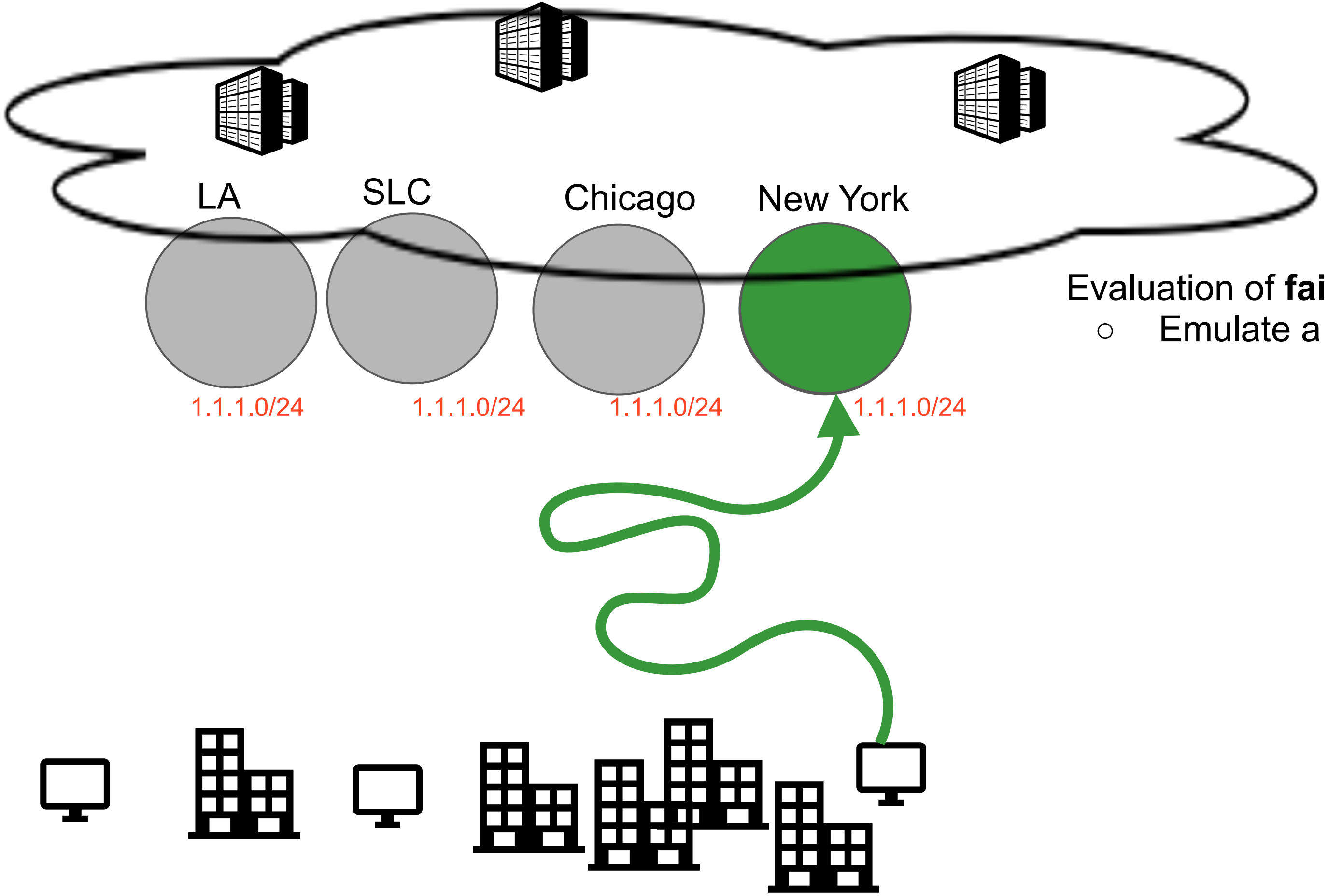
Routes to other sites are
learned by networks and
can attract traffic.



Anycast supports **fast failover** for **high availability**



Measuring anycast failover



Evaluation of **failover time**

- Emulate a cloud provider / CDN

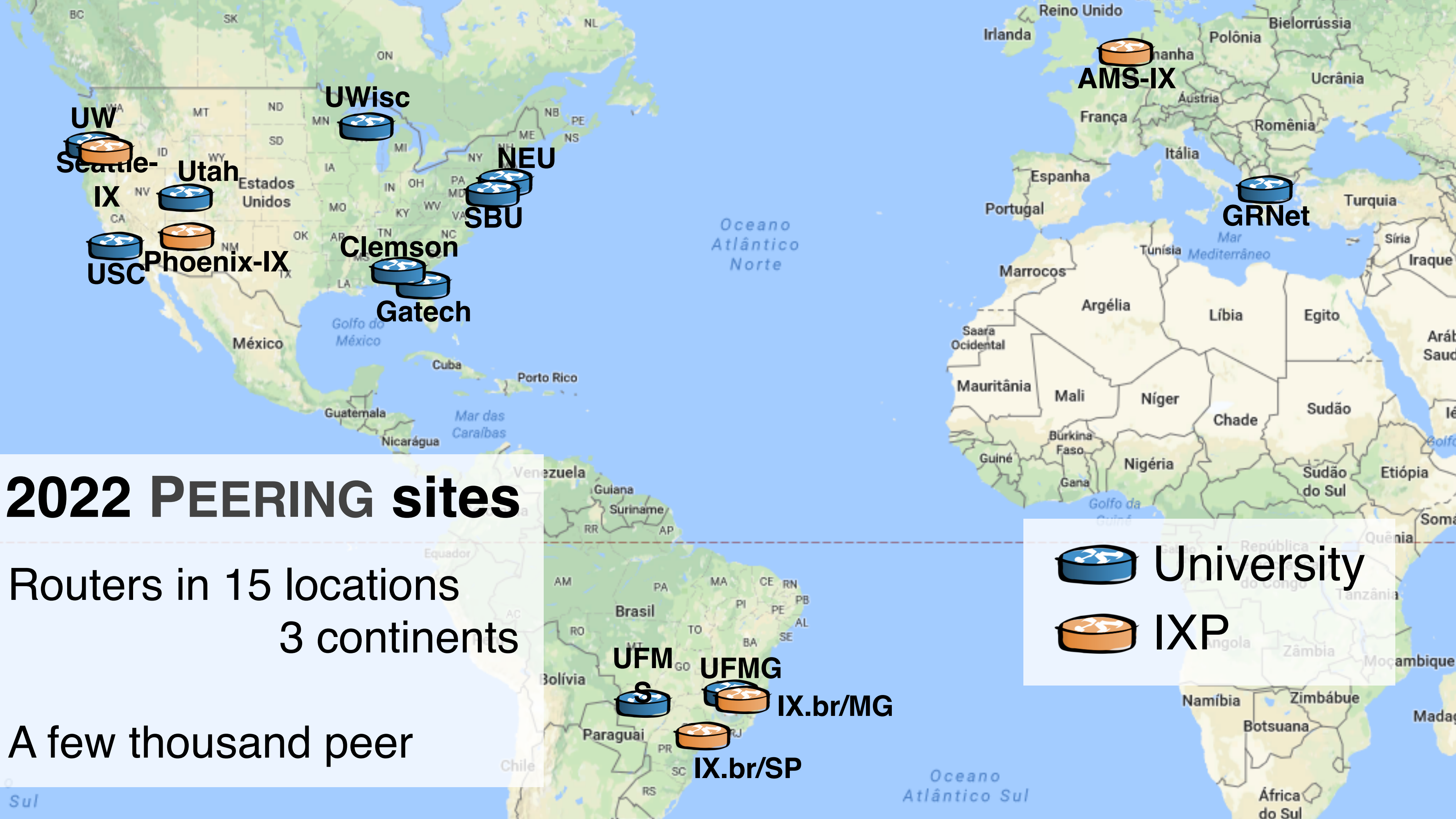
Updates on two community resources — please use them!

PEERING BGP testbed

- Exchange BGP routes and traffic with thousands of ASes at locations around the world

