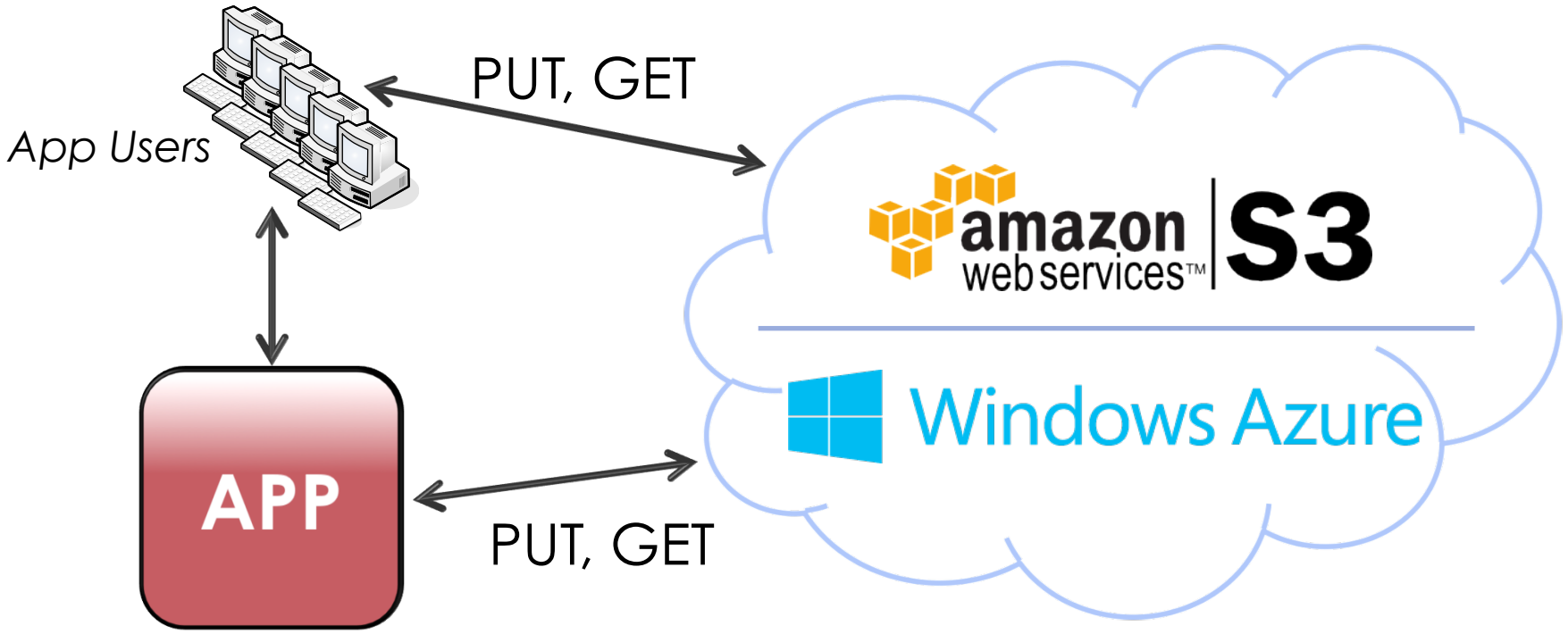
CosTLO: Cost-Effective Redundancy for Lower Latency Variance on Cloud Storage Services

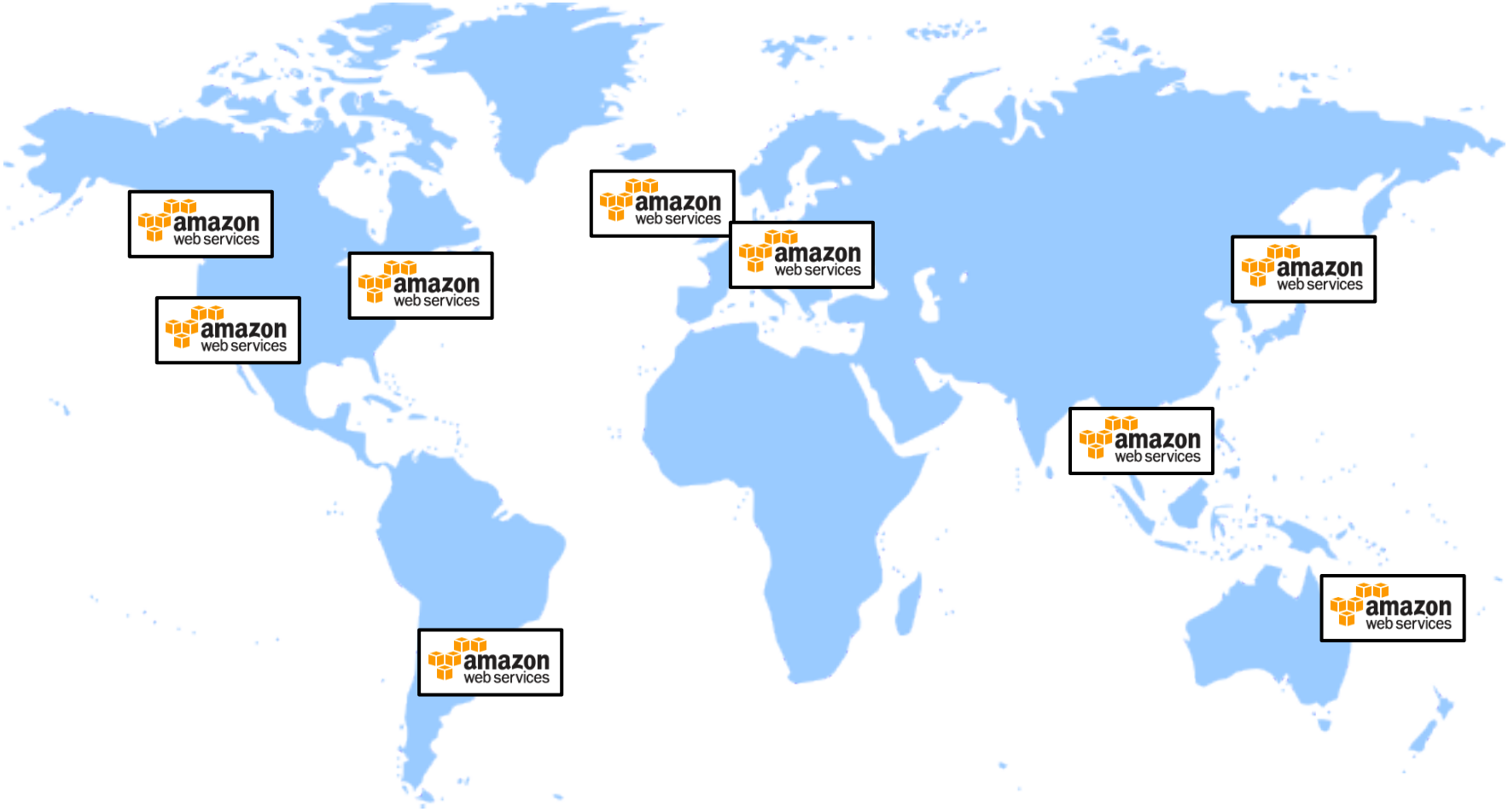
Zhe Wu, Curtis Yu, and Harsha V. Madhyastha

UC Riverside and University of Michigan

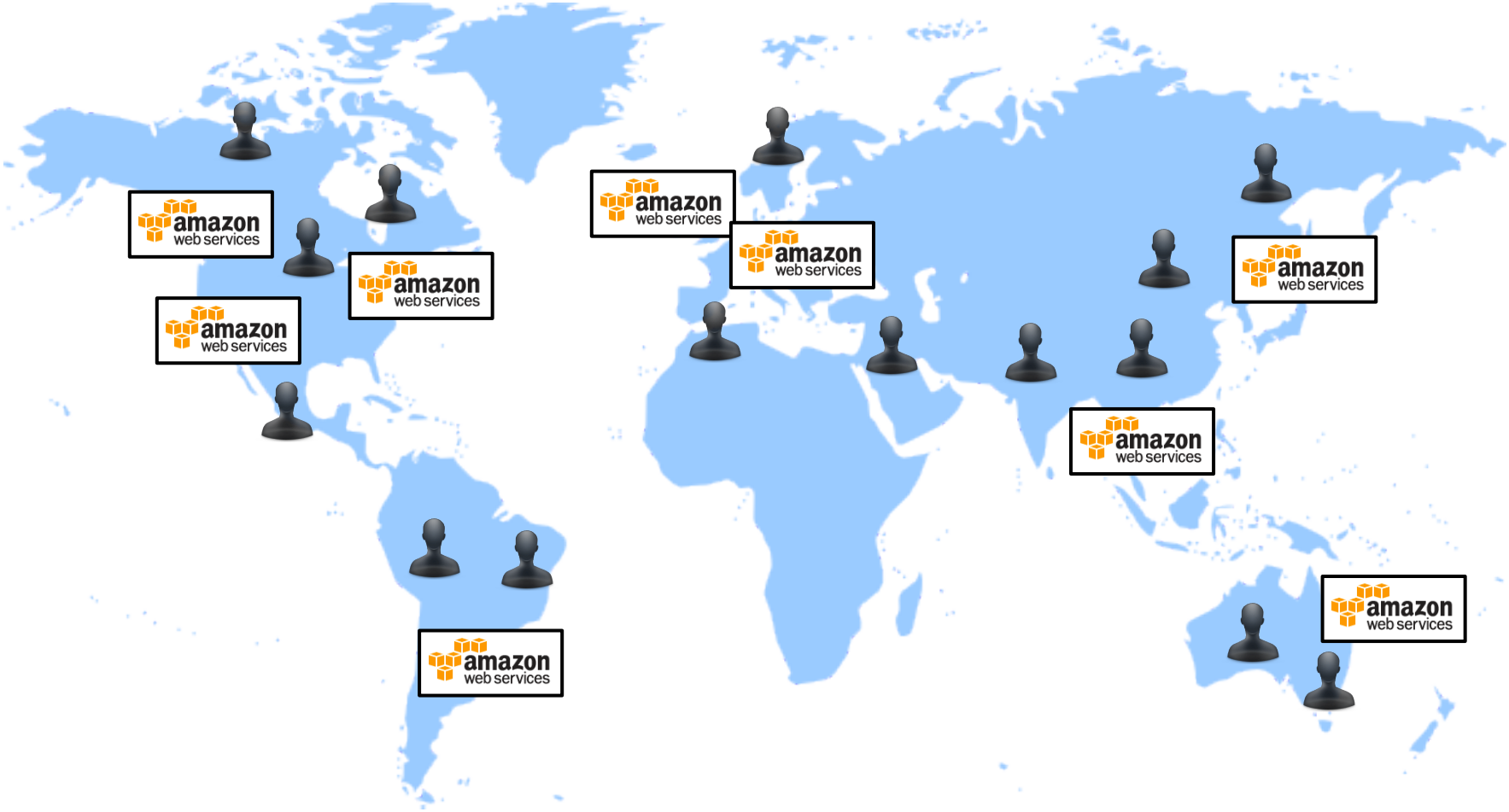
Cloud Storage Services



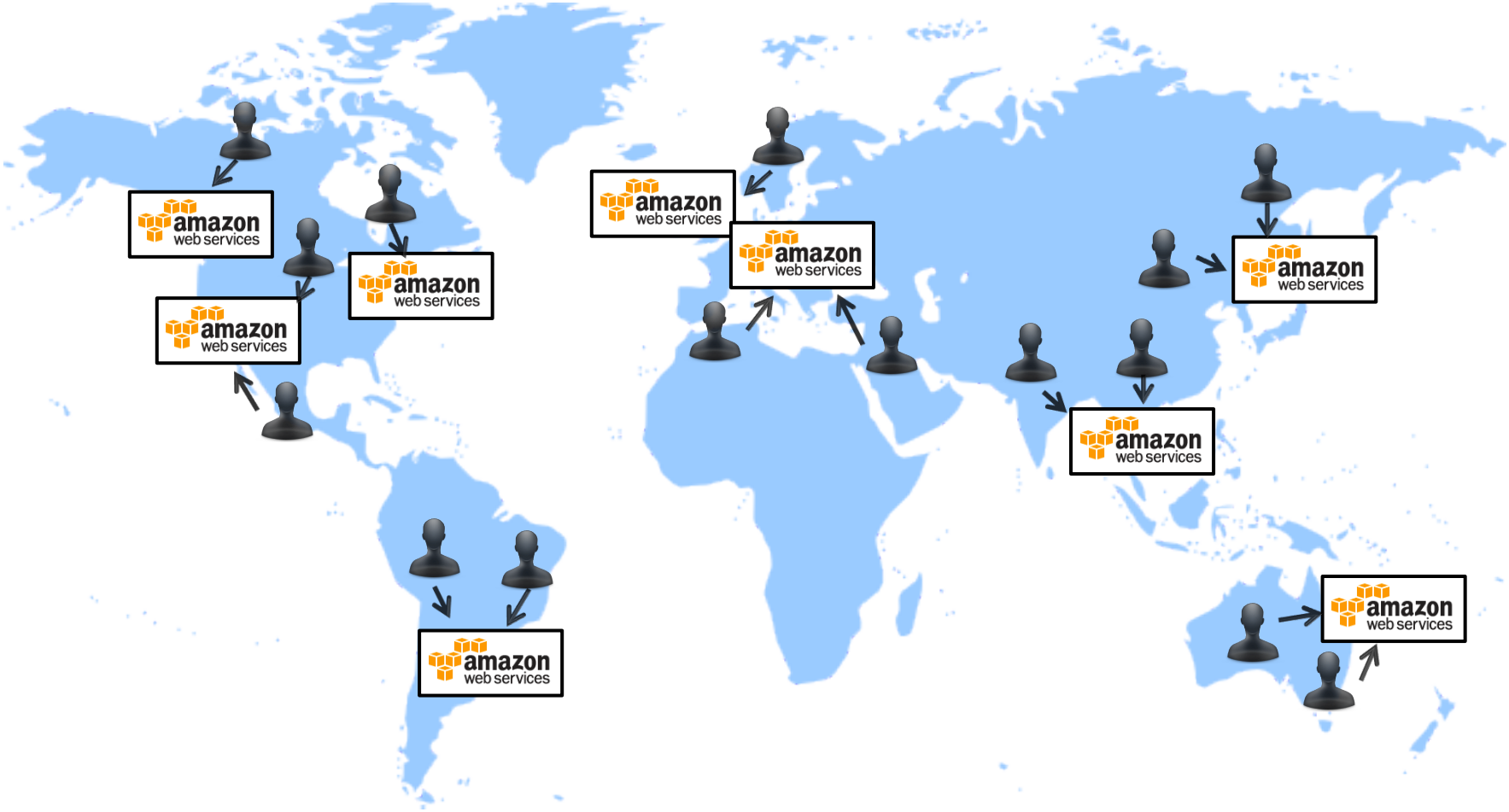
Geo-Distributed Cloud Services Enable Low Latency



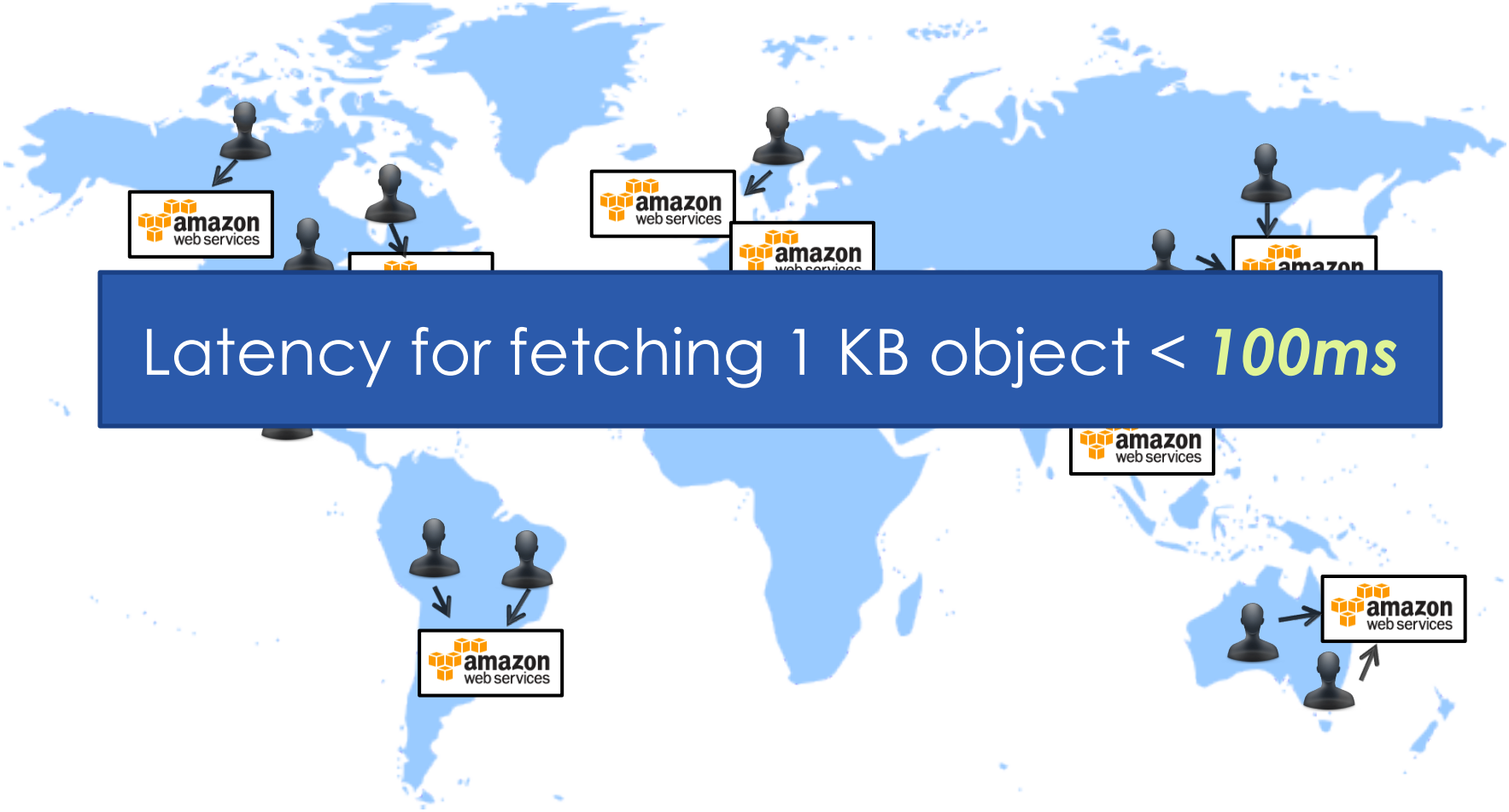
Geo-Distributed Cloud Services Enable Low Latency



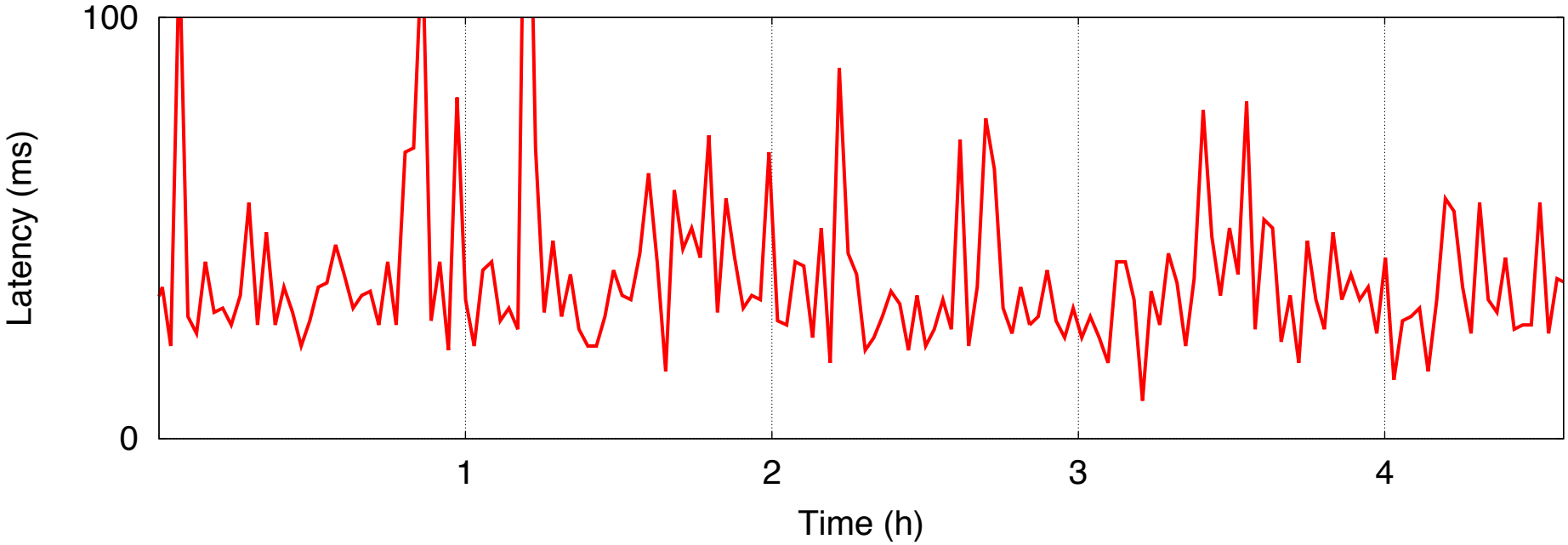
Geo-Distributed Cloud Services Enable Low Latency



Geo-Distributed Cloud Services Enable Low Latency



Problem: High Latency Variance



Quantifying Latency Variance

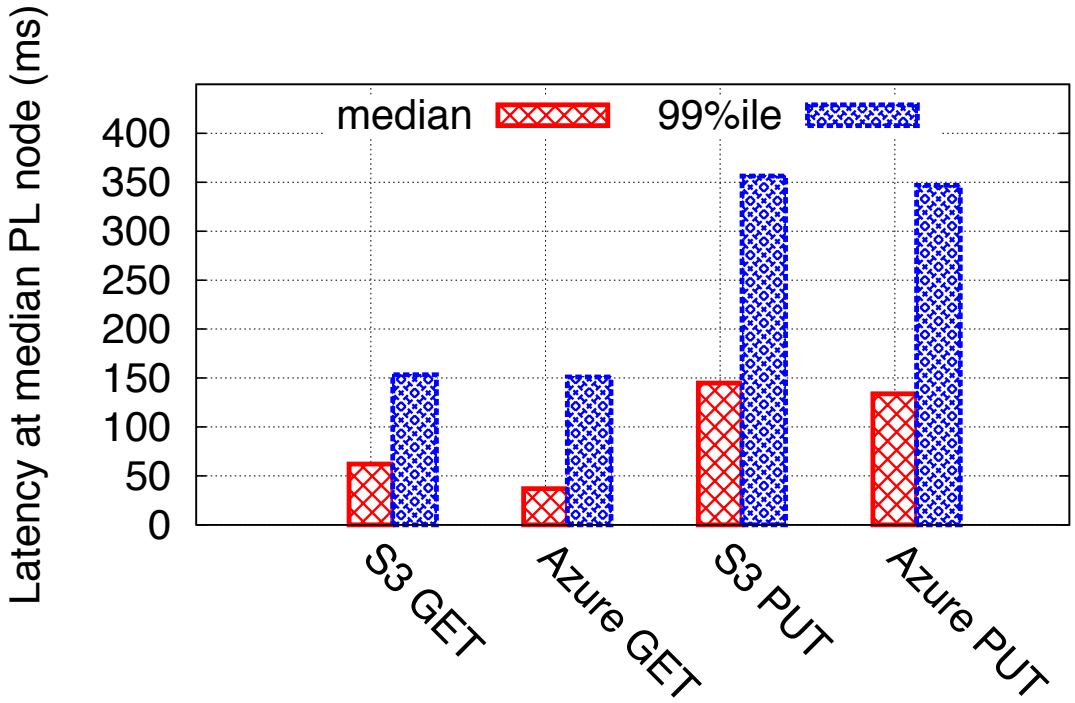
Measurements of S3 and Azure for one week

- 120 PlanetLab sites as clients
- Upload and download 1KB objects

Quantifying Latency Variance

Measurements of S3 and Azure for one week

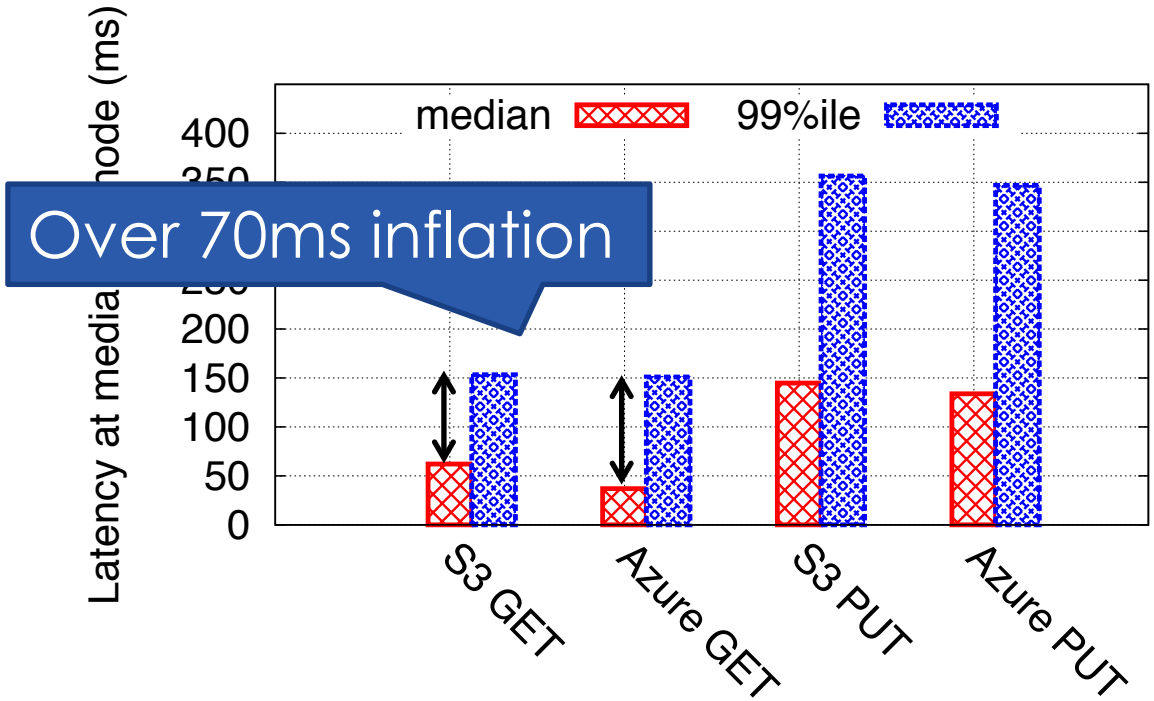
- 120 PlanetLab sites as clients
- Upload and download 1KB objects