Residential traffic traces

- Packet traces from ~1000 residences
  - Plan to scale to 8000 units, 24x7

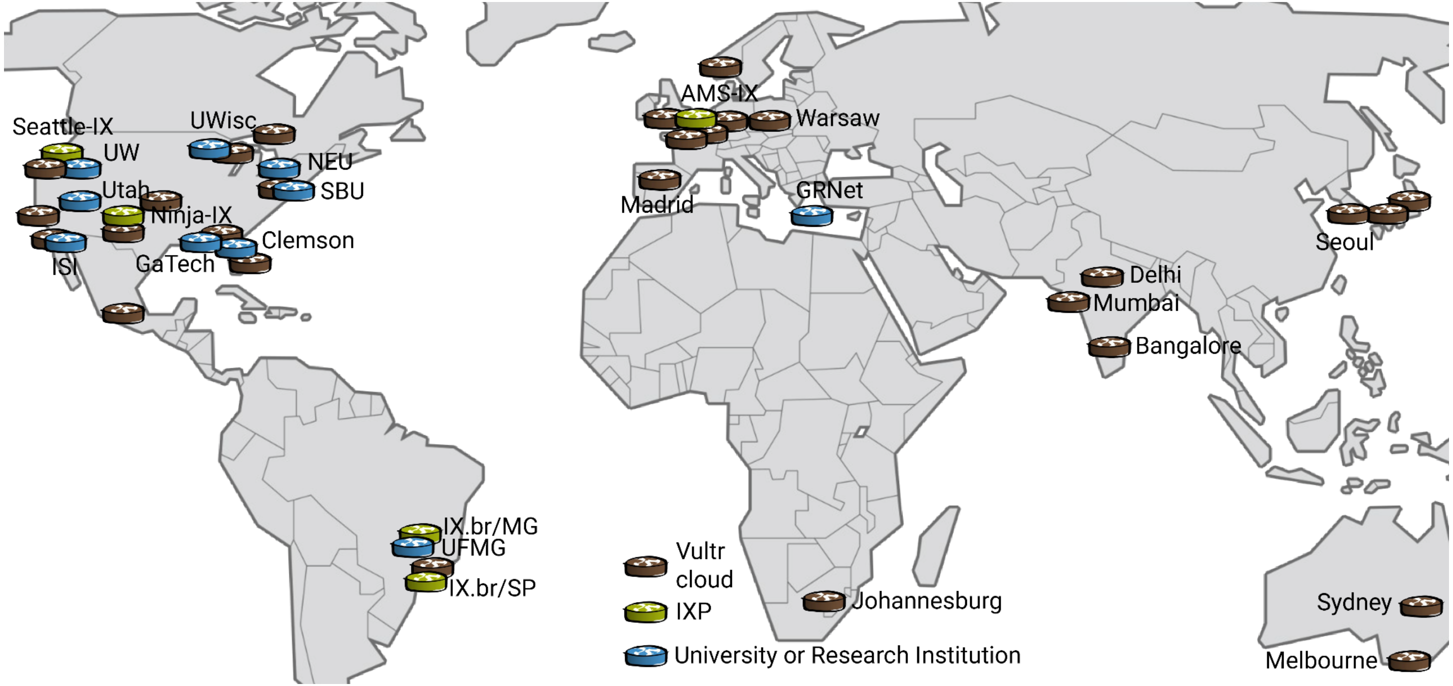


2022 PEERING sites

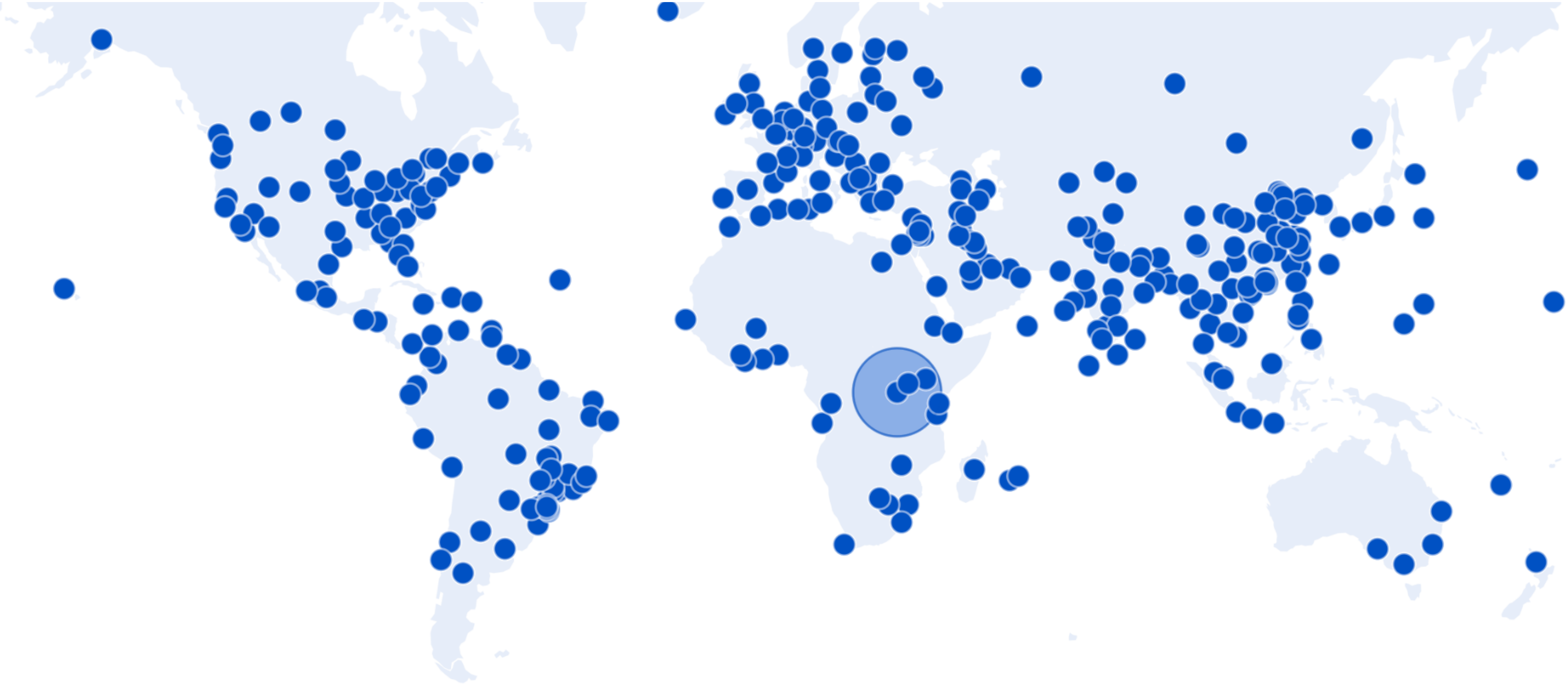
Routers in 15 locations
3 continents

A few thousand peer

PEERING sites - Deployed on Vultr data centers



PEERING sites - Announce from Cloudflare PoPs



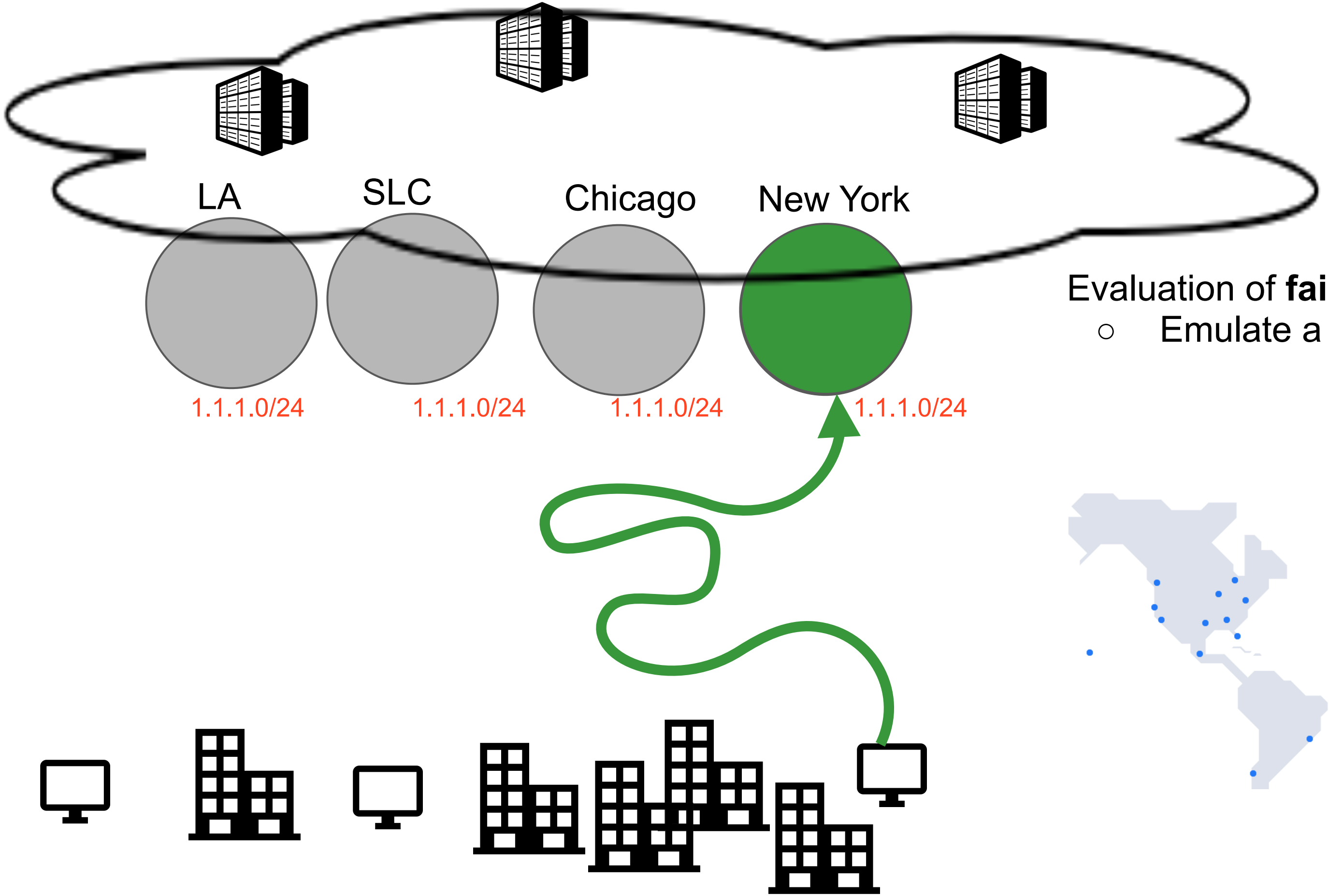
PEERING site capabilities

	# sites	# neighbor ASes	exchange traffic	control BGP announcements	select outgoing routes
universities	10	~10	Y	Y	Y
IXPs	5	~1500	Y	Y	Y
Vultr	32	~6000	Y	Y	N
Cloudflare	335	~13,000	Y	N	N

Data collection

- Looking Glass on PEERING routers so experimenters can view routes
    - Especially useful for debugging your own experiments to check your own experiments
- Traceroutes:
    - 48 teams of 400 RIPE Atlas probes run traceroute to PEERING prefixes every 20 minutes
    - Can configure exact source probes and destination PEERING prefixes/addresses
- Route monitoring
    - Monitor route visibility of PEERING announcements from RIPE RIS
    - https://github.com/PEERINGTestbed/peeringmon_exporter
- TODO: Feed routes to RouteViews/RIS/GIII
    - Announcements that experiments make
    - Routes we learn from the Internet