Quantifying Latency Variance

Measurements of S3 and Azure for one week

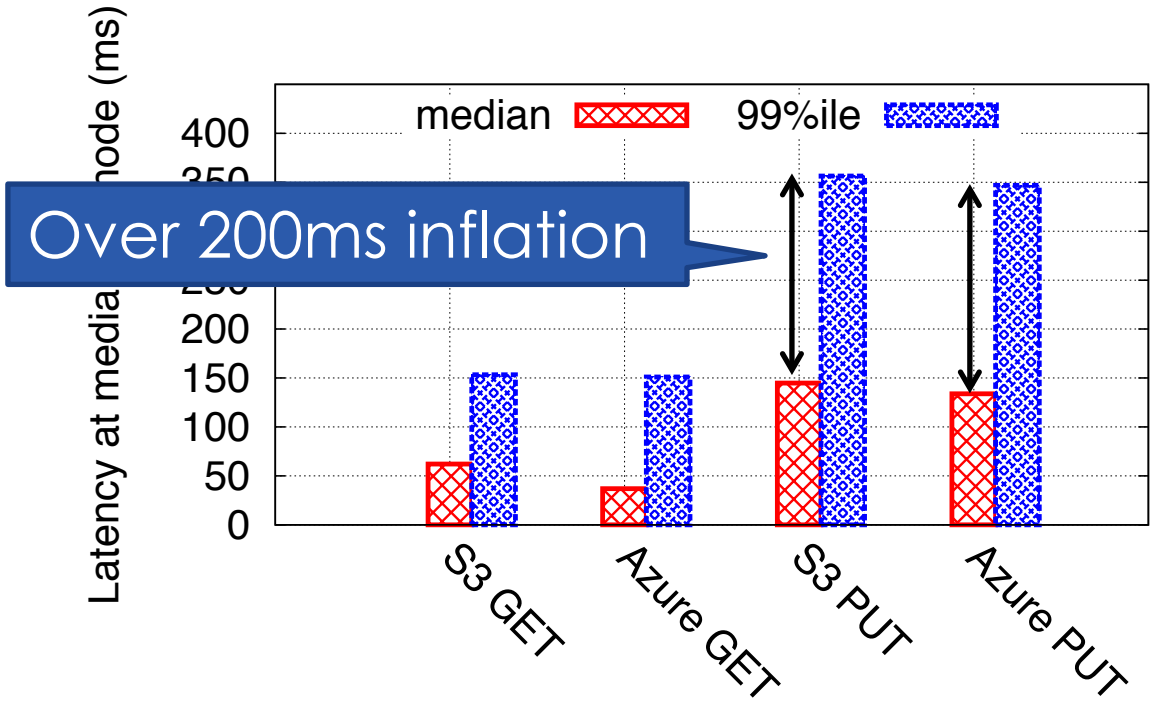
- 120 PlanetLab sites as clients
- Upload and download 1KB objects



Quantifying Latency Variance

Measurements of S3 and Azure for one week

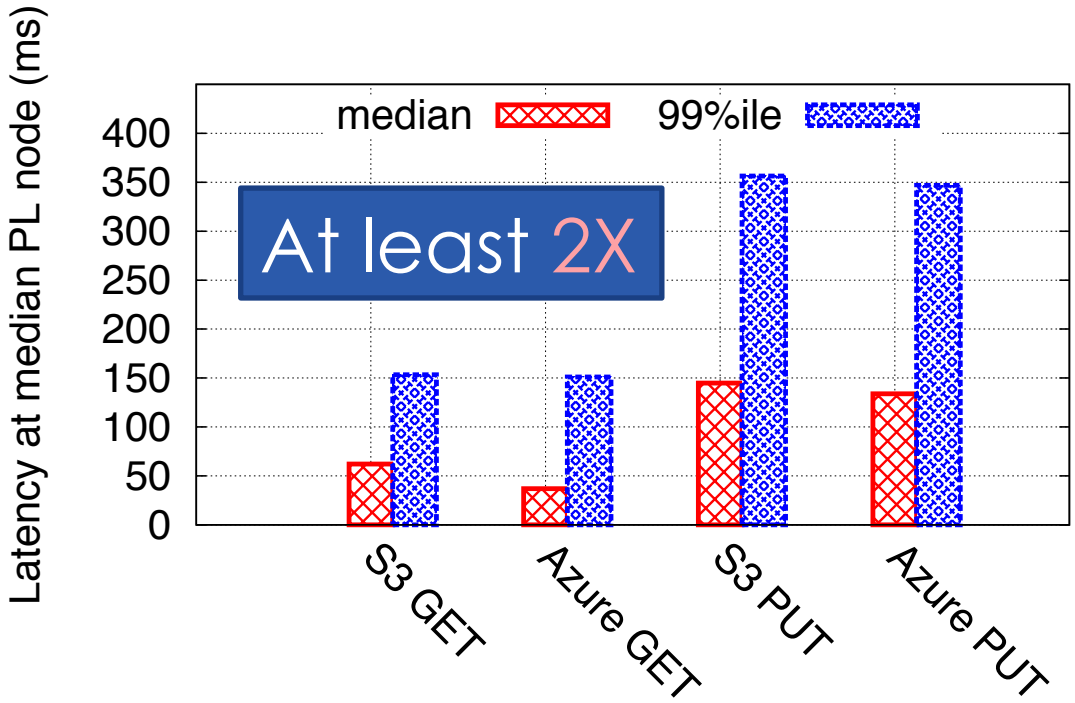
- 120 PlanetLab sites as clients
- Upload and download 1KB objects



Quantifying Latency Variance

Measurements of S3 and Azure for one week

- 120 PlanetLab sites as clients
- Upload and download 1KB objects

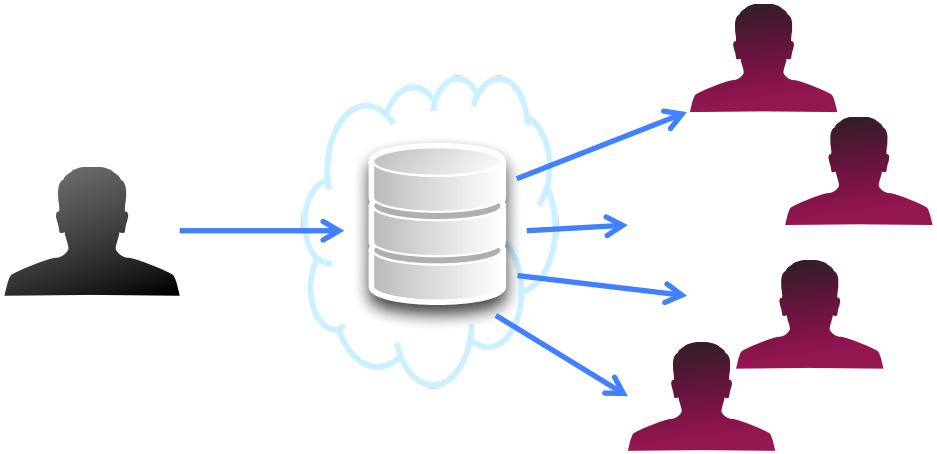


Impact of Latency Variance

Webpage loads

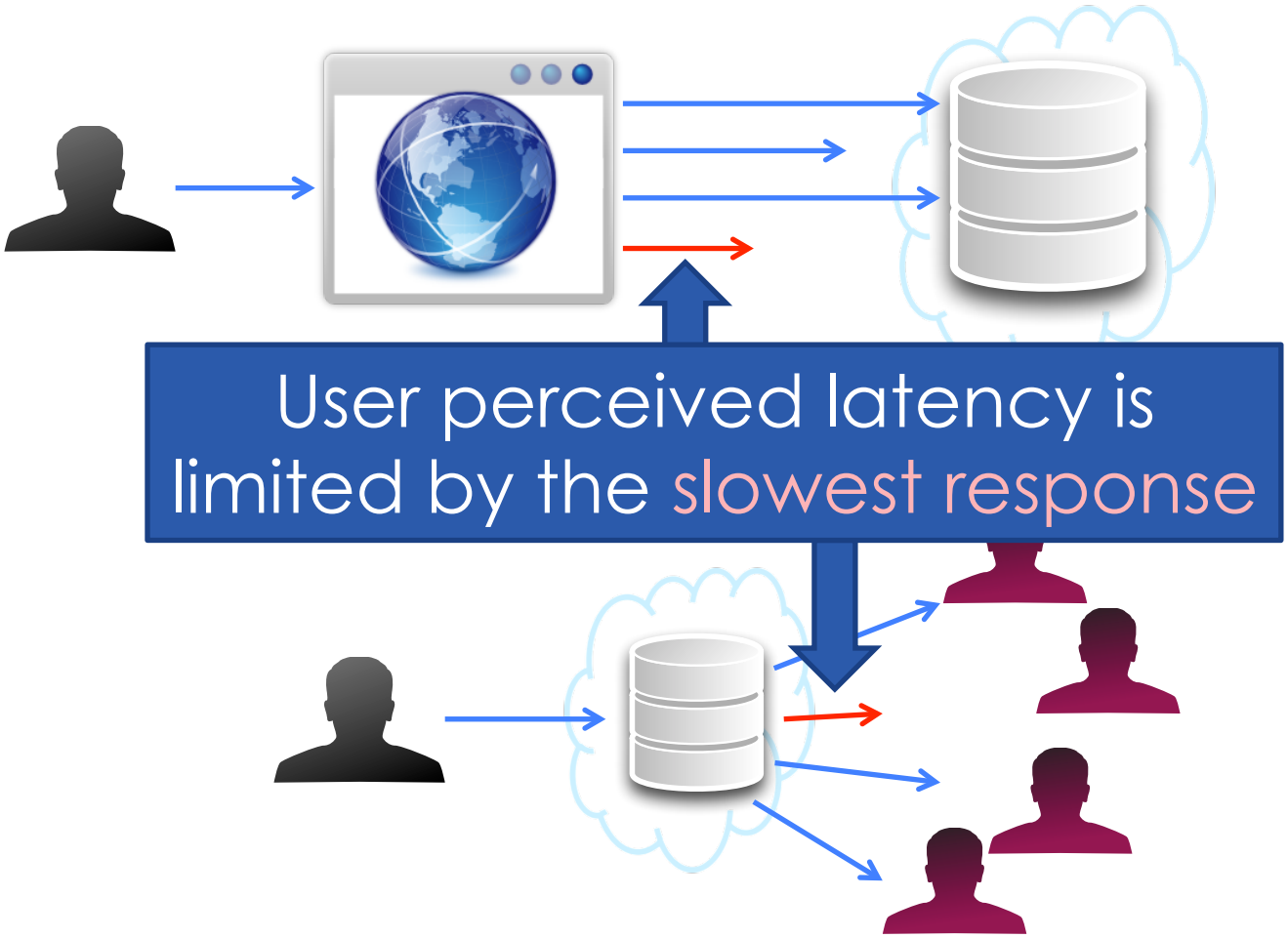
Impact of Latency Variance

Webpage loads



Social network synchronization

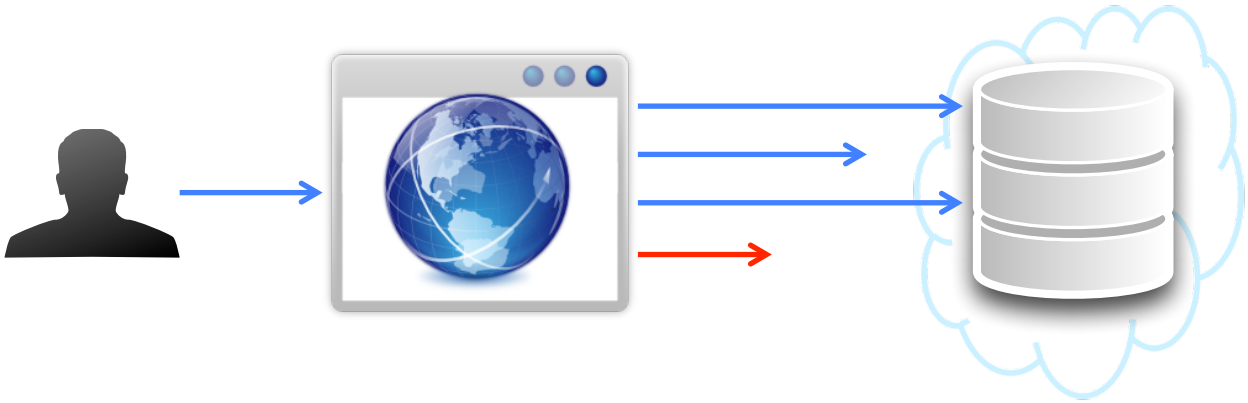
Impact of Latency Variance



Social network synchronization

Impact of Latency Variance

Impact of Latency Variance



- Measurements of PlanetLab sites downloading a webpage containing 50 objects
  - Measured median: 2X slower than ideal
  - Measured 99%ile: 20X slower than ideal

Impact of Latency Variance

Important to reduce single request tail latency to improve median application performance

- Measurements of PlanetLab sites downloading a webpage containing 50 objects
  - Measured median: 2X slower than ideal
  - Measured 99%ile: 20X slower than ideal

How to Combat Latency Variance?

- Lots of recent work
  - DeTail (SIGCOMM'12), Bobtail (NSDI'13), PriorityMeister (SoCC'14), C3 (NSDI'15)…
  - Require modification of underlying cloud system

How to Combat Latency Variance?