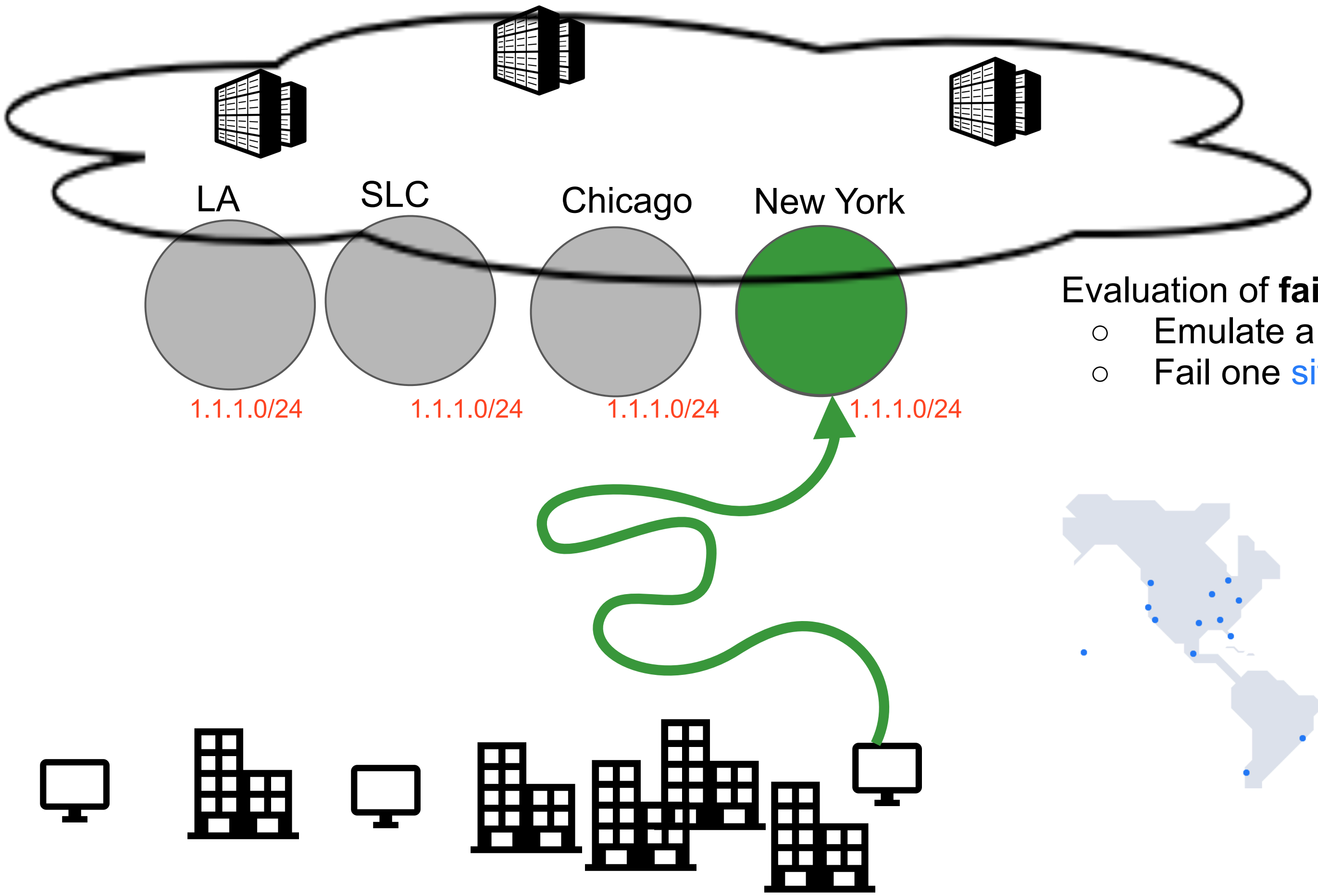
Measuring anycast failover



Evaluation of **failover time**

- Emulate a cloud provider / CDN

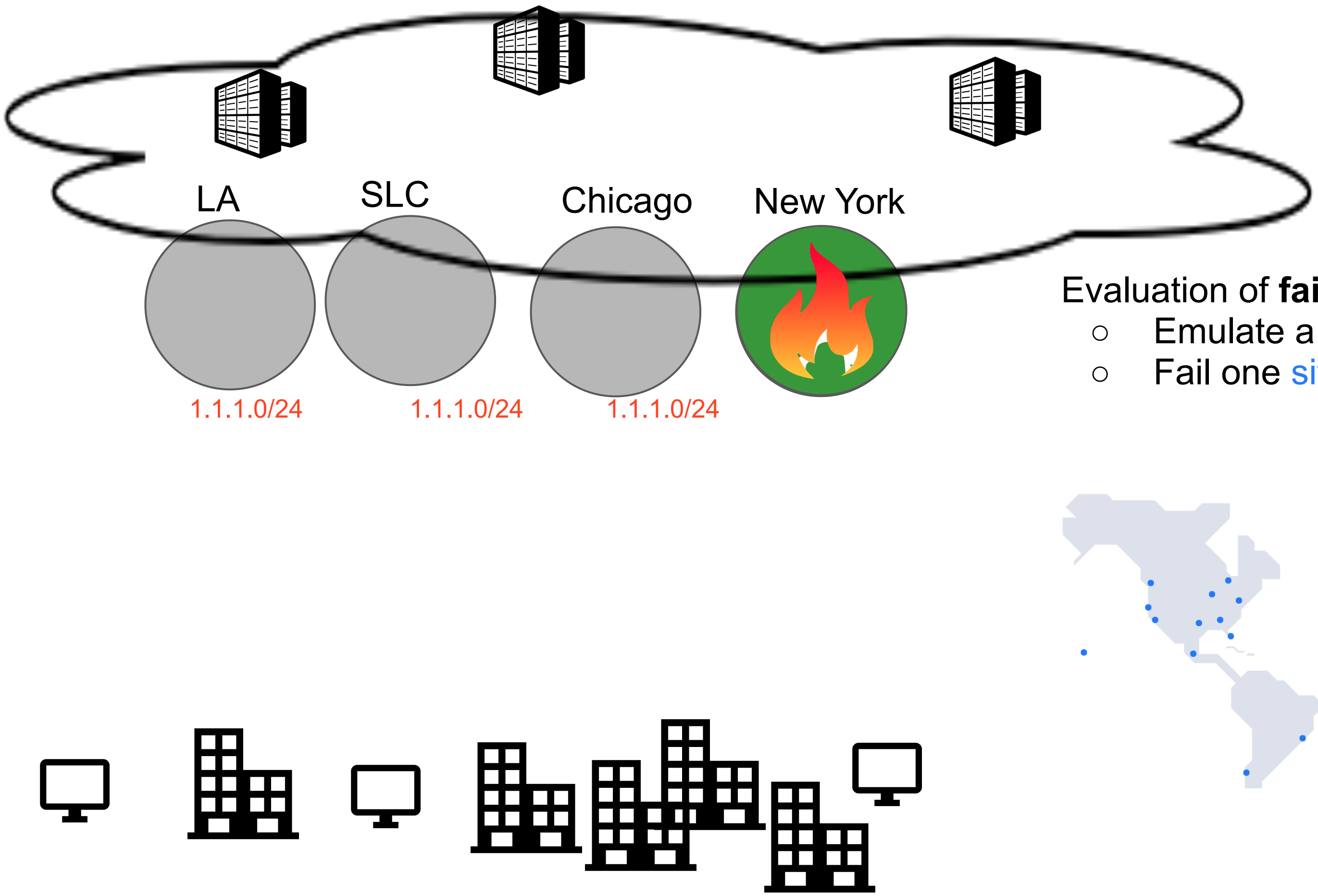
Measuring anycast failover



Evaluation of **failover time**

- Emulate a cloud provider / CDN
- Fail one site at a time

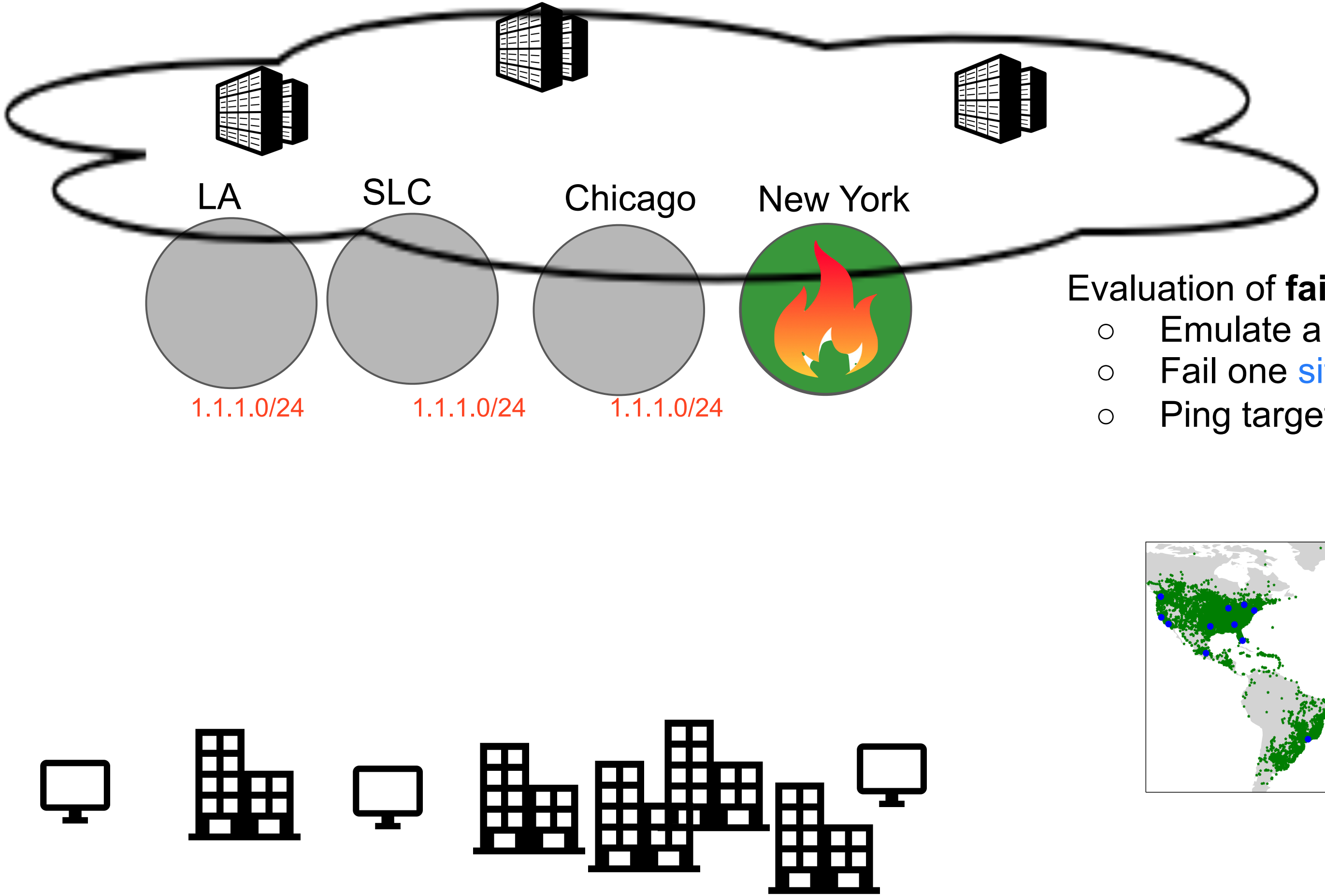
Measuring anycast failover



Evaluation of **failover time**

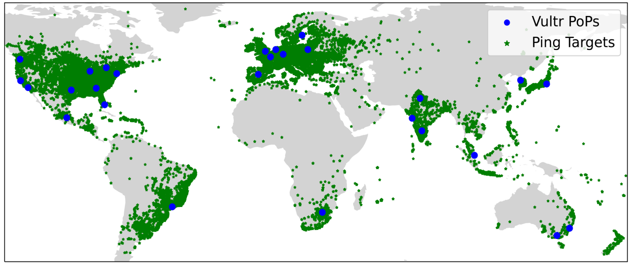
- Emulate a cloud provider / CDN
- Fail one site at a time

Measuring anycast failover

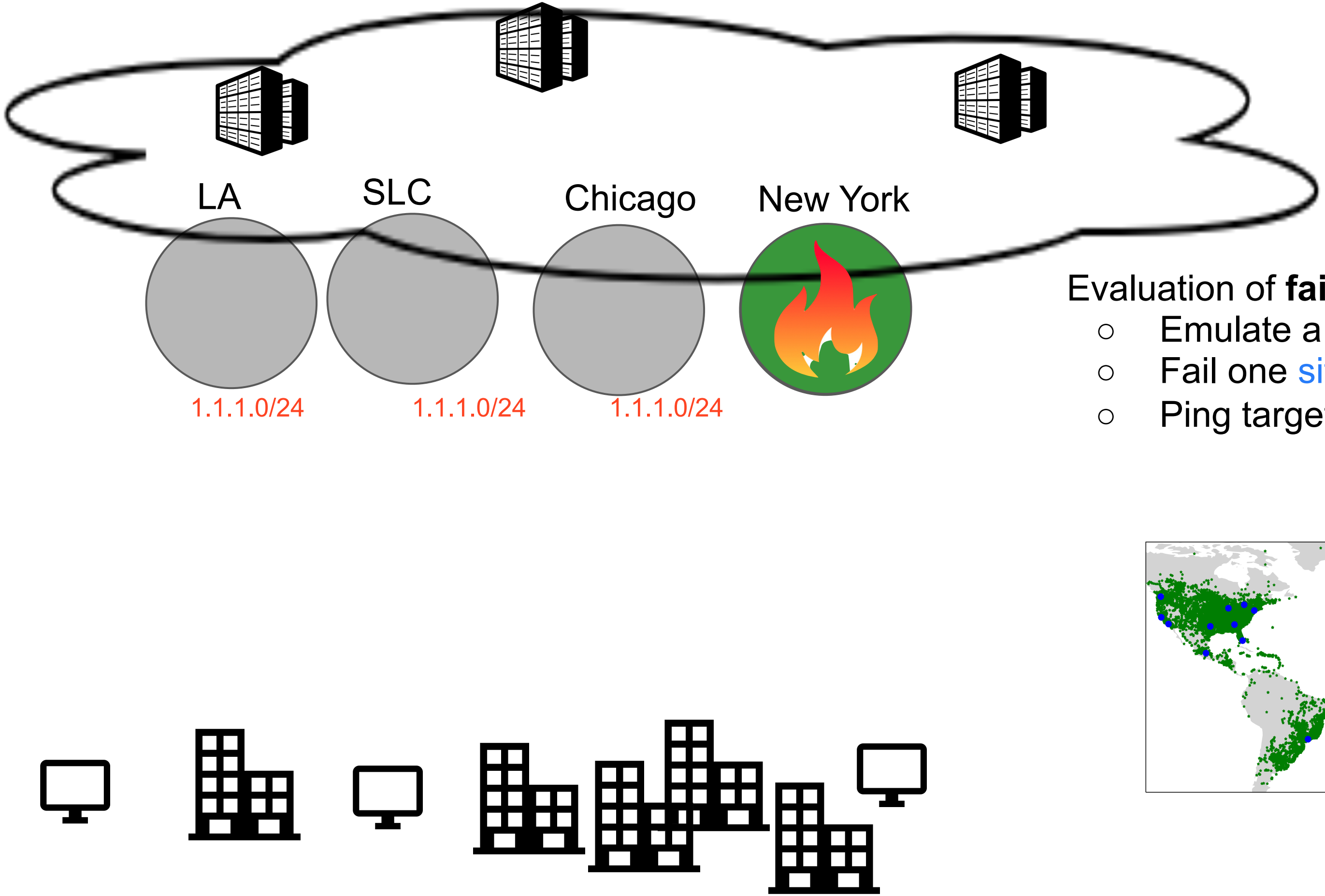


Evaluation of **failover time**

- Emulate a cloud provider / CDN
- Fail one site at a time
- Ping target "clients" to measure failover

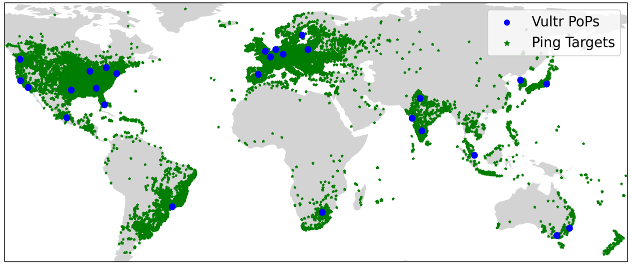


Measuring anycast failover

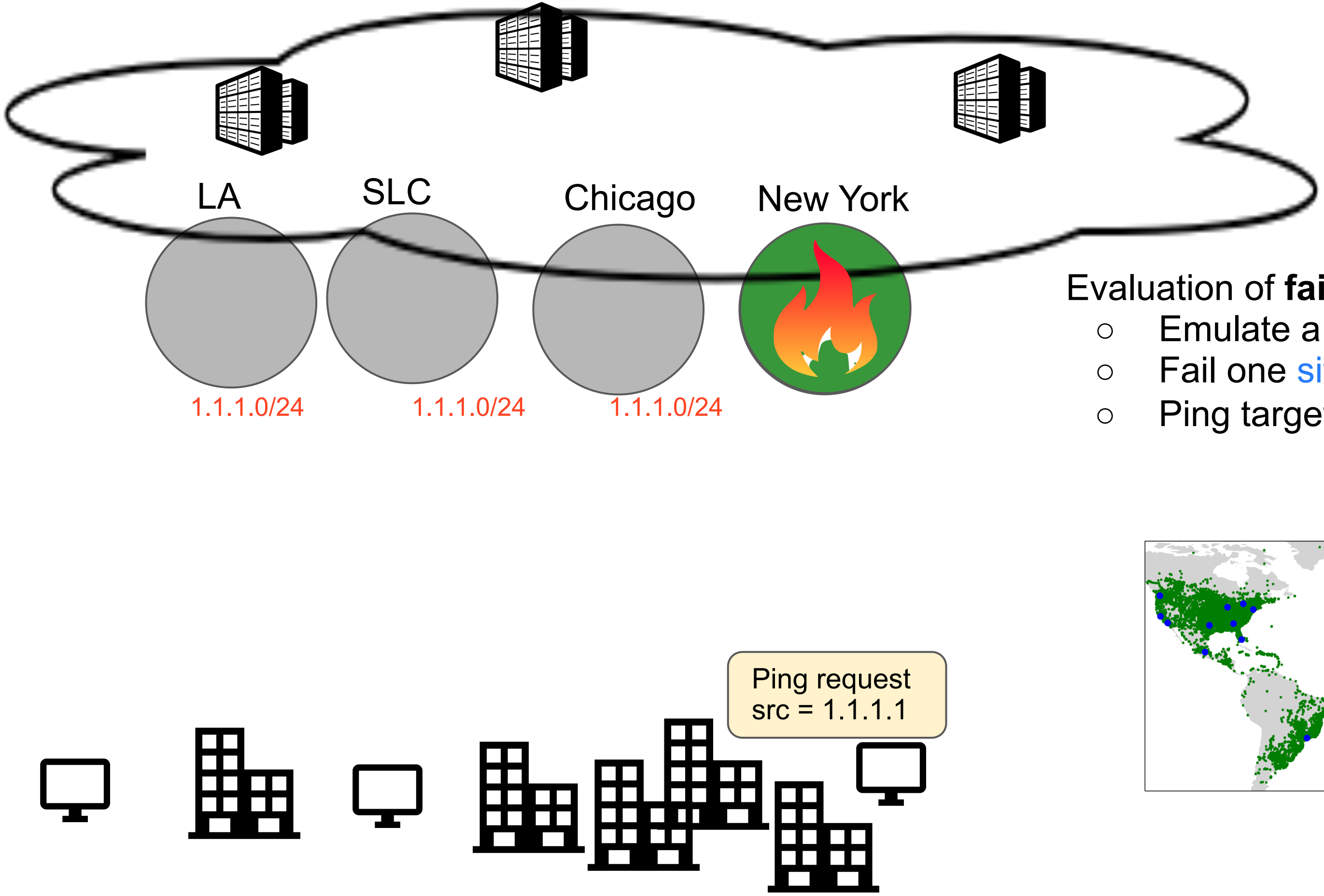


Evaluation of **failover time**

- Emulate a cloud provider / CDN
- Fail one site at a time
- Ping target “clients” to measure failover

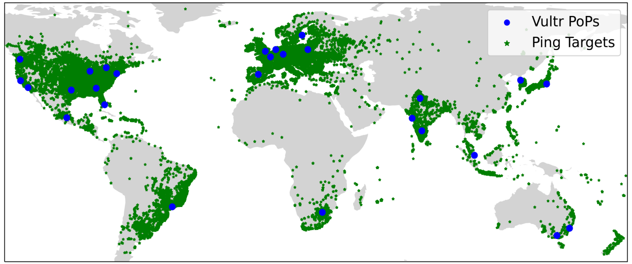


Measuring anycast failover

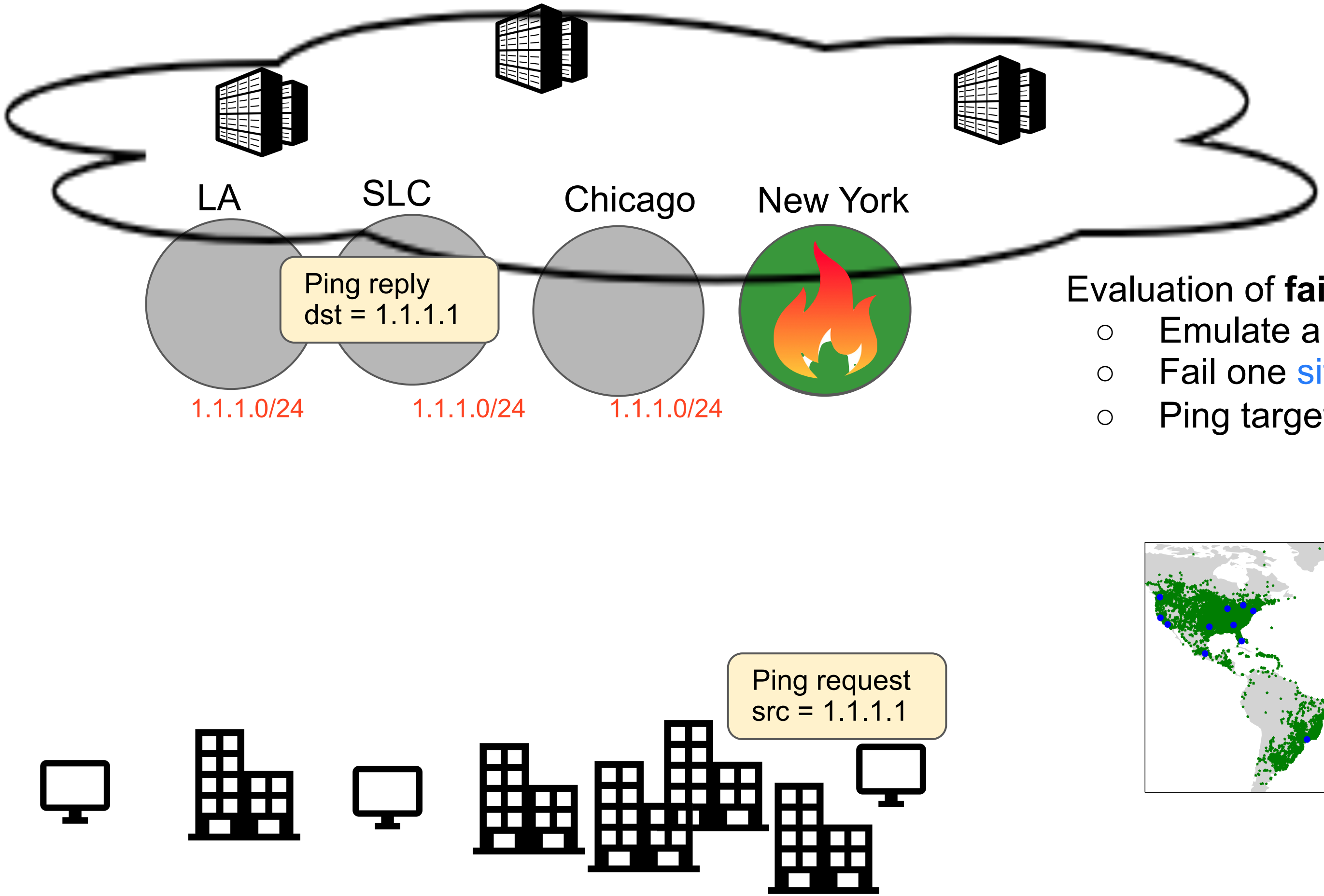


Evaluation of **failover time**

- Emulate a cloud provider / CDN
- Fail one site at a time
- Ping target “clients” to measure failover

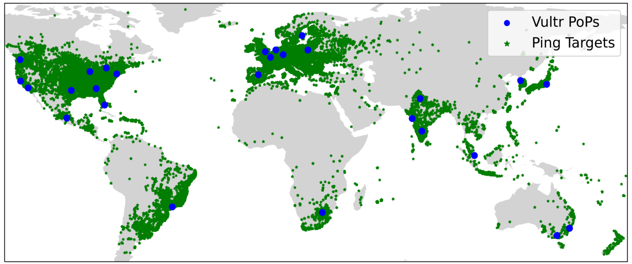


Measuring anycast failover

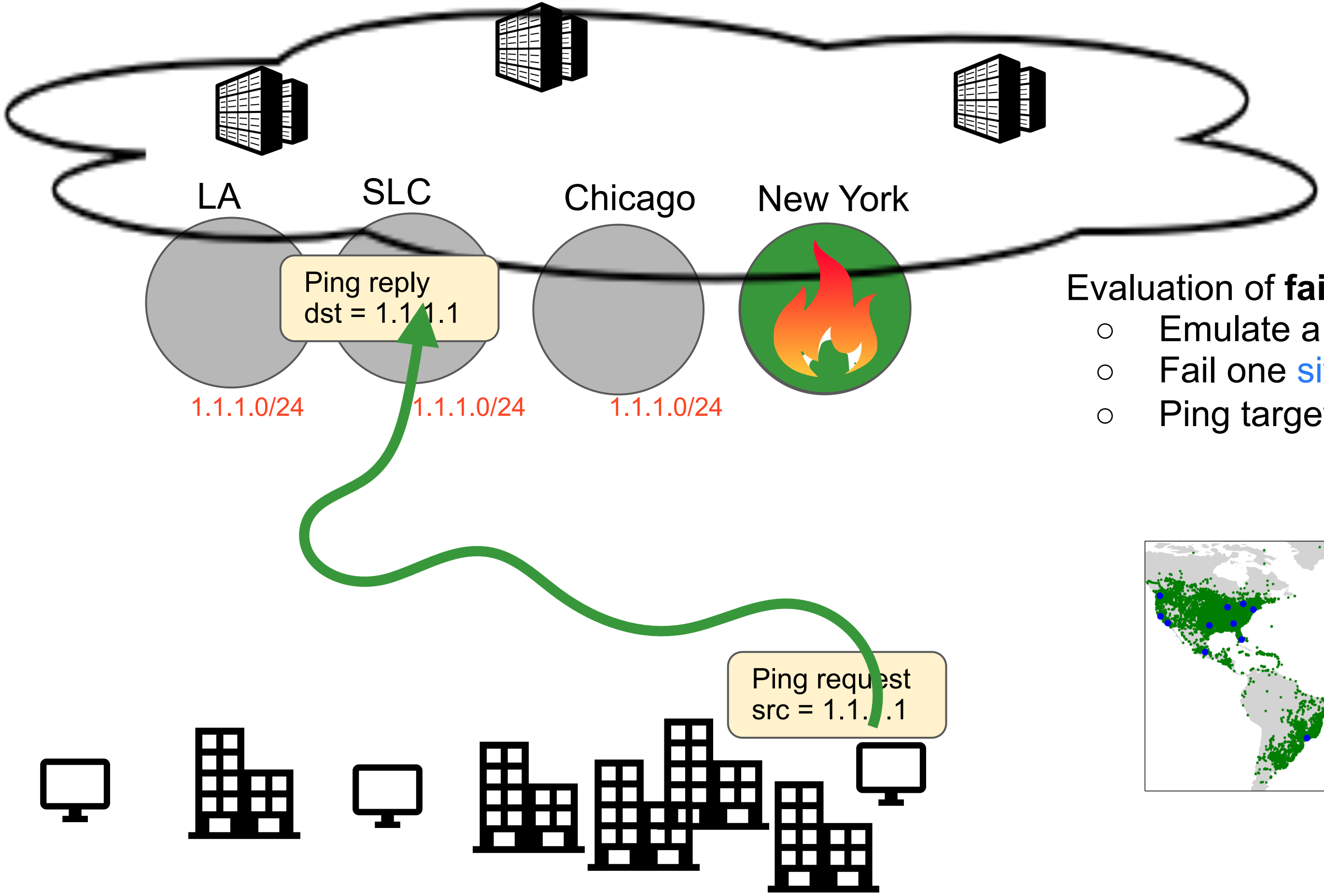


Evaluation of **failover time**

- Emulate a cloud provider / CDN
- Fail one site at a time
- Ping target “clients” to measure failover

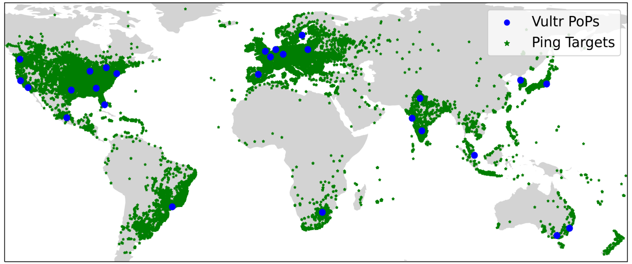


Measuring anycast failover

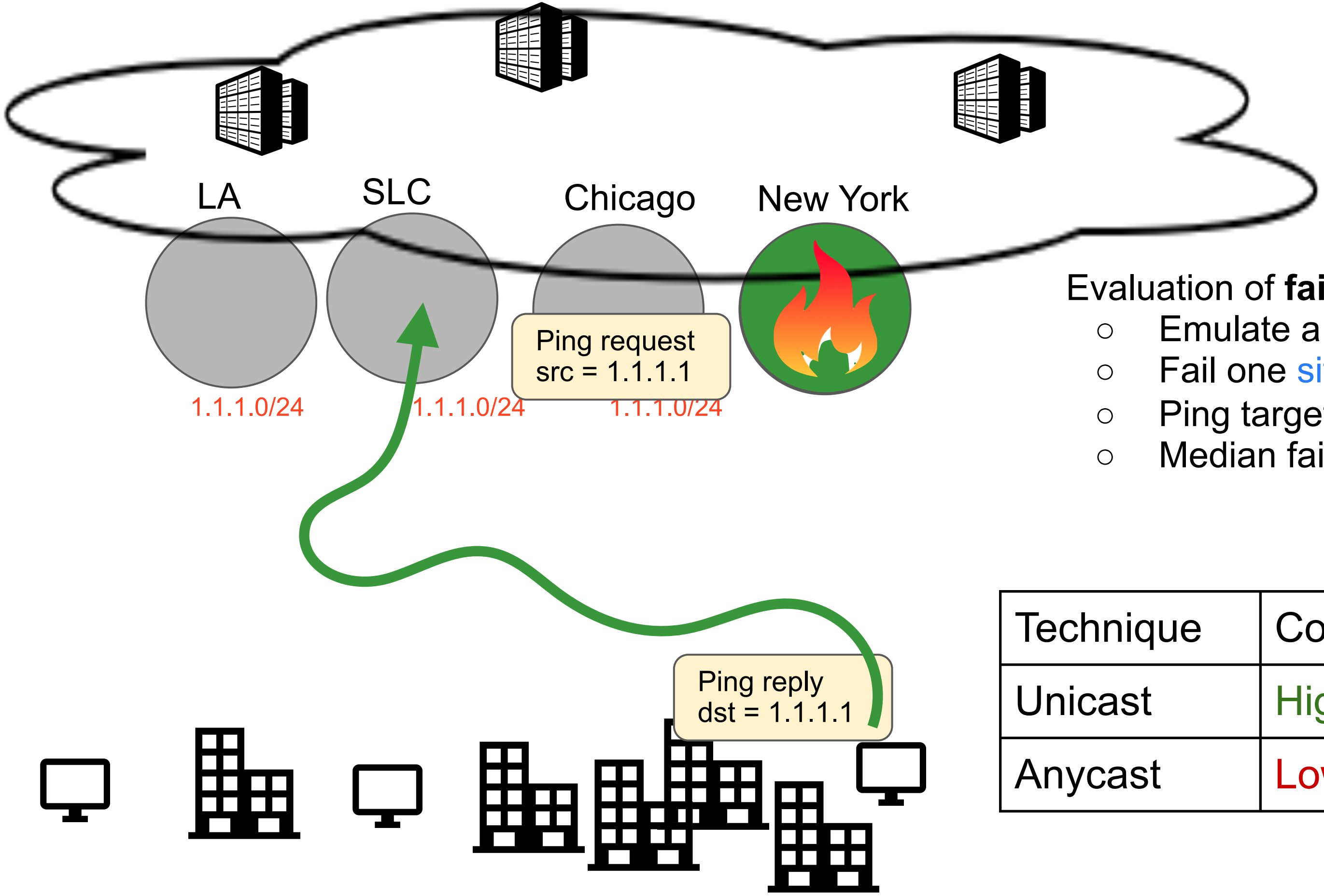


Evaluation of **failover time**

- Emulate a cloud provider / CDN
- Fail one site at a time
- Ping target "clients" to measure failover



Measuring anycast failover

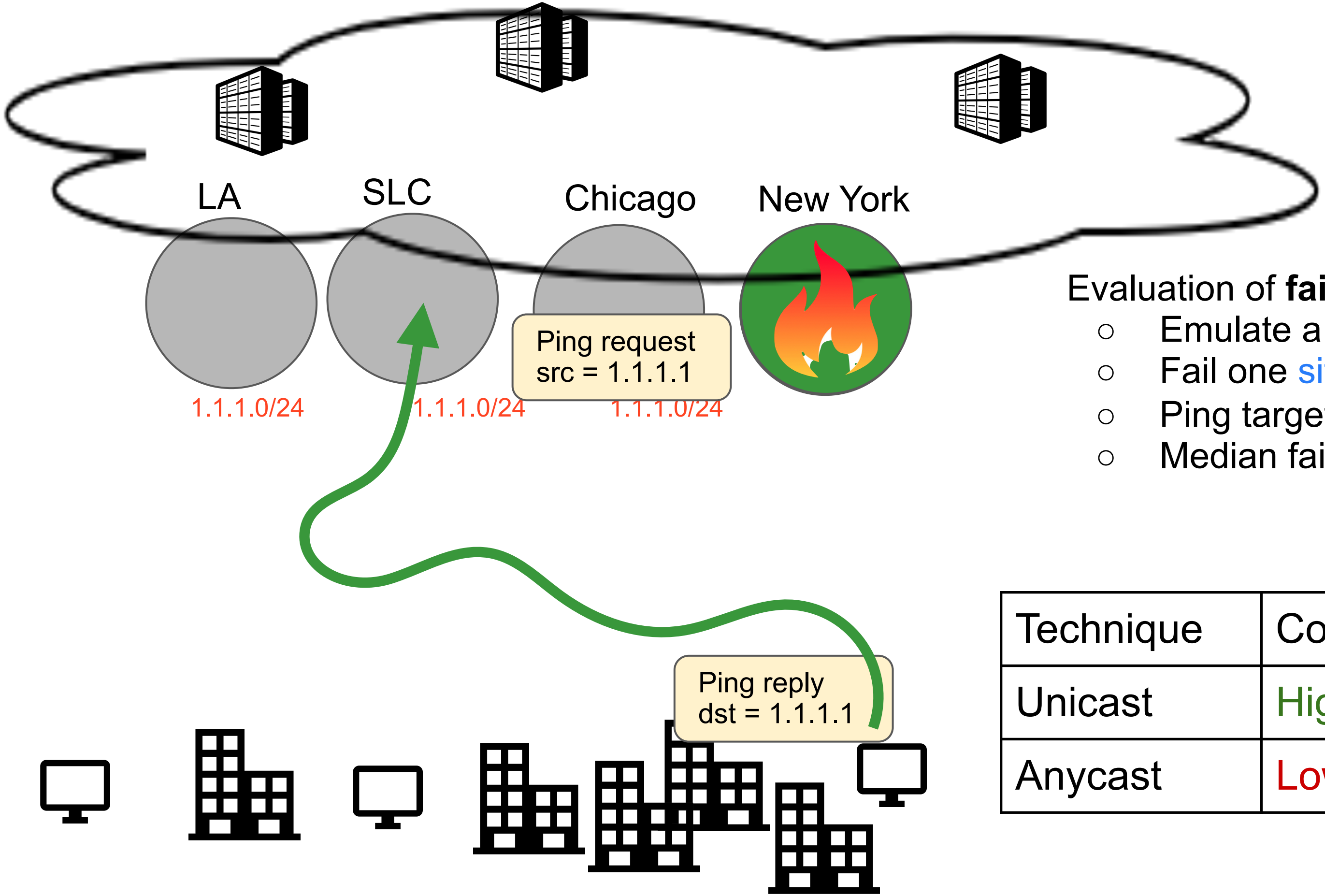


Evaluation of **failover time**

- Emulate a cloud provider / CDN
- Fail one site at a time
- Ping target "clients" to measure failover
- Median failover < 2 seconds

Technique	Control	Availability
Unicast	High	Low
Anycast	Low	

Measuring anycast failover



Evaluation of **failover time**

- Emulate a cloud provider / CDN
- Fail one site at a time
- Ping target "clients" to measure failover
- Median failover < 2 seconds

Technique	Control	Availability
Unicast	High	Low
Anycast	Low	High

These are community resources — please use them!

PEERING BGP testbed

Residential traffic traces

These are community resources — please use them!

PEERING BGP testbed

- Submit proposal at peering.ee.columbia.edu
- We approve and allocate you prefixes (or bring your own)

Residential traffic traces

These are community resources — please use them!