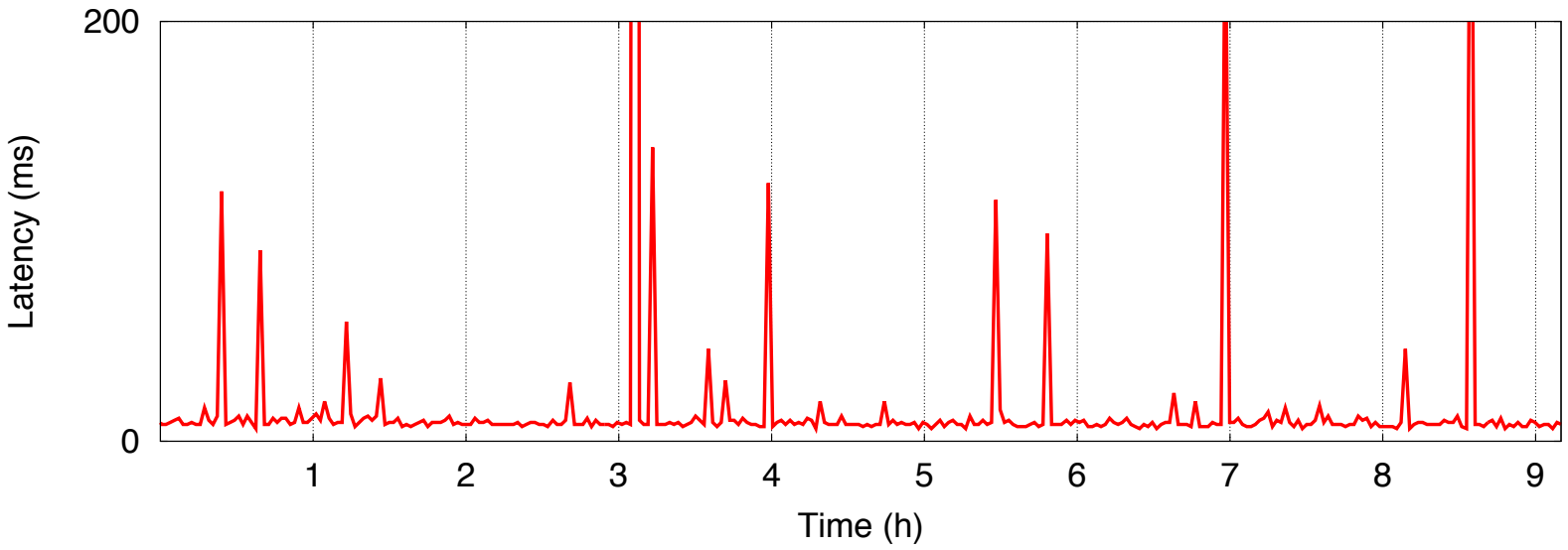
- Lots of recent work
  - DeTail (SIGCOMM'12), Bobtail (NSDI'13), PriorityMeister (SoCC'14), C3 (NSDI'15)…
  - Require modification of underlying cloud system

- Our consideration

What can application providers do to reduce latency variance?

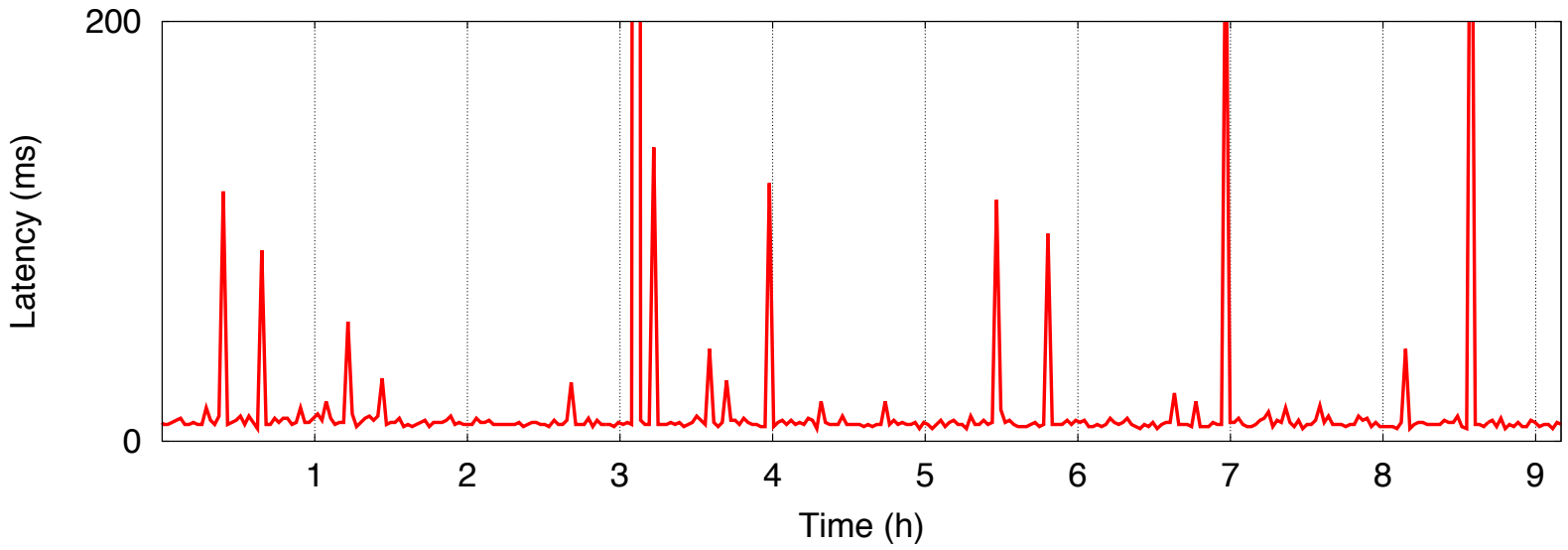
Approach: Redundancy

- Tail latencies dominated by isolated spikes



Approach: Redundancy

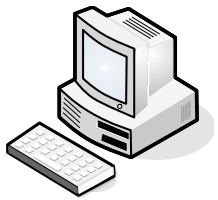
- Tail latencies dominated by isolated spikes



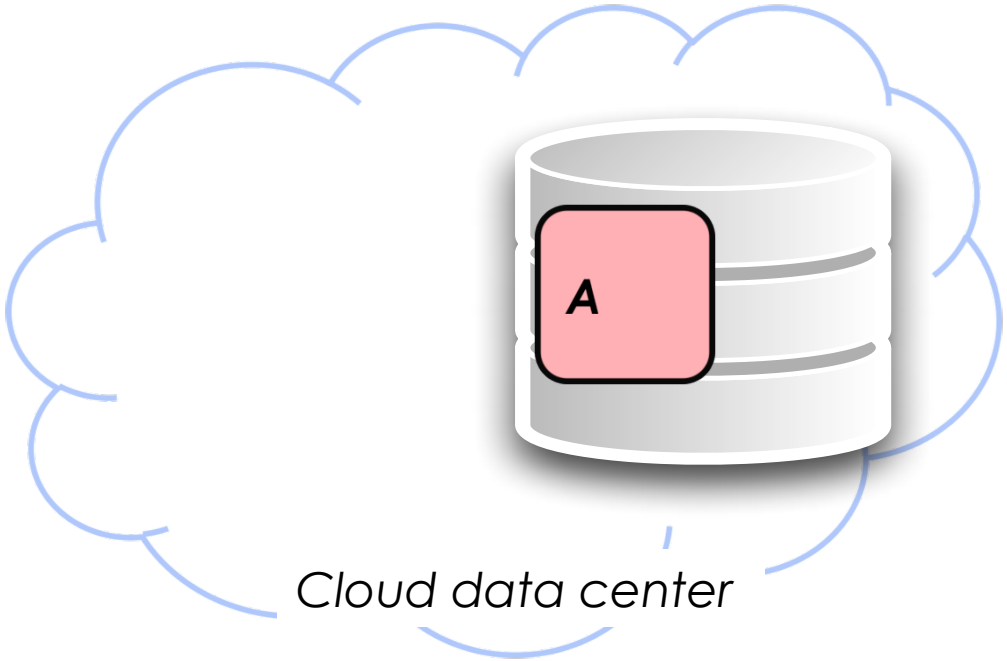
- Our approach: use redundant requests

How To Use Redundancy?

Simplest way: Redundant requests to same object

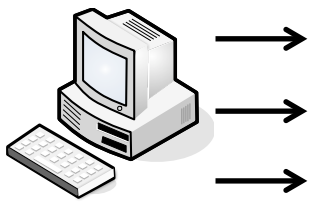


Client

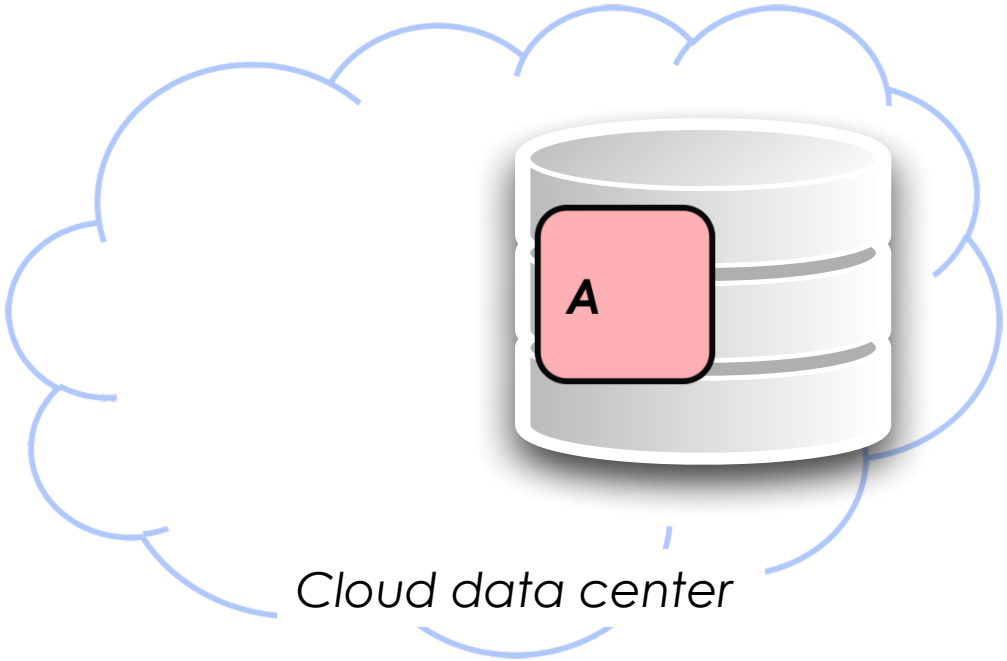


How To Use Redundancy?