PEERING BGP testbed

- Submit proposal at peering.ee.columbia.edu
- We approve and allocate you prefixes (or bring your own)
- Set up BGP sessions with our routers to:
    - Make announcements
    - Receive and select routes
    - Exchange traffic
- Filters restrict announcements to approved capabilities for experiment

Residential traffic traces

These are community resources — please use them!

PEERING BGP testbed

- Submit proposal at peering.ee.columbia.edu
- We approve and allocate you prefixes
  (or bring your own)
- Set up BGP sessions with our routers to:
  - Make announcements
  - Receive and select routes
  - Exchange traffic
- Filters restrict announcements to
  approved capabilities for experiment
- Popular use cases include RPKI measurements:
  ImpROV: Measurement and Practical Mitigation of
  Collateral Damage of RPKI Route Origin Validation.
  Weitong Li, Yuze Li, **Taejoong Chung**.
  USENIX Security Symposium 2025

Residential traffic traces

These are community resources — please use them!

PEERING BGP testbed

- Submit proposal at peering.ee.columbia.edu
- We approve and allocate you prefixes (or bring your own)
- Set up BGP sessions with our routers to:
  - Make announcements
  - Receive and select routes
  - Exchange traffic
- Filters restrict announcements to approved capabilities for experiment
- Popular use cases include RPKI measurements: ImpROV: Measurement and Practical Mitigation of Collateral Damage of RPKI Route Origin Validation. Weitong Li, Yuze Li, **Taejoong Chung**. USENIX Security Symposium 2025

Residential traffic traces

- Collecting since 2023 and plan to continue indefinitely
- Currently ~1000 units, 4 hrs / day
  - Plan to scale to 8000 units, 24x7

These are community resources — please use them!

PEERING BGP testbed

- Submit proposal at peering.ee.columbia.edu
- We approve and allocate you prefixes (or bring your own)
- Set up BGP sessions with our routers to:
  - Make announcements
  - Receive and select routes
  - Exchange traffic
- Filters restrict announcements to approved capabilities for experiment
- Popular use cases include RPKI measurements: ImpROV: Measurement and Practical Mitigation of Collateral Damage of RPKI Route Origin Validation. Weitong Li, Yuze Li, **Taejoong Chung**. USENIX Security Symposium 2025

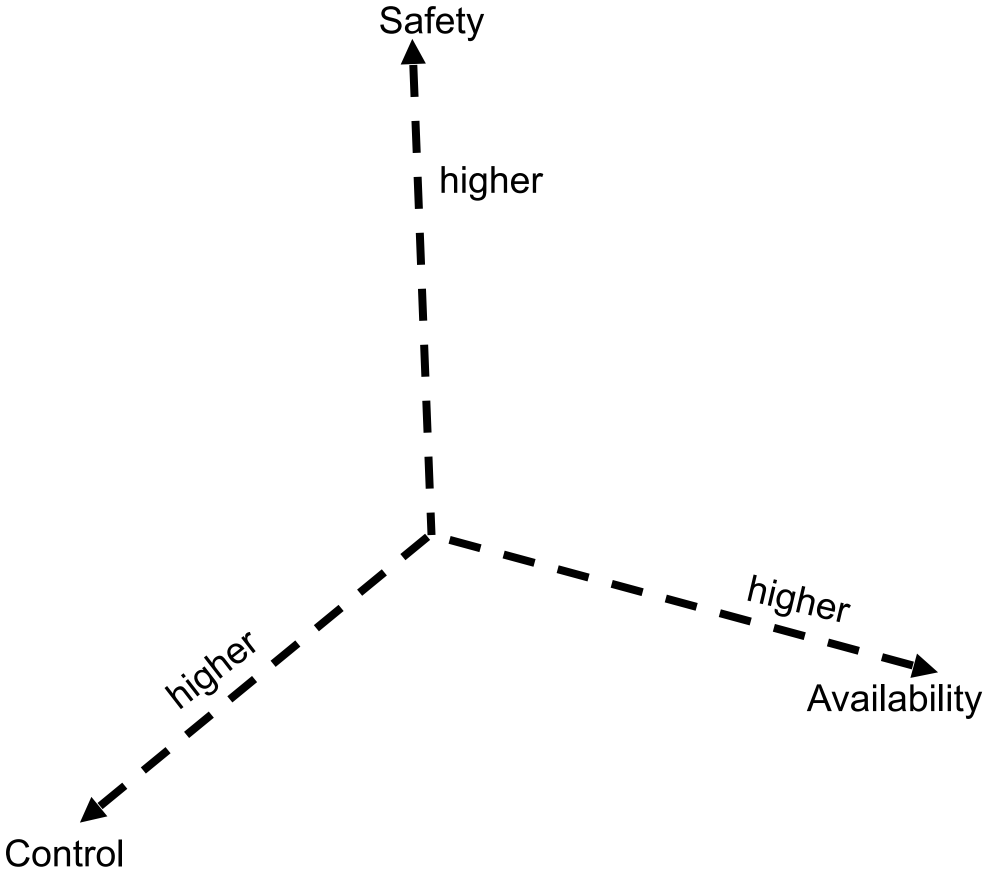
Residential traffic traces

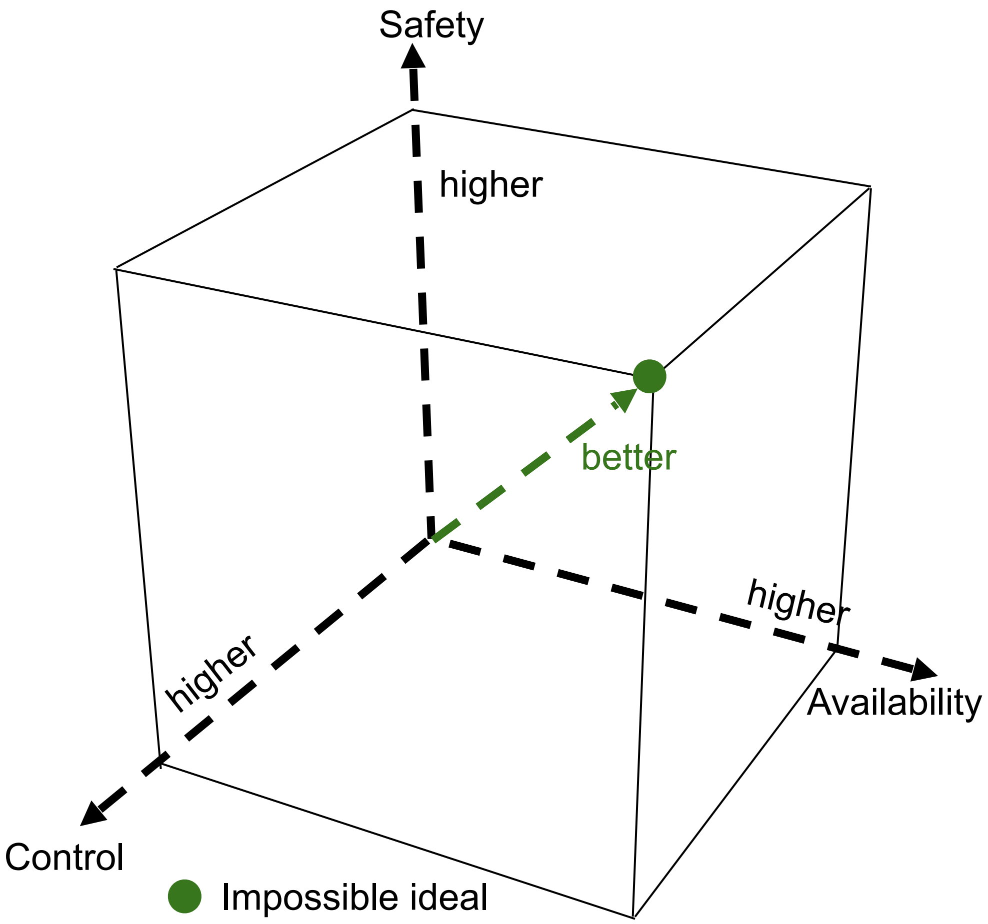
- Collecting since 2023 and plan to continue indefinitely
- Currently ~1000 units, 4 hrs / day
  - Plan to scale to 8000 units, 24x7
- We can share the data
- Submit IRB approval/exemption including description of data needed
- Data aggregated and anonymized as appropriate
  - Flows or packets
  - Individual (anonymized) units (rotating anonymization key), or truncated by prefix

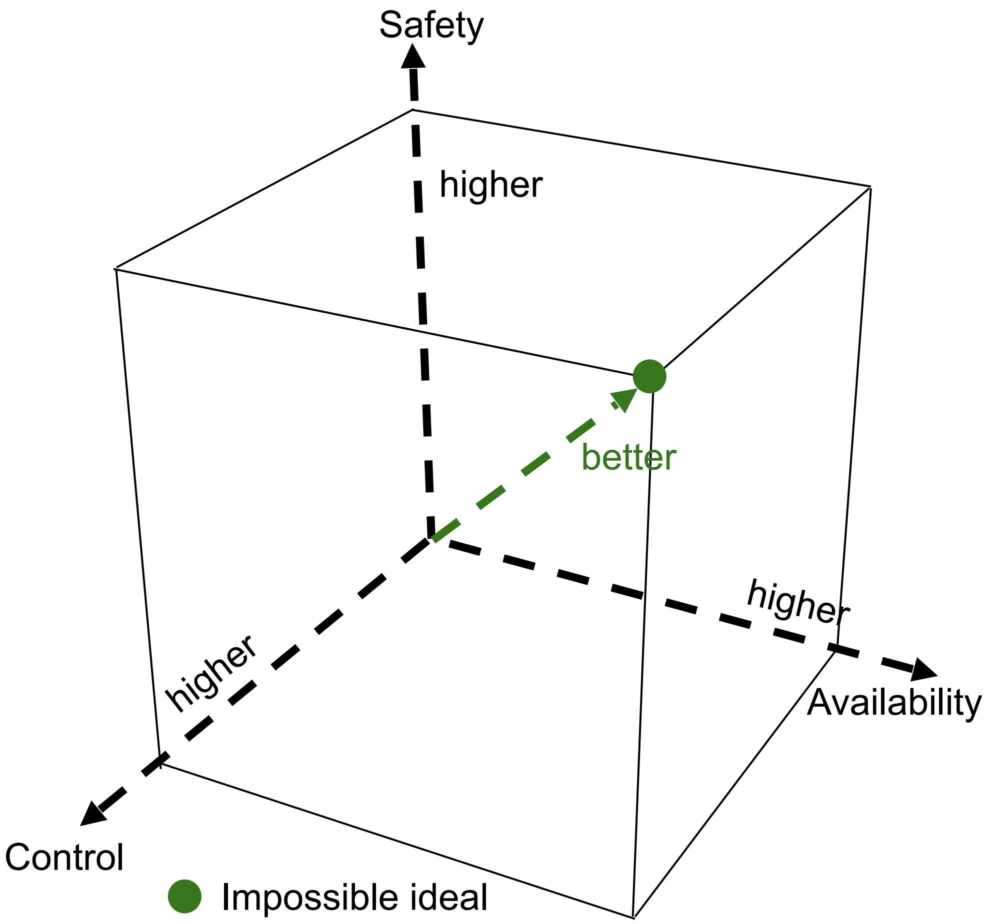
Fundamental tradeoffs in cloud/CDN ingress routing

Technique	Control	Availability
Unicast	High	Low
Anycast	Low	High

- Existing techniques compromise control or availability
- Announcing failed site's prefix from other sites upon failure (***reactive*** anycast) runs risk of turning a local failure into a widespread one, compromising ***safety***
- Tradeoffs are fundamental:
  any technique relying on DNS + BGP for content redirection
  must compromise at least one of *control, availability, or safety*

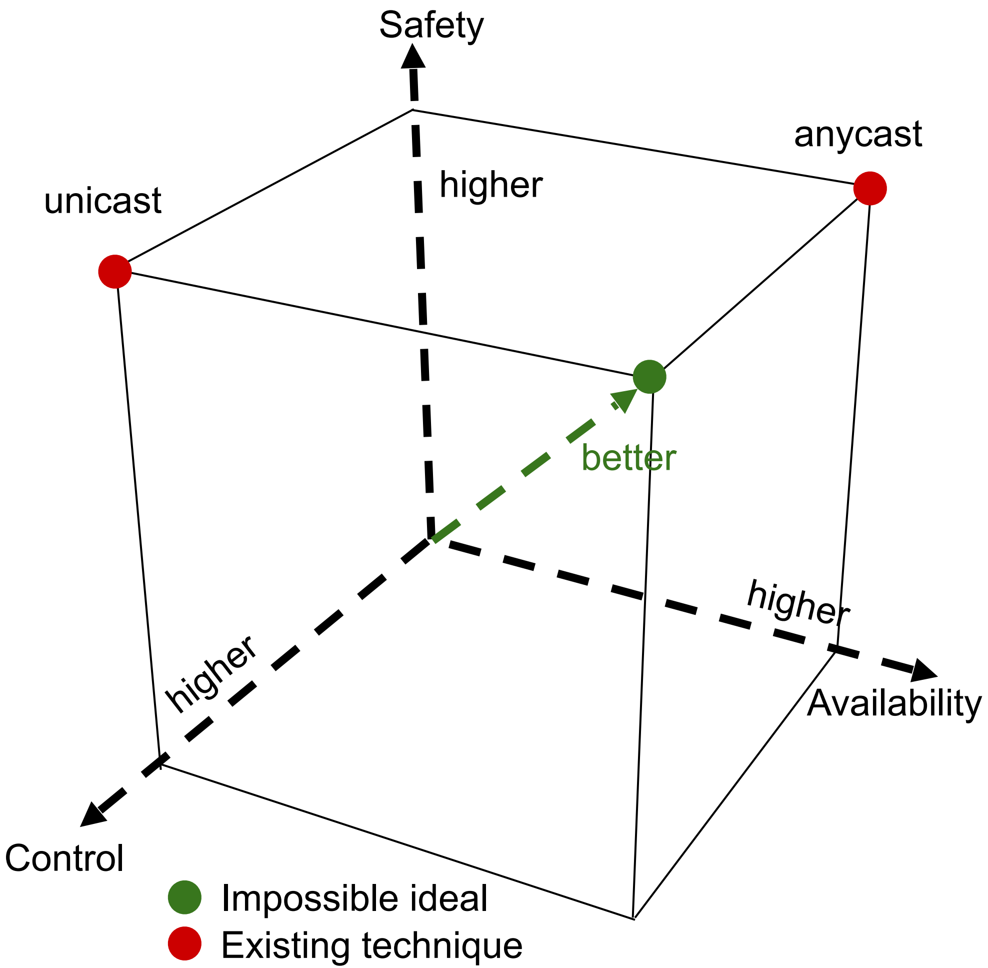

Safety
higher
higher
Availability
higher
Control


Safety
higher
better
higher
Availability
higher
Control
Impossible ideal



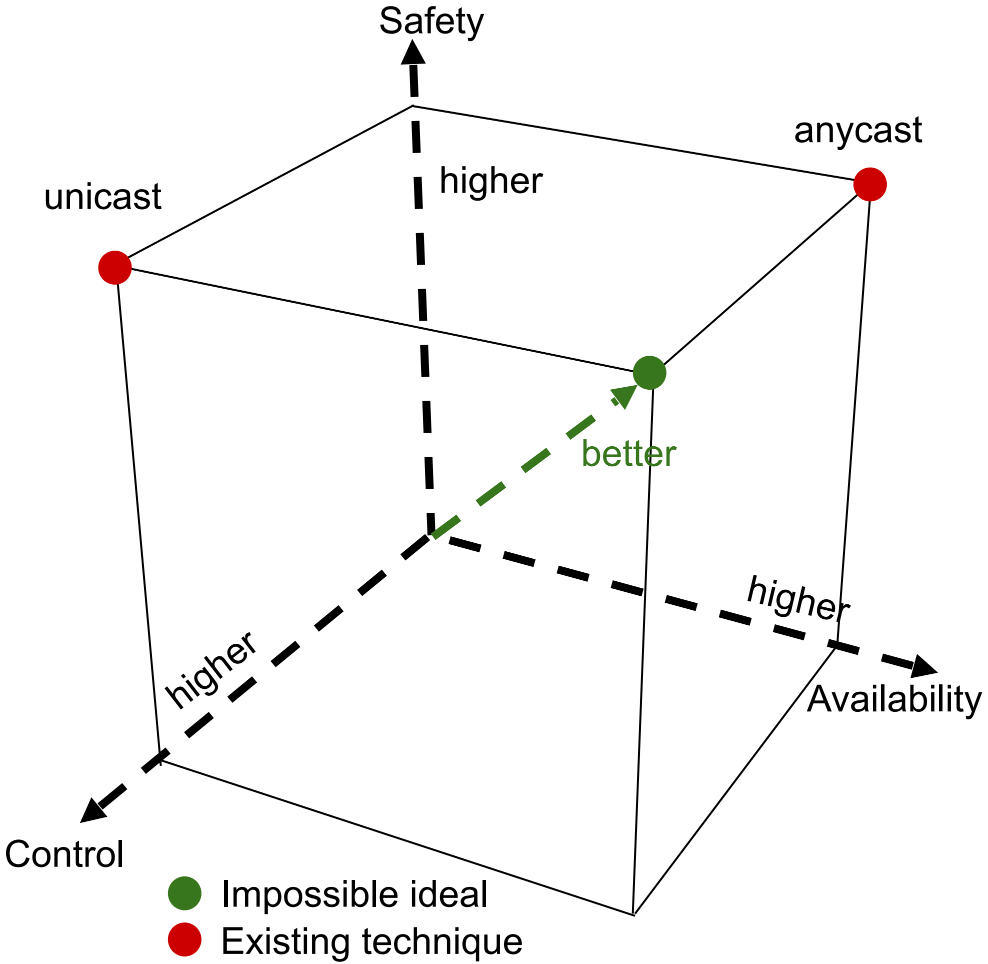
- Existing techniques compromise **control** or **availability**.

Technique	Control	Availability	Safety
Unicast	High	Low	High
Anycast	Low	High	High



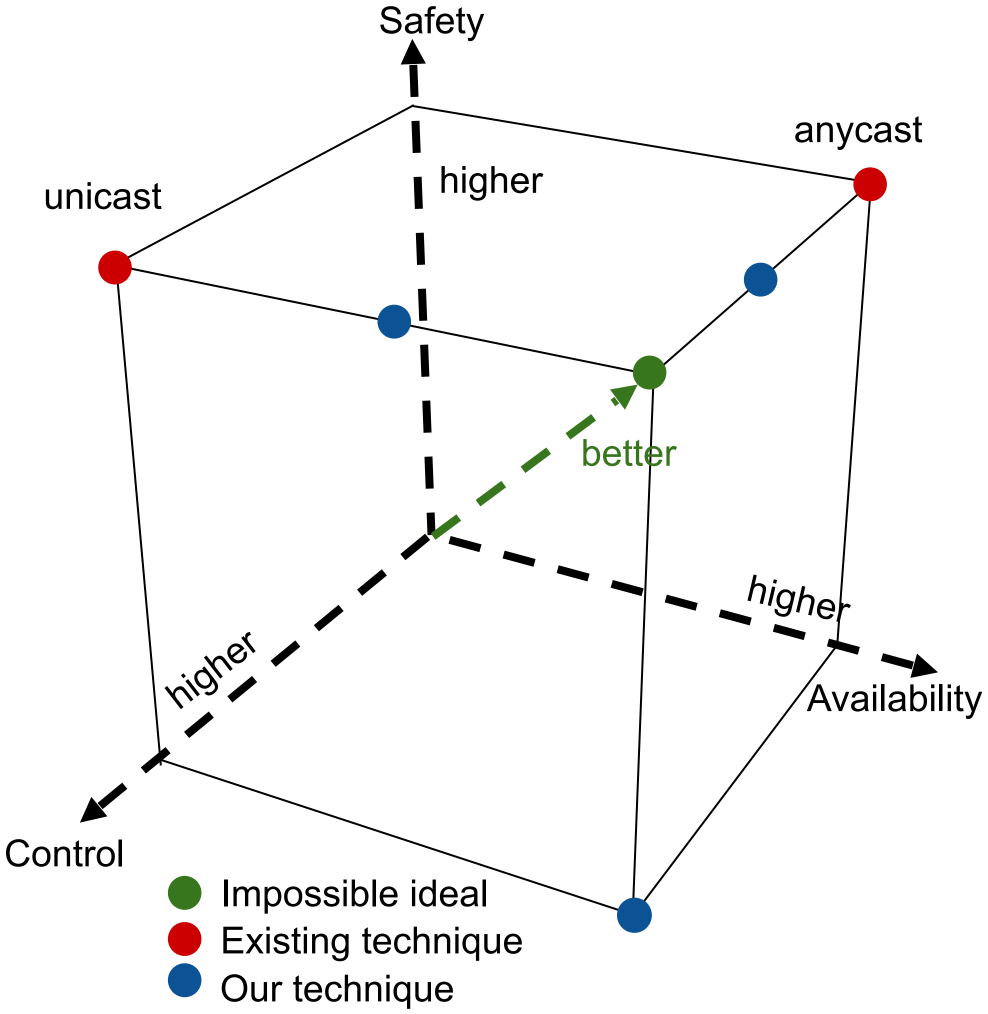
- Existing techniques compromise **control** or **availability**.

Technique	Control	Availability	Safety
Unicast	High	Low	High
Anycast	Low	High	High



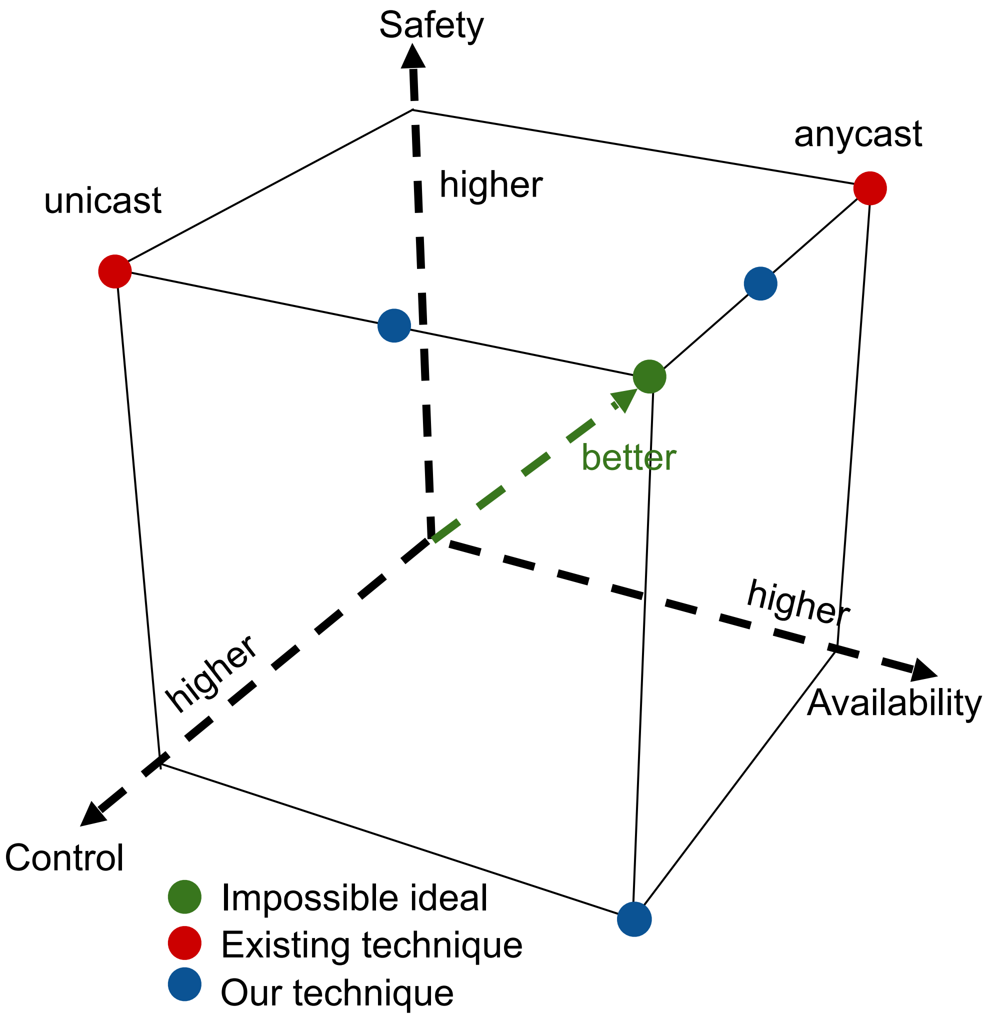
- Existing techniques compromise **control** or **availability**.
- We developed three new techniques.

Technique	Control	Availability	Safety
Unicast	High	Low	High
Anycast	Low	High	High



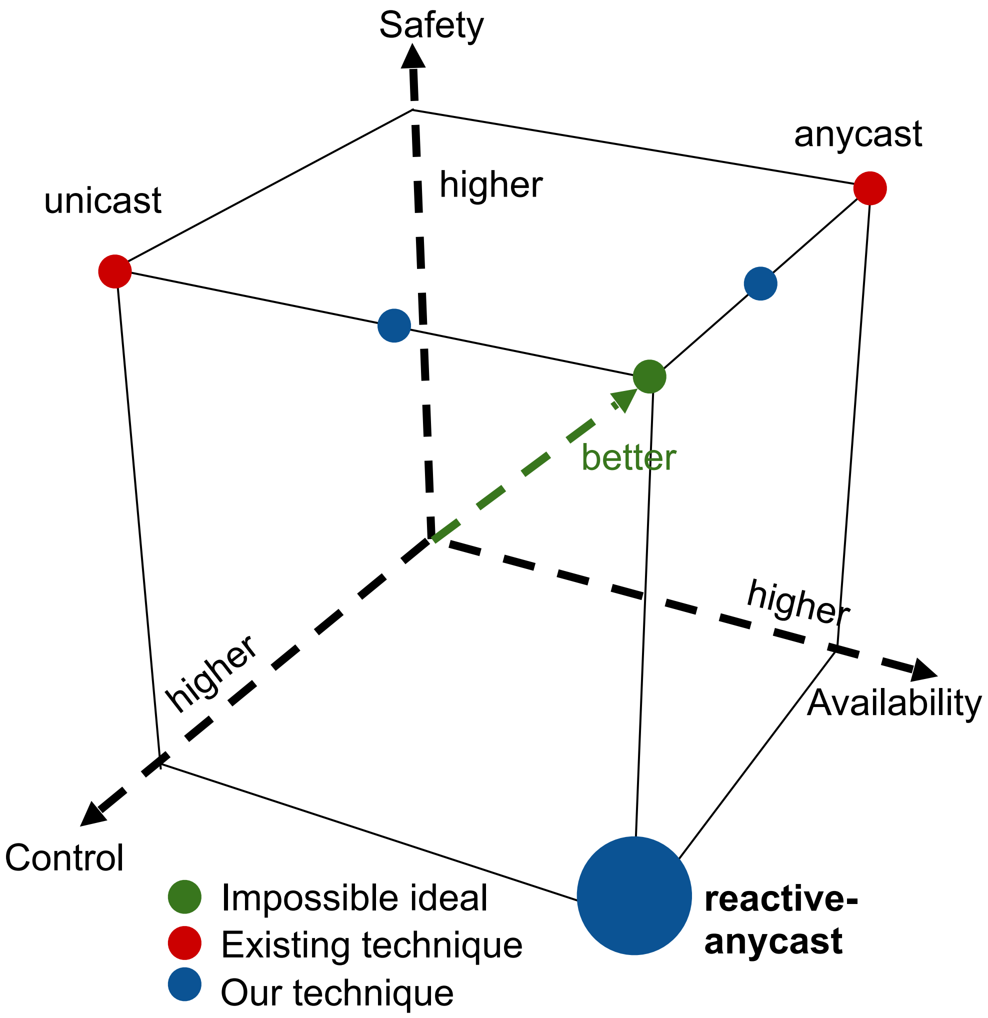
- Existing techniques compromise **control** or **availability**.
- We developed three new techniques.