Simplest way: Redundant requests to same object



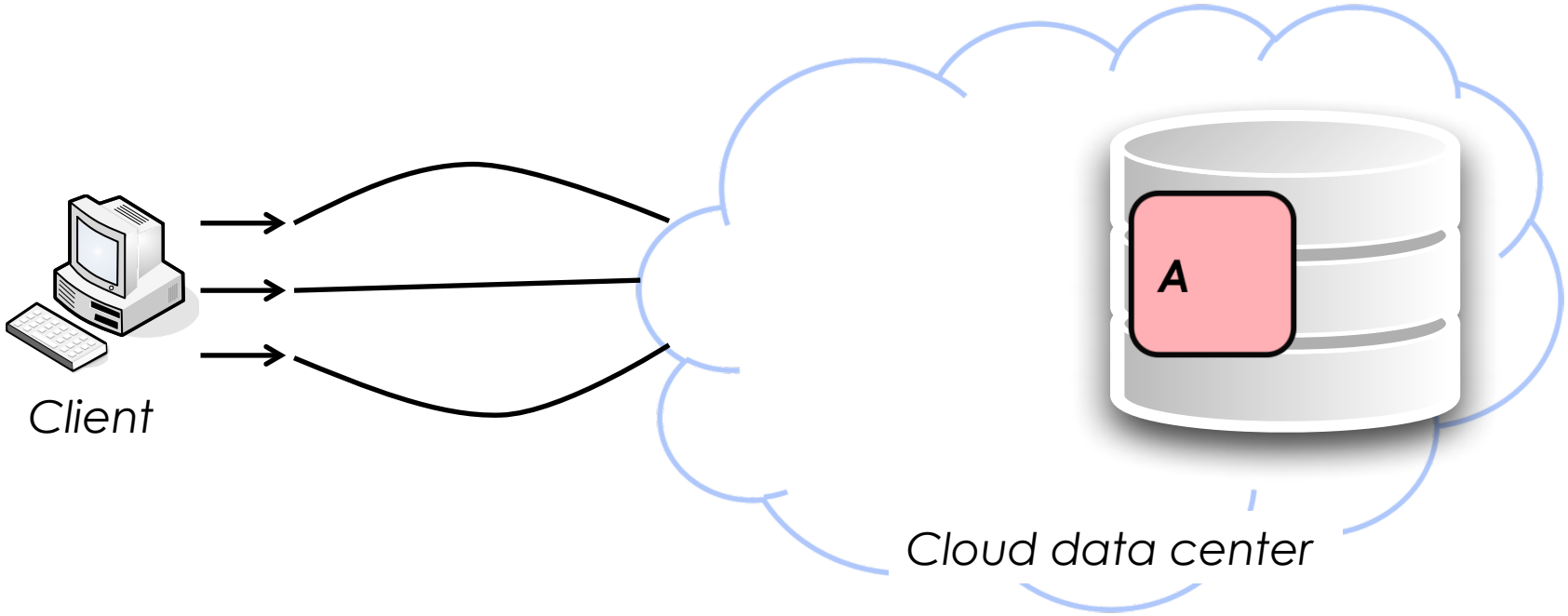
Client



Cloud data center

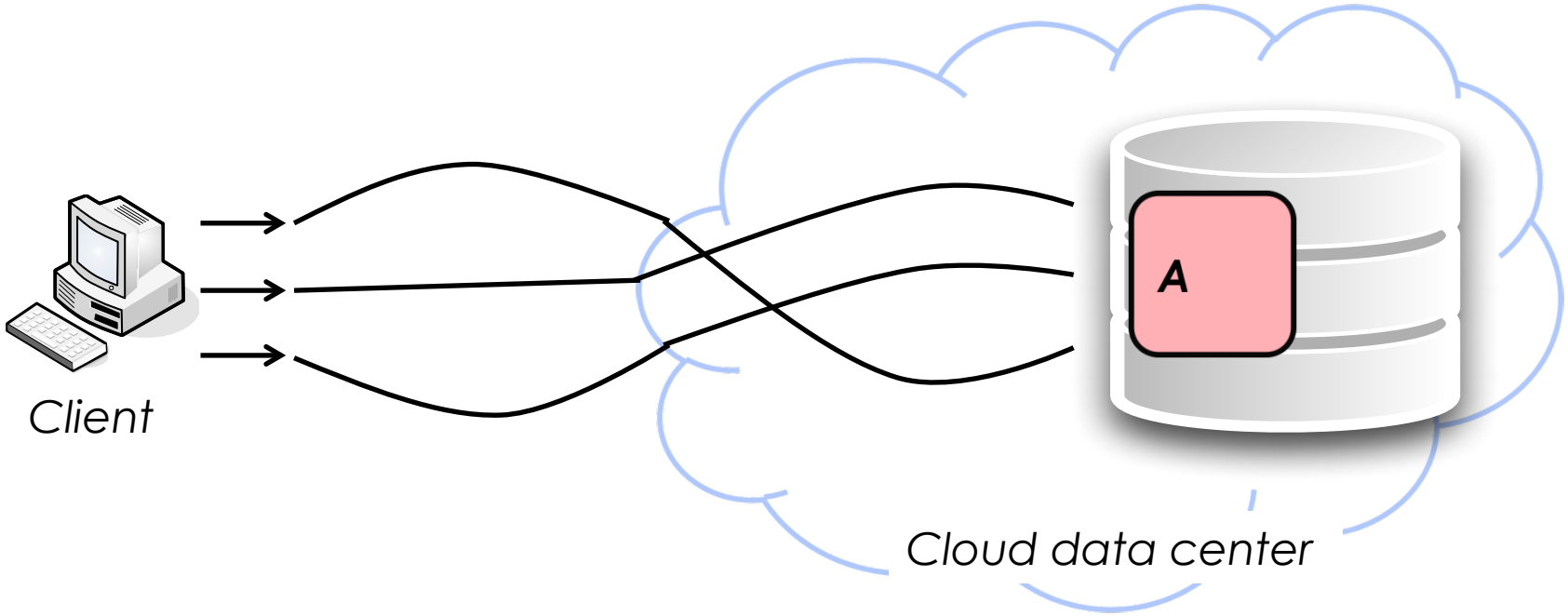
How To Use Redundancy?

Simplest way: Redundant requests to same object



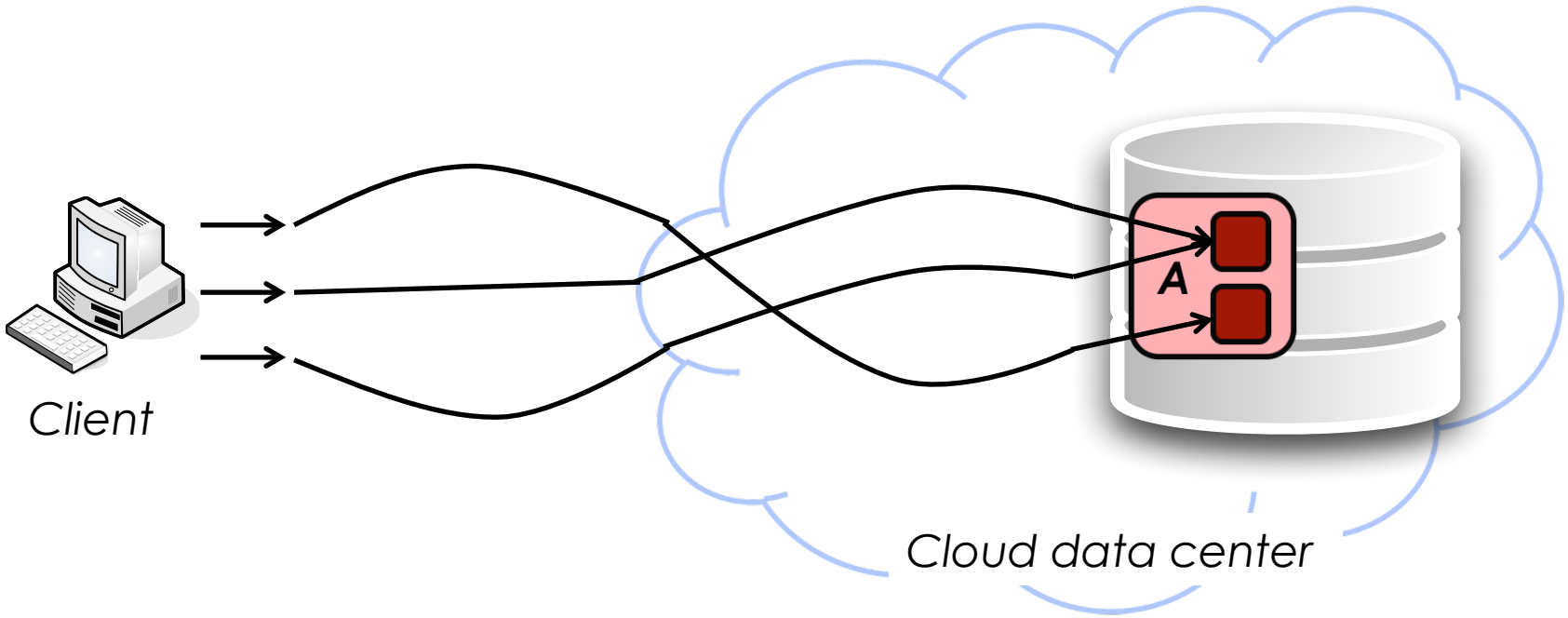
How To Use Redundancy?

Simplest way: Redundant requests to same object



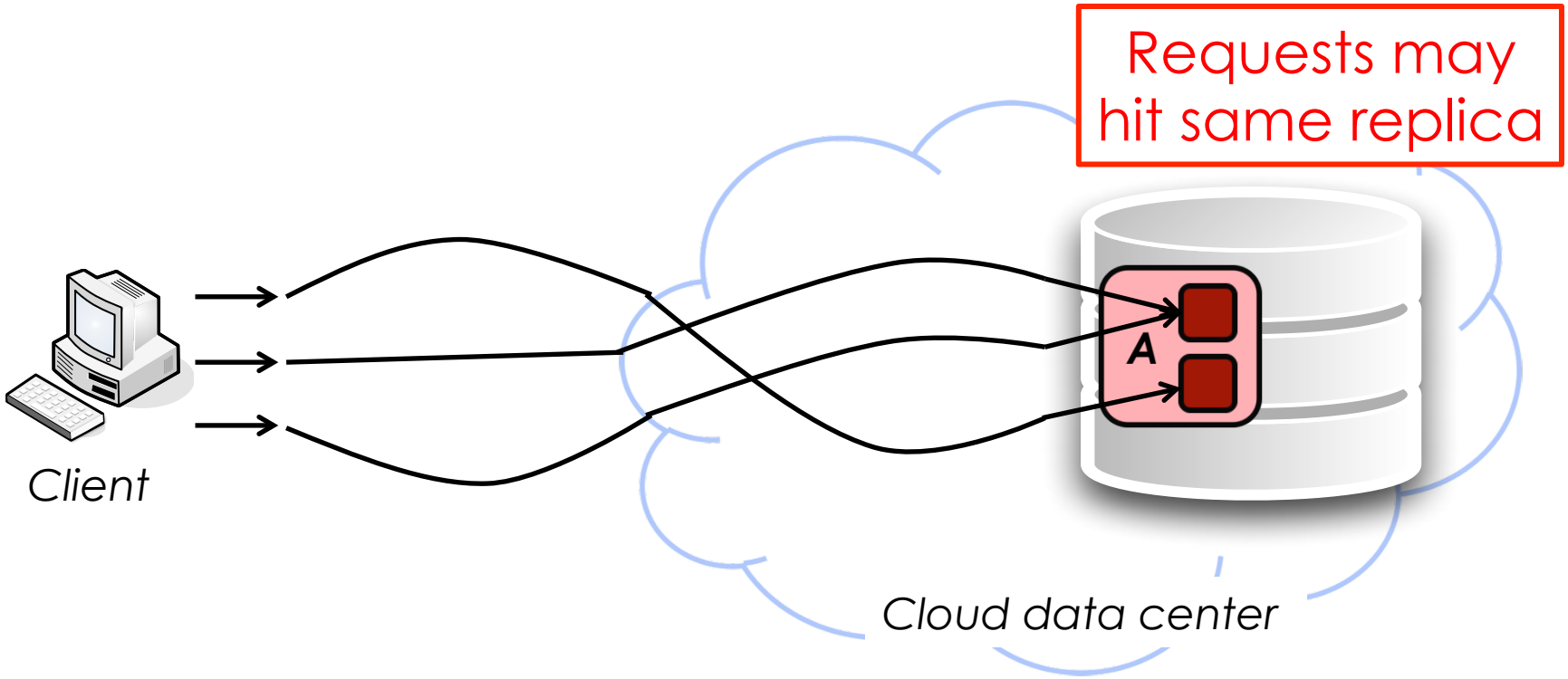
How To Use Redundancy?

Simplest way: Redundant requests to same object



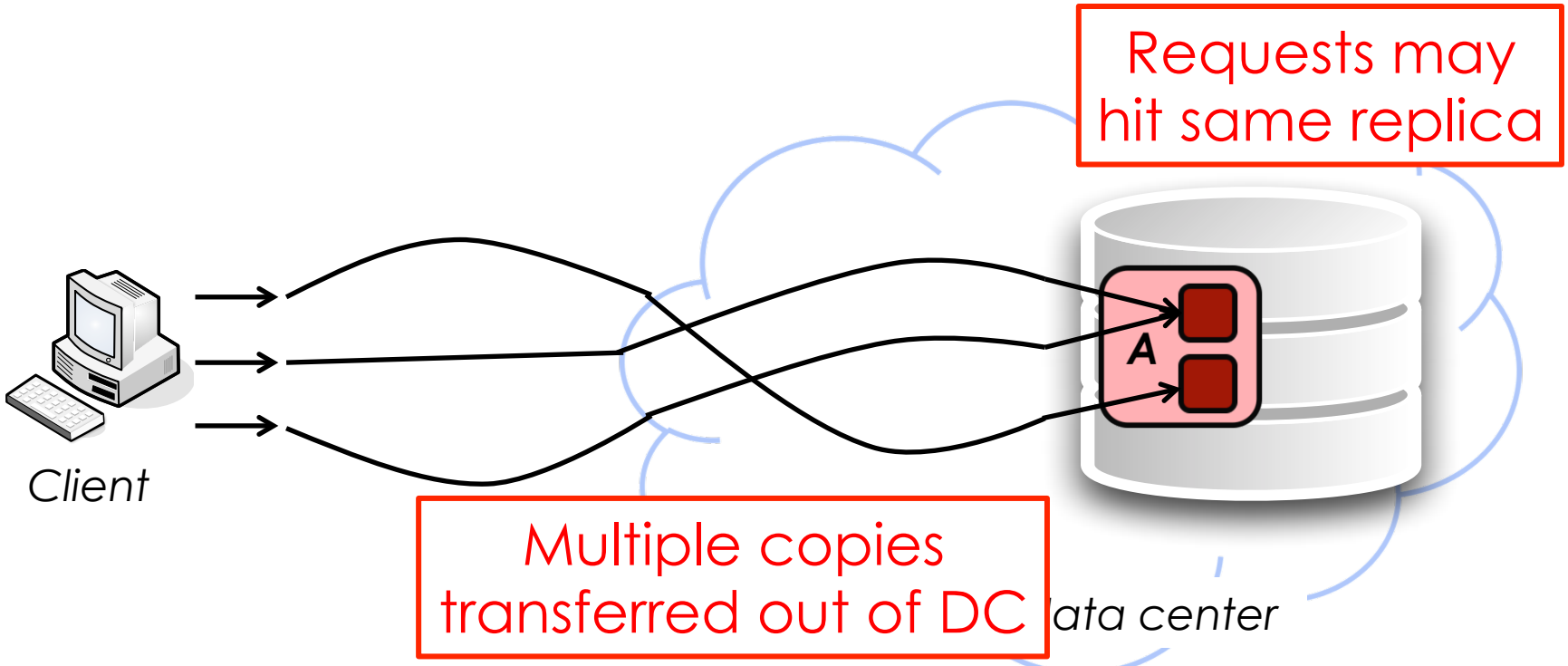
How To Use Redundancy?