Technique	Control	Availability	Safety
Unicast	High	Low	High
Anycast	Low	High	High



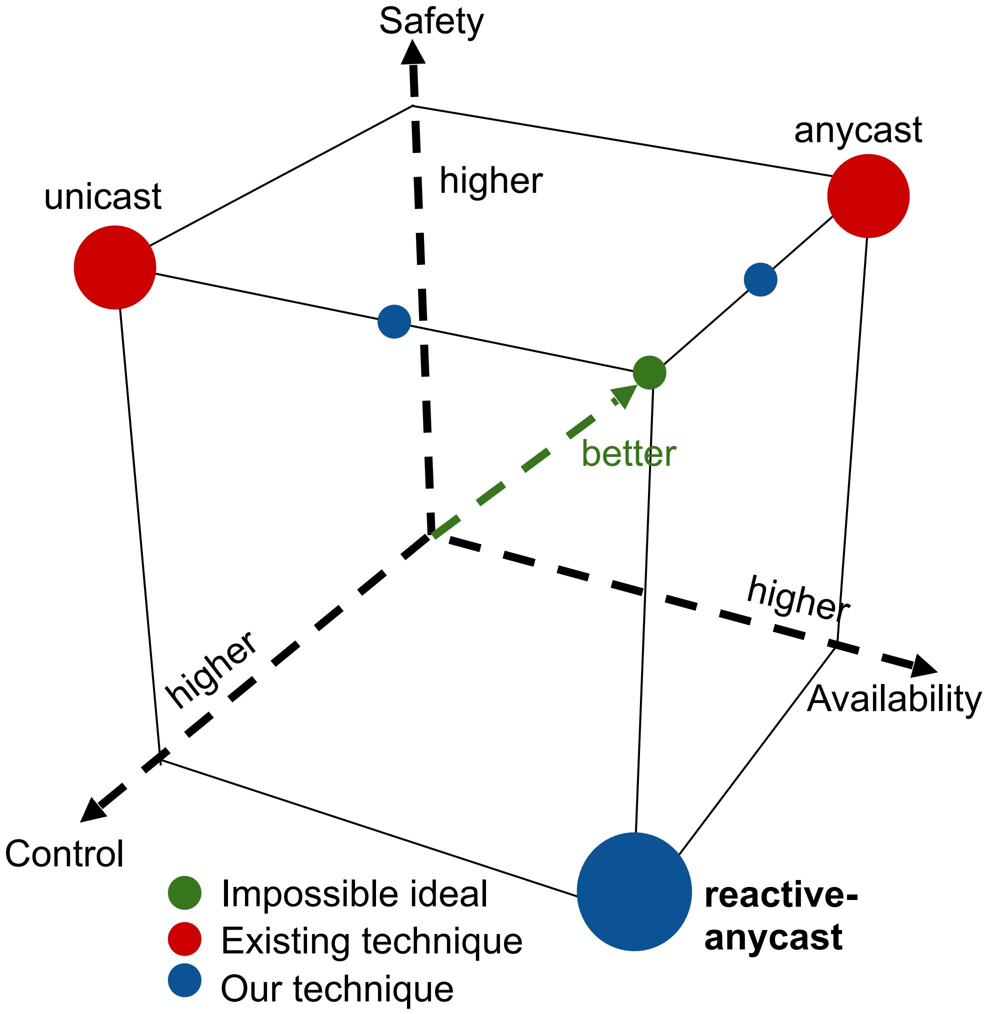
- Existing techniques compromise **control** or **availability**.
- We developed three new techniques.

Technique	Control	Availability	Safety
Unicast	High	Low	High
Anycast	Low	High	High



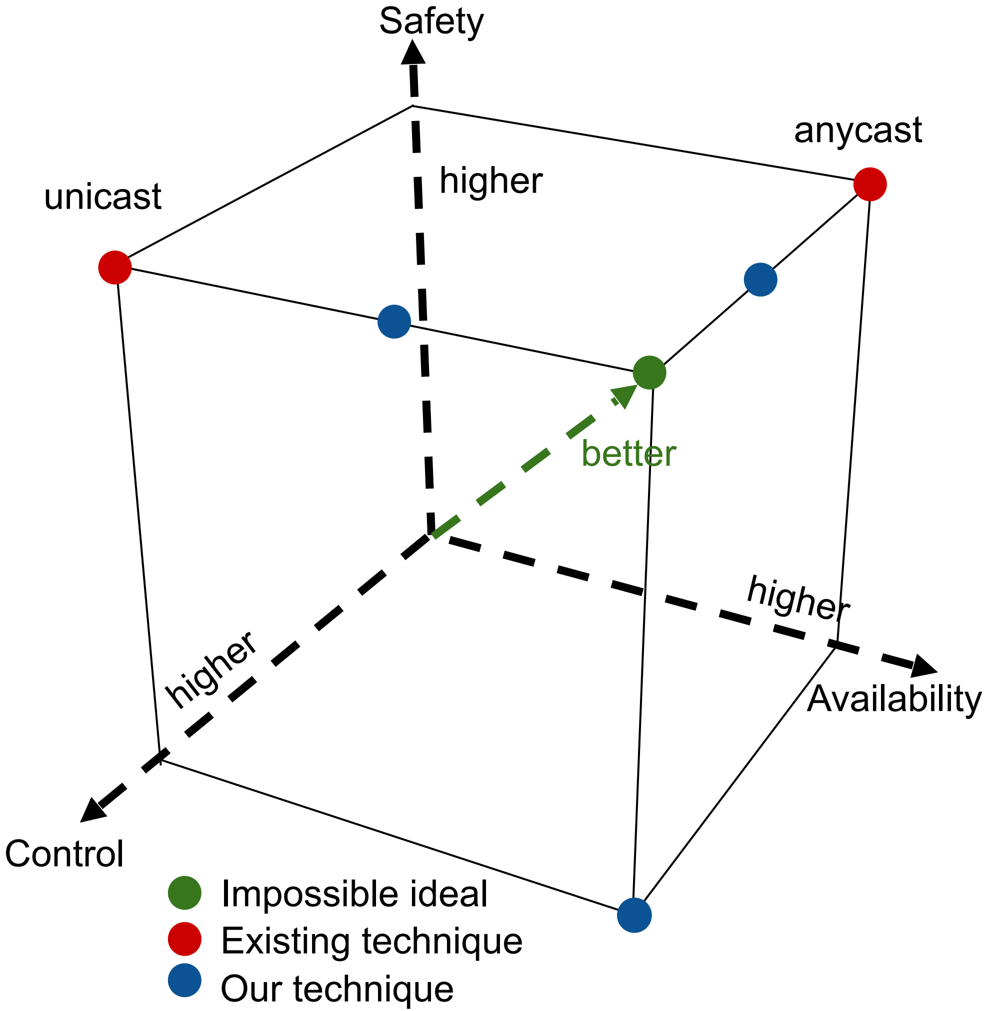
- Existing techniques compromise **control** or **availability**.
- We developed three new techniques.

Technique	Control	Availability	Safety
Unicast	High	Low	High
Anycast	Low	High	High
Reactive	High	High	Low



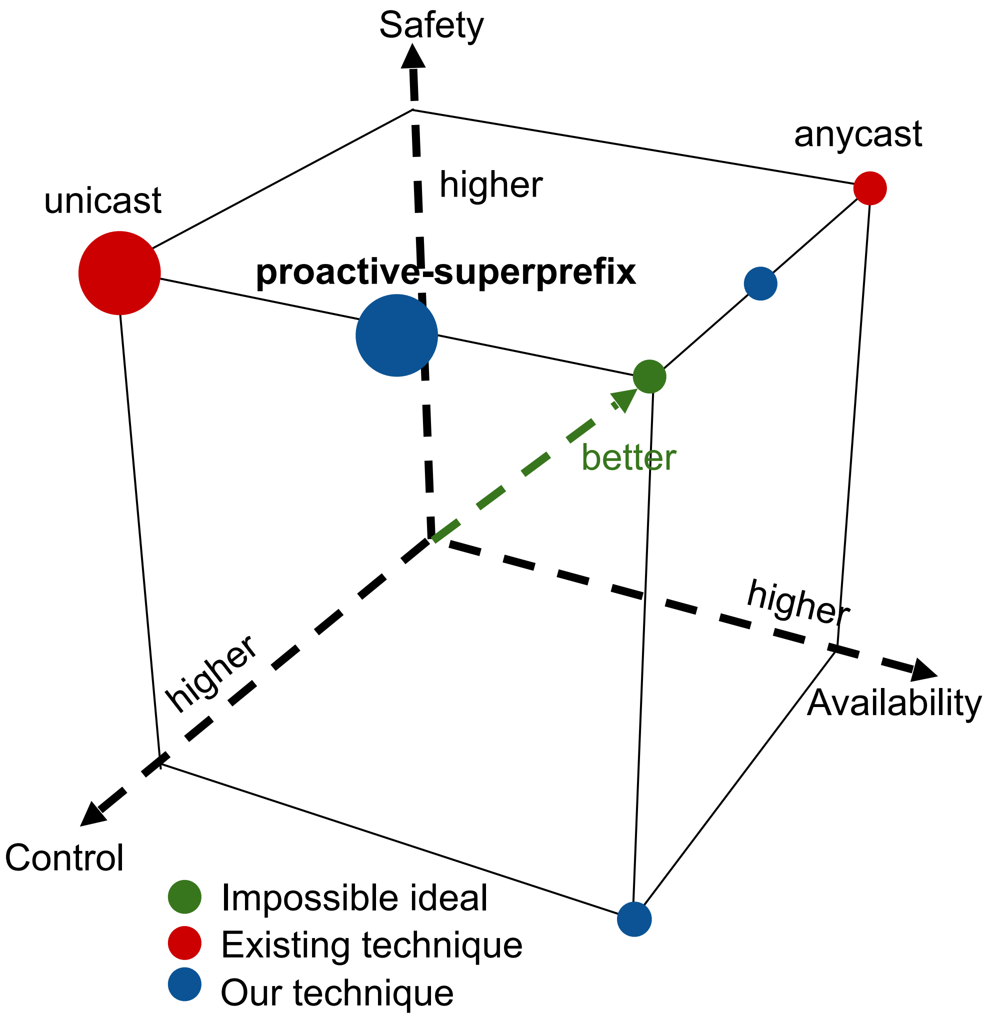
- Existing techniques compromise **control** or **availability**.
- We developed three new techniques.

Technique	Control	Availability	Safety
Unicast	High	Low	High
Anycast	Low	High	High
Reactive	High	High	Low



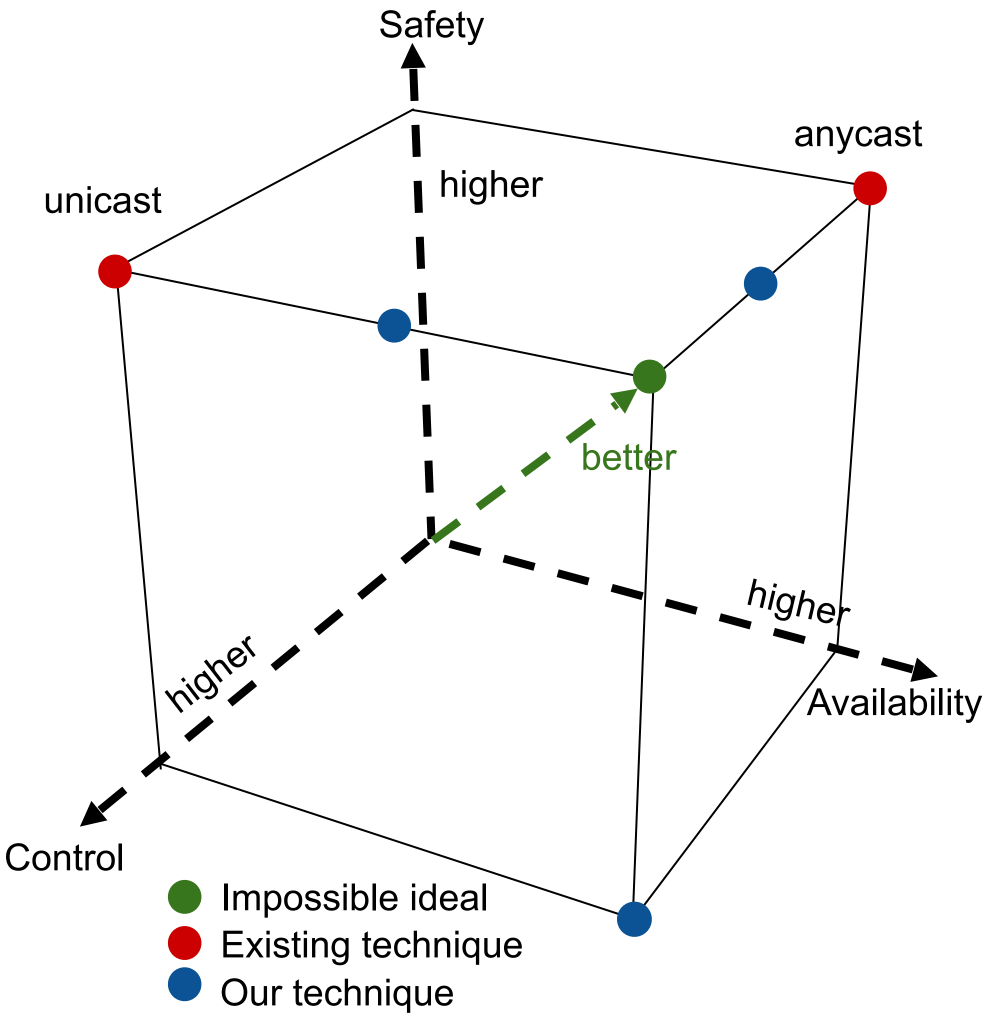
- Existing techniques compromise **control** or **availability**.
- We developed three new techniques.