Simplest way: Redundant requests to same object



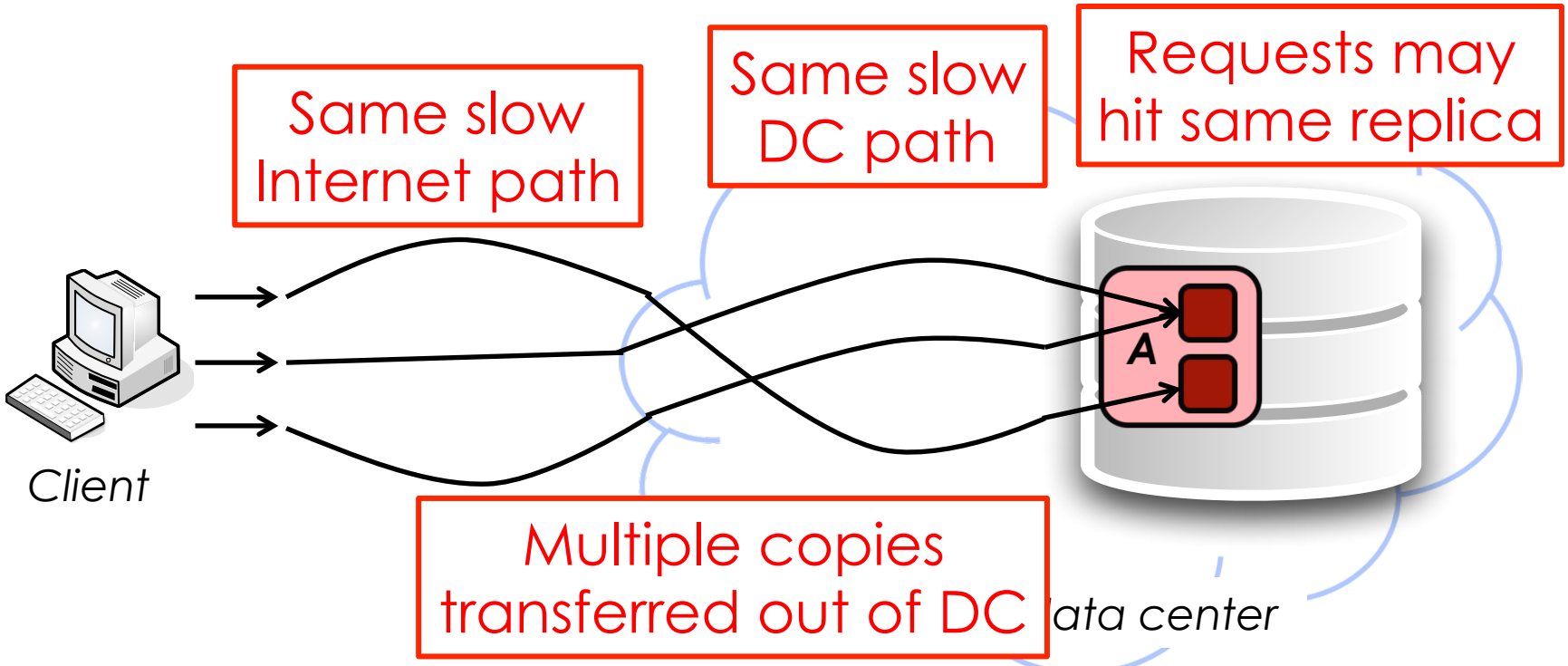
How To Use Redundancy?

Simplest way: Redundant requests to same object



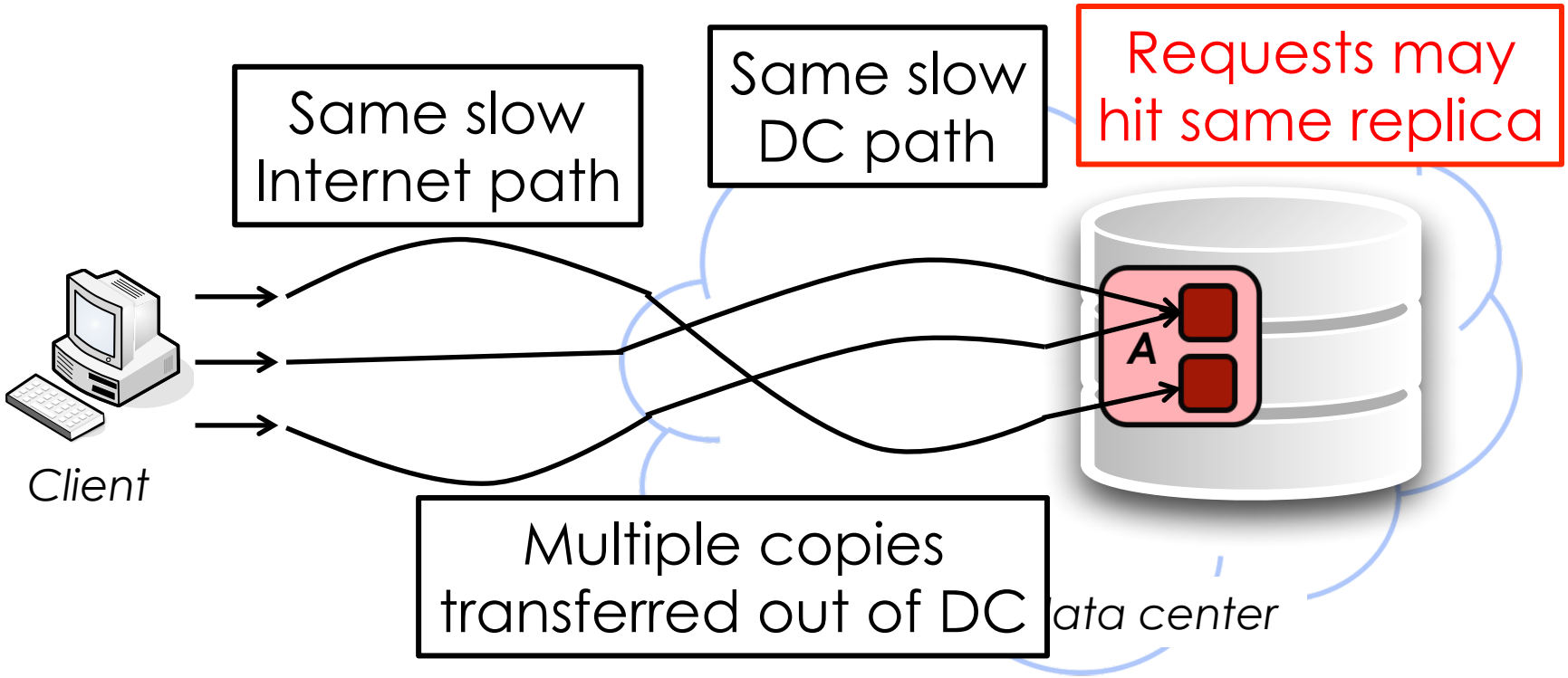
How To Use Redundancy?

Simplest way: Redundant requests to same object

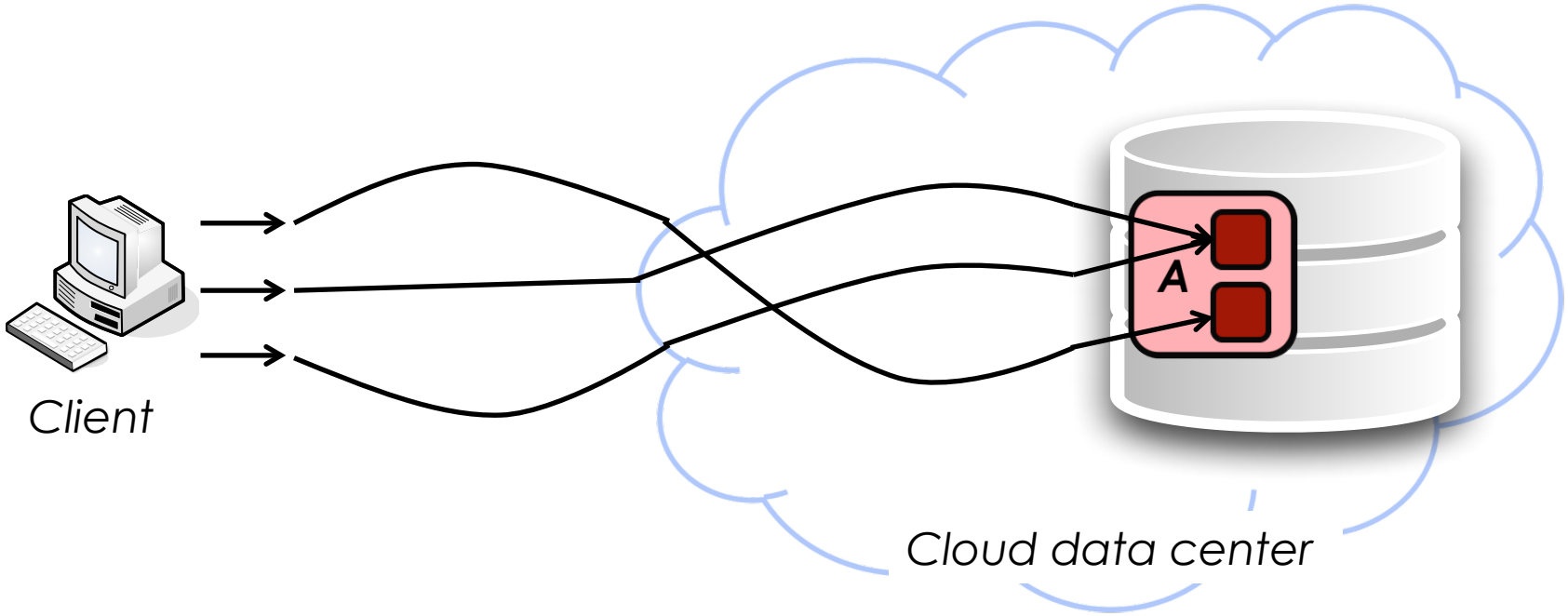


How To Use Redundancy?

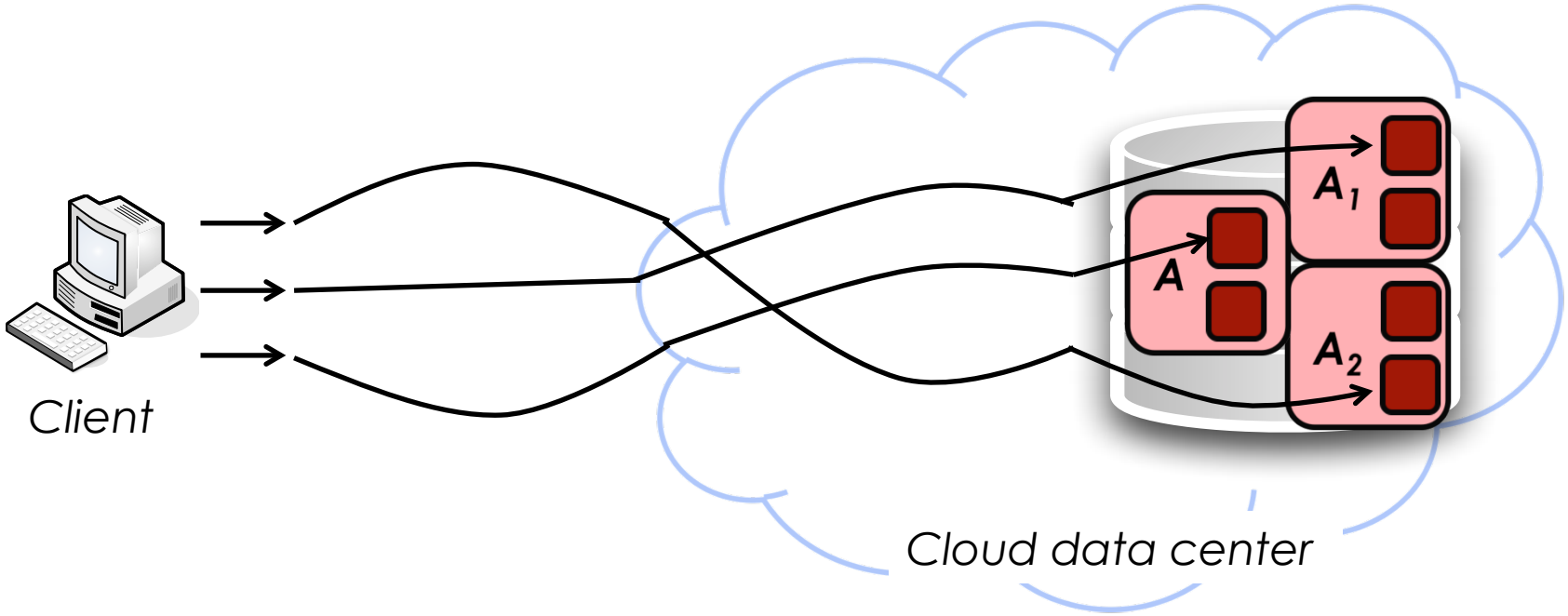
Simplest way: Redundant requests to same object



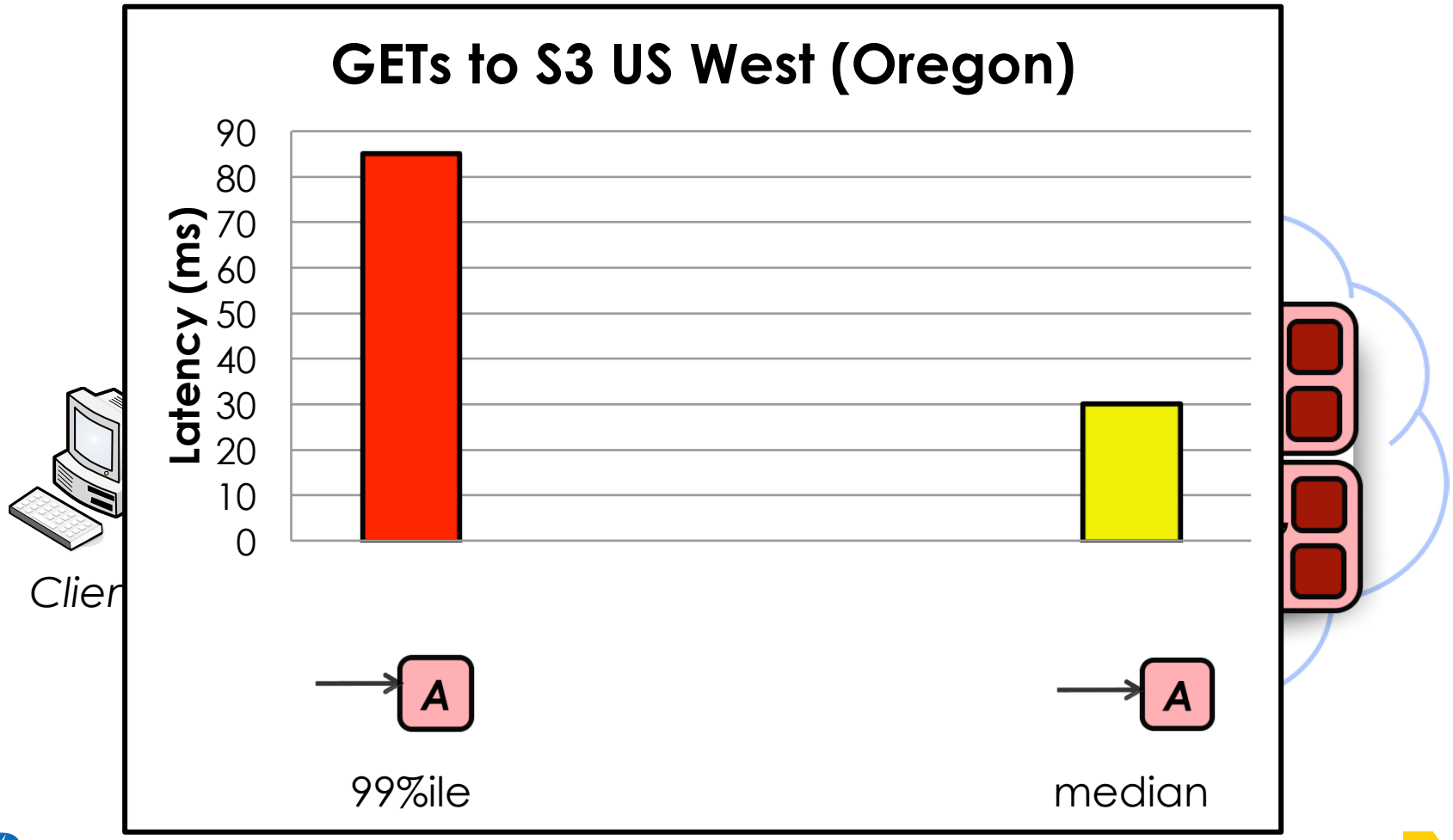
Use Multiple Copies of Object



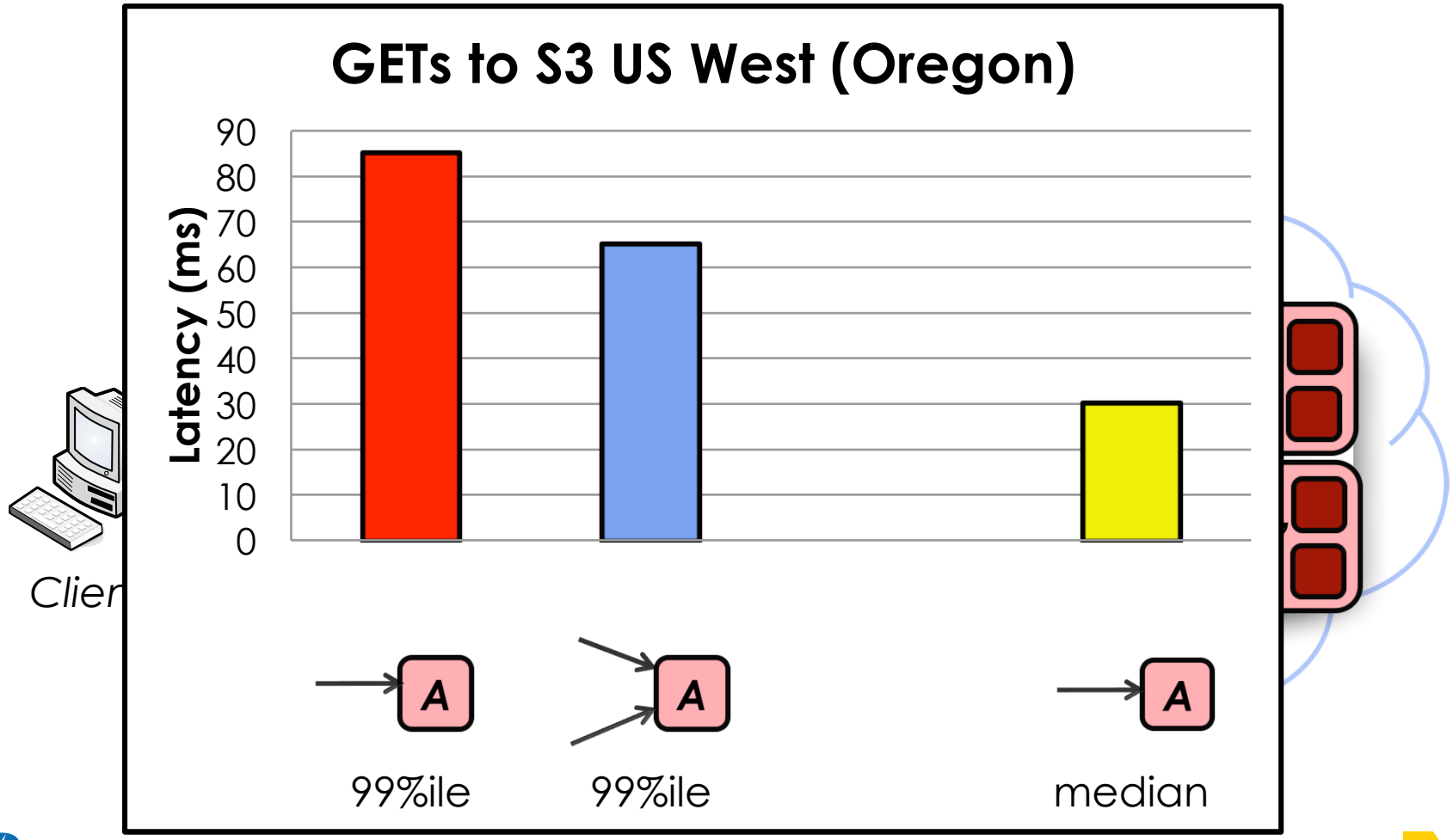
Use Multiple Copies of Object



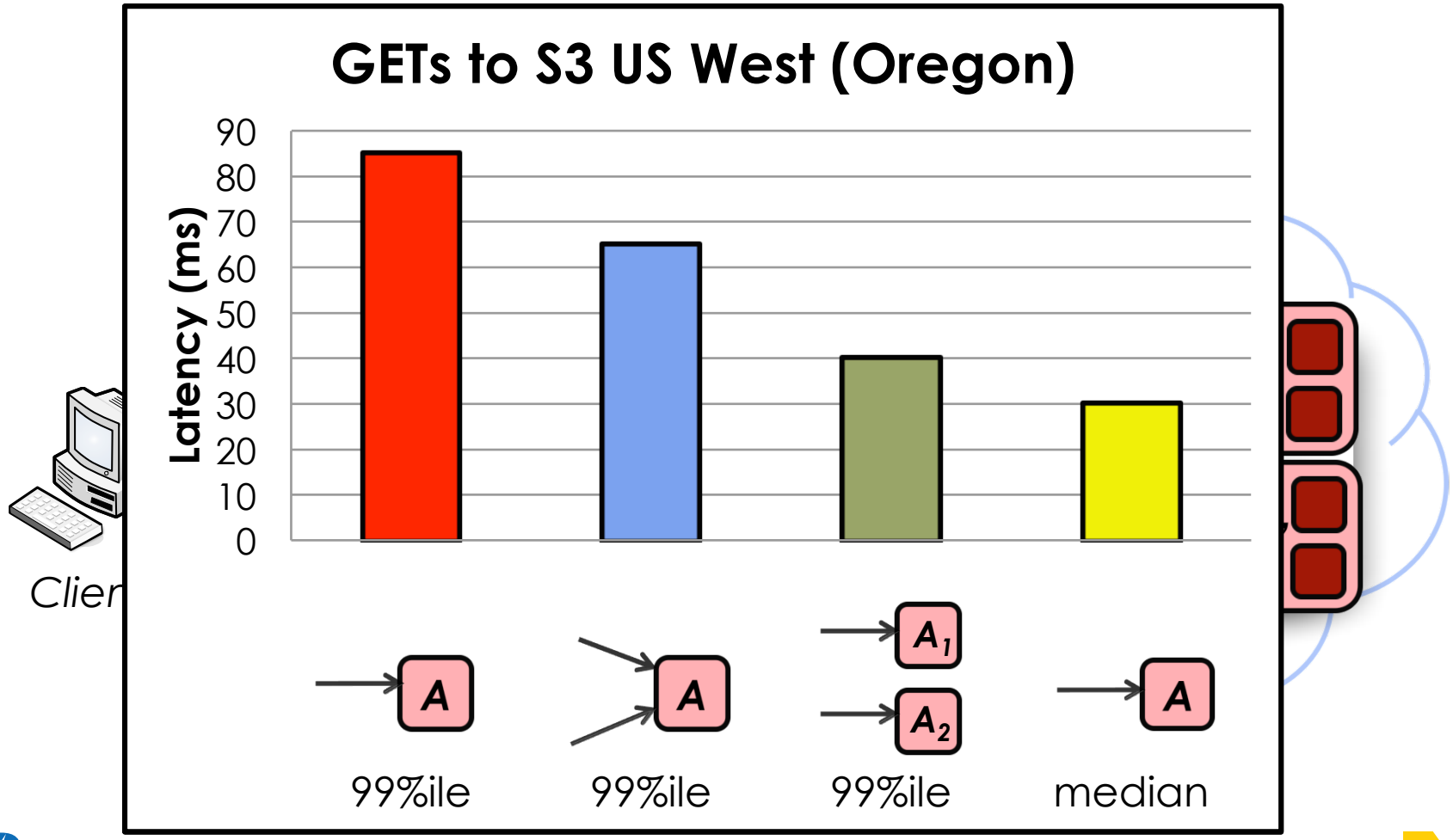
Use Multiple Copies of Object



Use Multiple Copies of Object

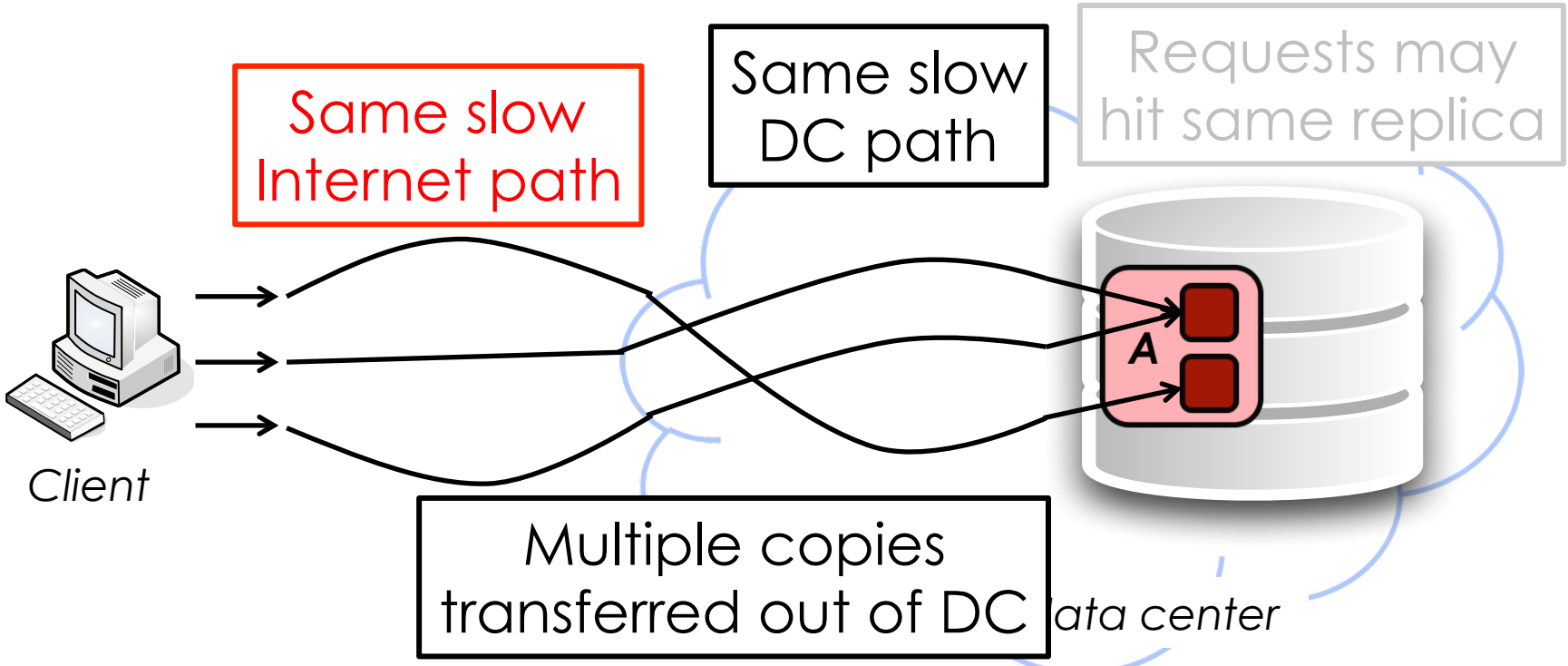


Use Multiple Copies of Object



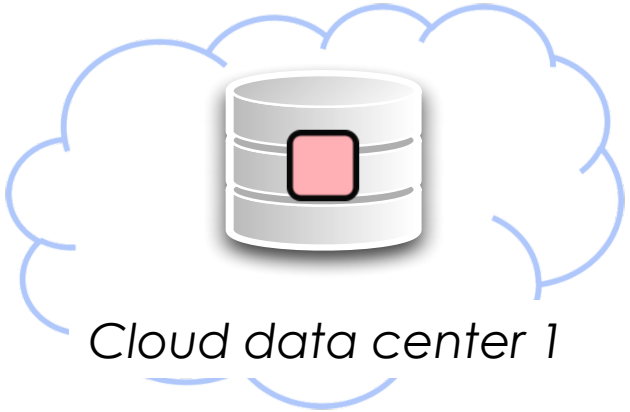
How To Use Redundancy?