Technique	Control	Availability	Safety
Unicast	High	Low	High
Anycast	Low	High	High
Reactive	High	High	Low



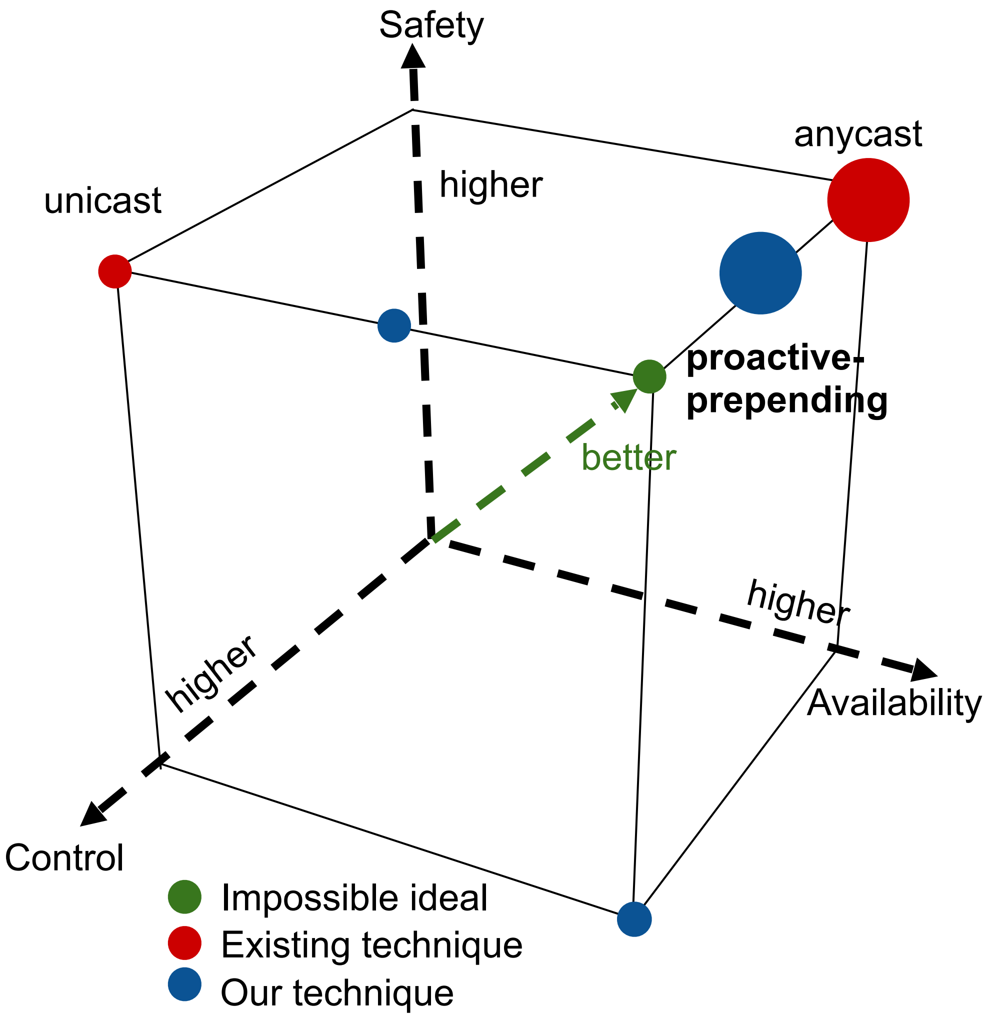
- Existing techniques compromise **control** or **availability**.
- We developed three new techniques.

	Technique	Control	Availability	Safety
	Unicast	High	Low	High
Proactive	*Superprefix*	High	Medium	High
Proactive	Anycast	Low	High	High
	Reactive	High	High	Low



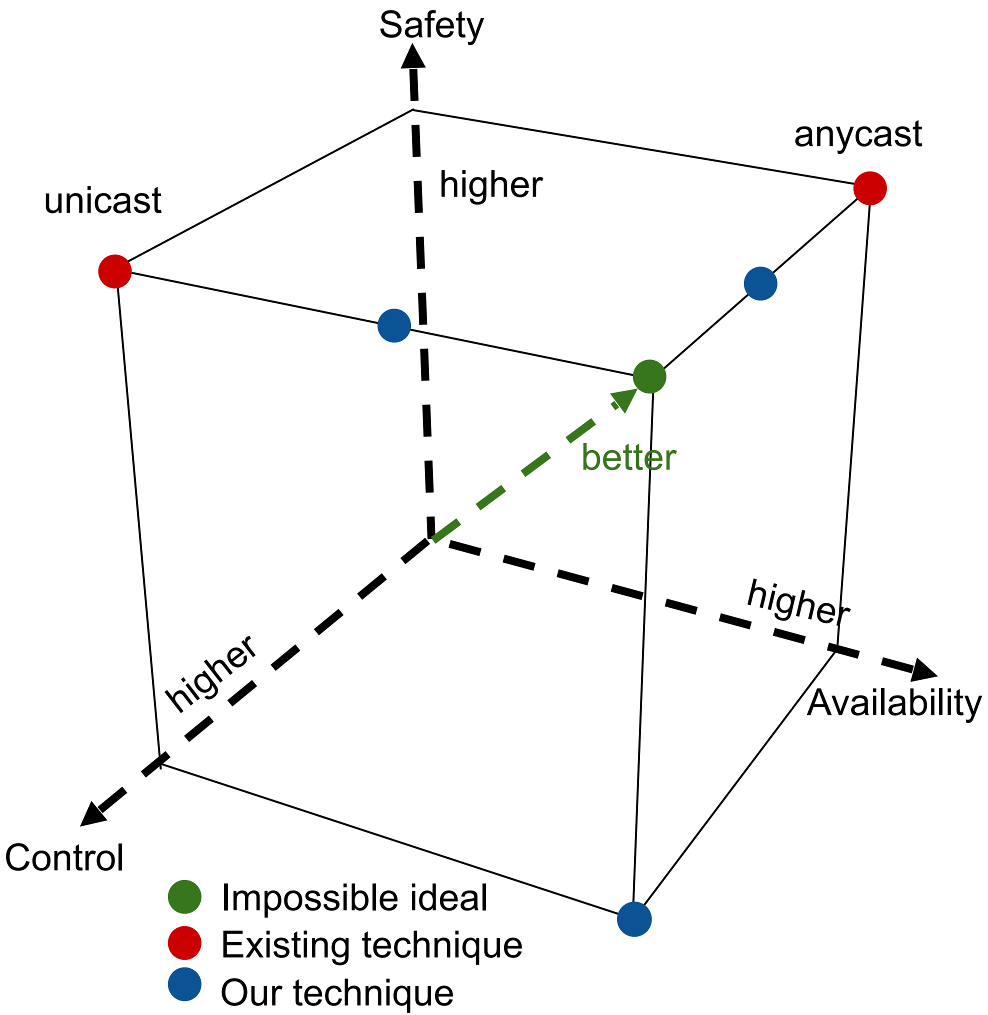
- Existing techniques compromise **control** or **availability**.
- We developed three new techniques.

	Technique	Control	Availability	Safety
	Unicast	High	Low	High
Proactive	*Superprefix*	High	Medium	High
Proactive	Anycast	Low	High	High
	Reactive	High	High	Low



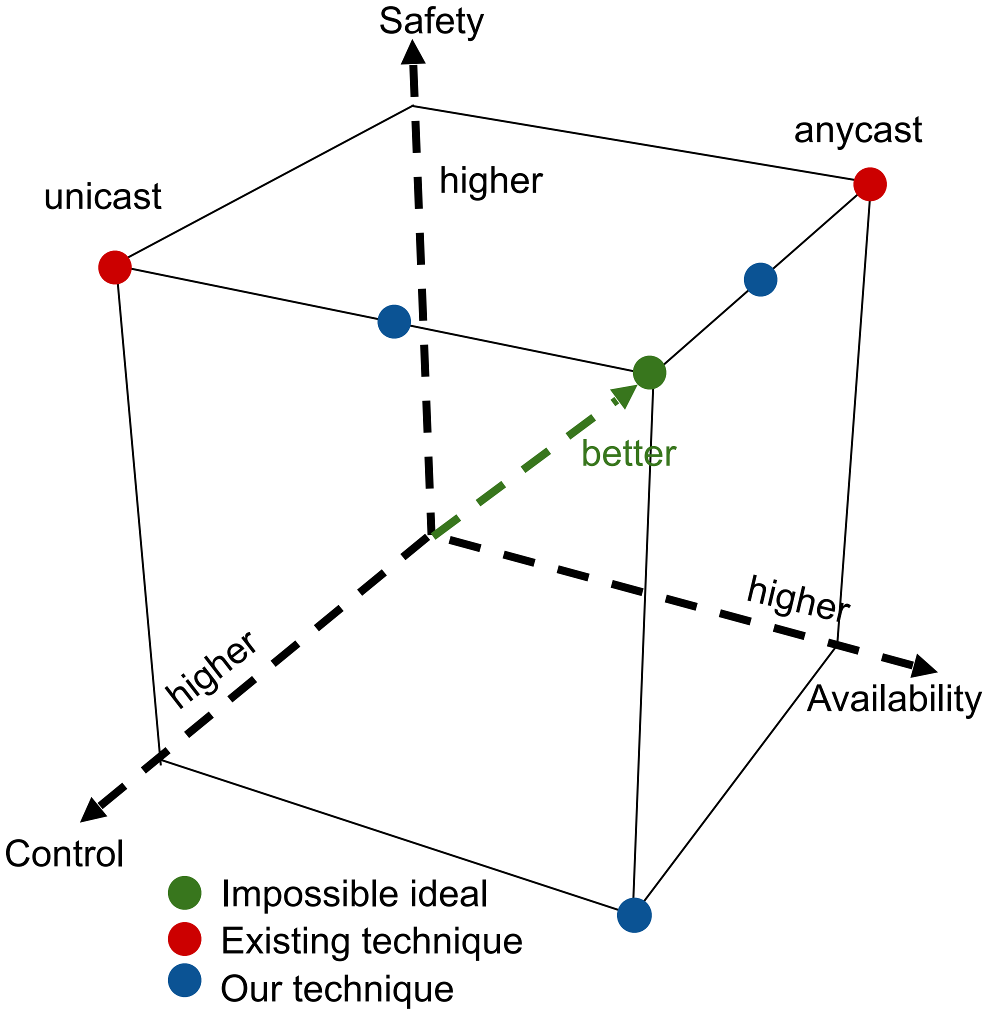
- Existing techniques compromise **control** or **availability**.
- We developed three new techniques.

	Technique	Control	Availability	Safety
	Unicast	High	Low	High
Proactive	*Superprefix*	High	Medium	High
Proactive	*Depref*	Medium	High	High
Proactive	Anycast	Low	High	High
	Reactive	High	High	Low



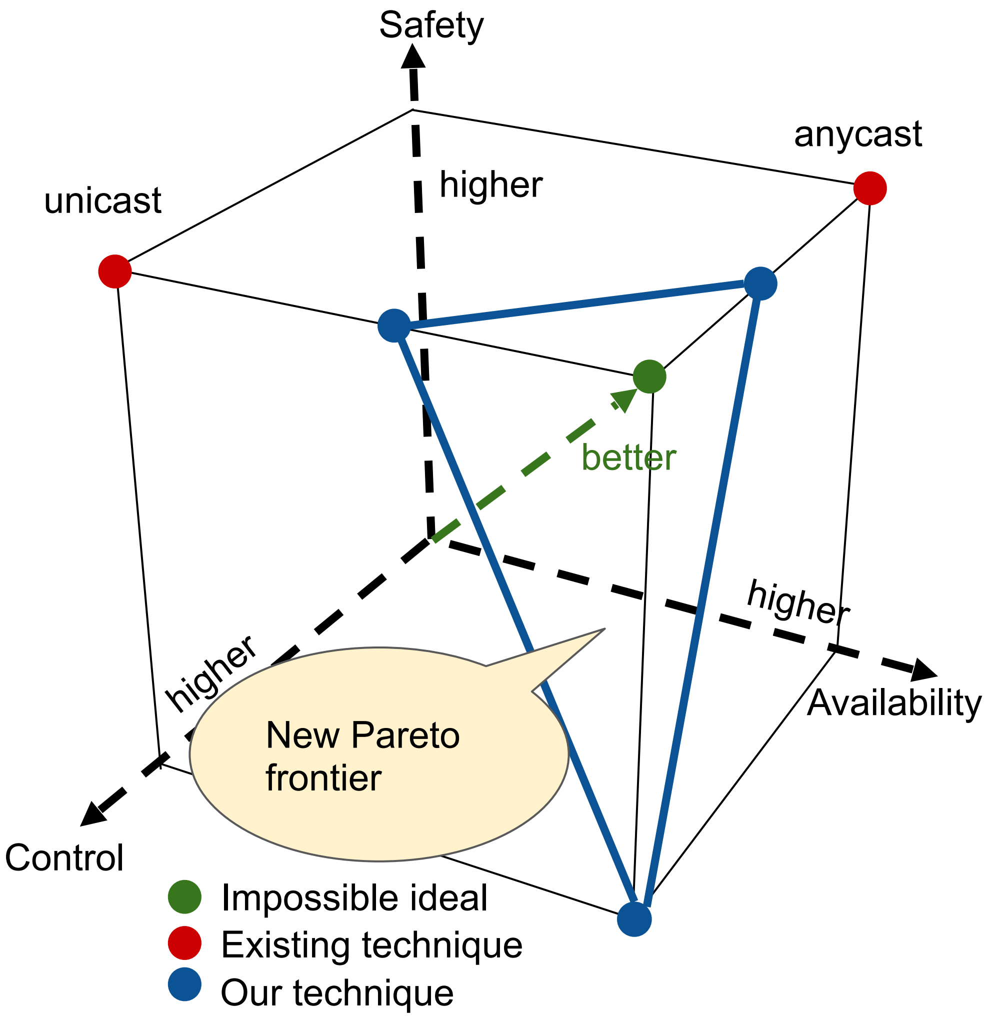
- Existing techniques compromise **control** or **availability**.
- We developed three new techniques.

	Technique	Control	Availability	Safety
	Unicast	High	Low	High
Proactive	*Superprefix*	High	Medium	High
Proactive	*Depref*	Medium	High	High
Proactive	Anycast	Low	High	High
	Reactive	High	High	Low



- Existing techniques compromise **control** or **availability**.
- We developed three new techniques.
- For each pair of goals, a new technique optimizes them while achieving better trade-offs than existing techniques.

	Technique	Control	Availability	Safety
	Unicast	High	Low	High
Proactive	*Superprefix*	High	Medium	High
Proactive	*Depref*	Medium	High	High
Proactive	Anycast	Low	High	High
	Reactive	High	High	Low



- Existing techniques compromise **control** or **availability**.
- We developed three new techniques.
- For each pair of goals, a new technique optimizes them while achieving better trade-offs than existing techniques.

	Technique	Control	Availability	Safety
	Unicast	High	Low	High
Proactive	*Superprefix*	High	Medium	High
Proactive	*Depref*	Medium	High	High
Proactive	Anycast	Low	High	High
	Reactive	High	High	Low

New approaches for cloud/CDN ingress routing enable new tradeoffs

- Existing techniques compromise control or availability
- Announcing failed site's prefix from other sites upon failure (***reactive*** anycast) runs risk of turning a local failure into a widespread one, compromising ***safety***
- Tradeoffs are fundamental:
  any technique relying on DNS + BGP for content redirection
  must compromise at least one of *control, availability, or safety*
- For each pair of goals, one of our new technique optimizes them while achieving better trade-offs than existing techniques.
  Initial techniques at IMC 2022 (Best Short Paper). Improvements under submission
- Or: Use special deployments to sidestep DNS + BGP
  to optimize all 3 goals, without being universal
  PAINTER, SIGCOMM 2023. SCULPTOR, under submission.