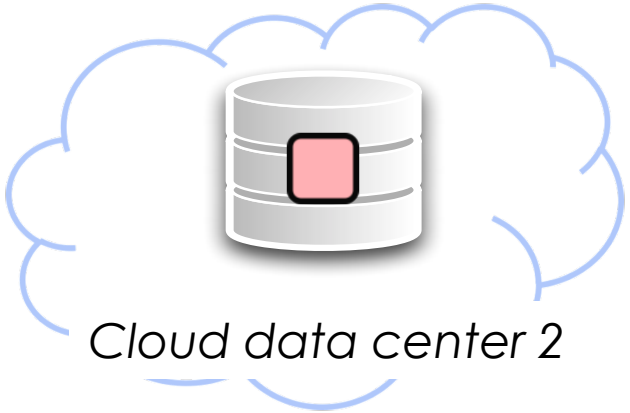
Simplest way: Redundant requests to same object



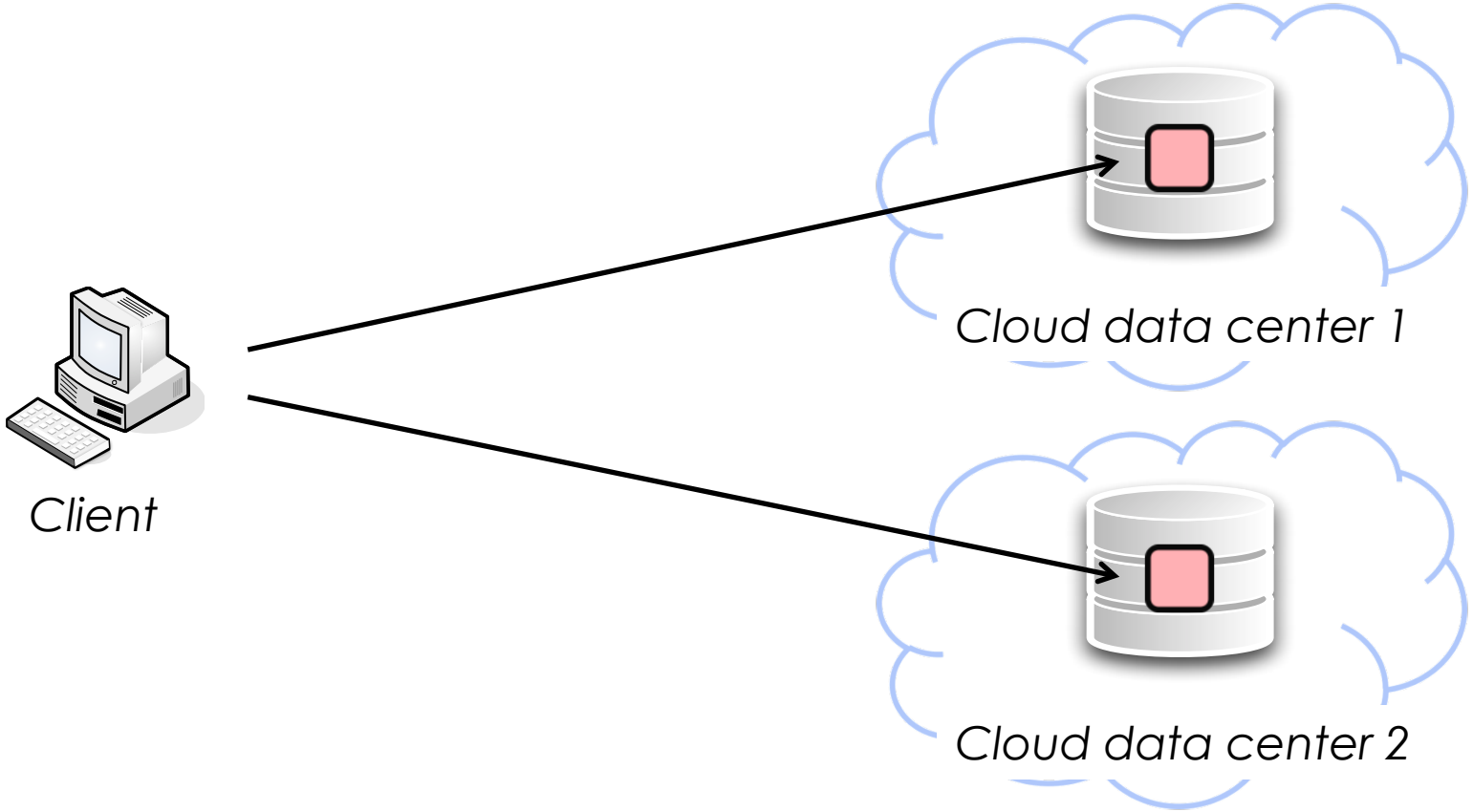
Use Multiple Data Centers

Client

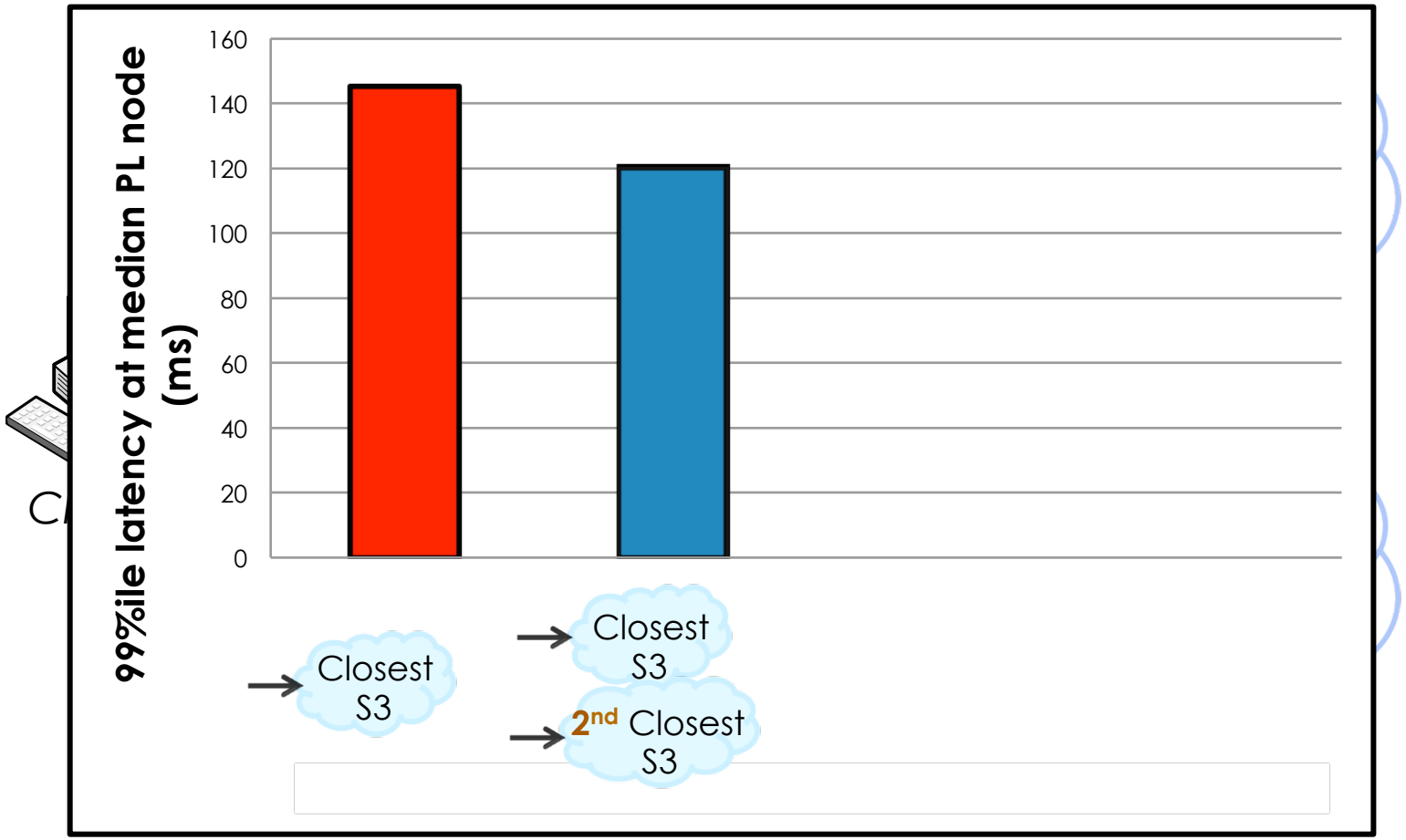




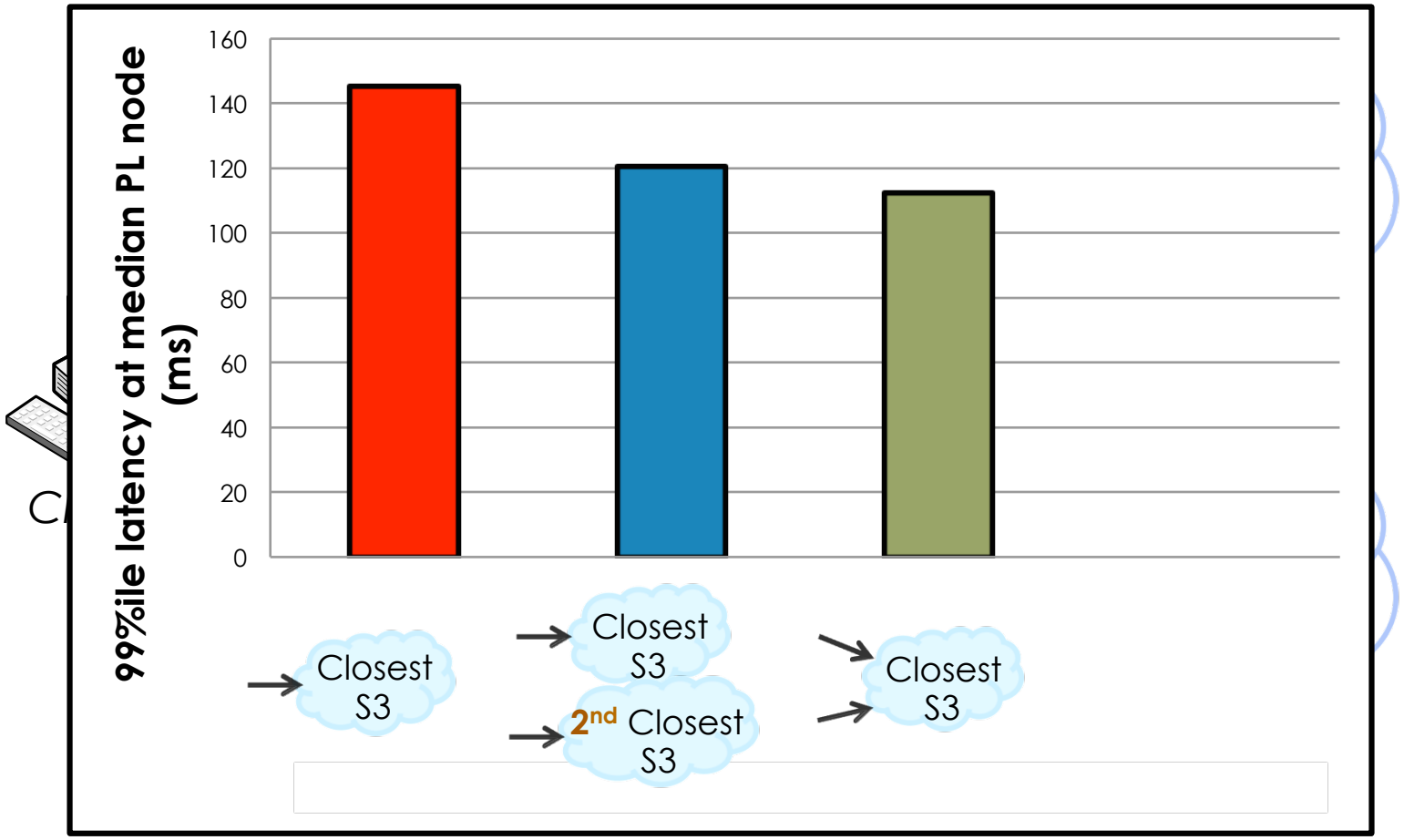
Use Multiple Data Centers



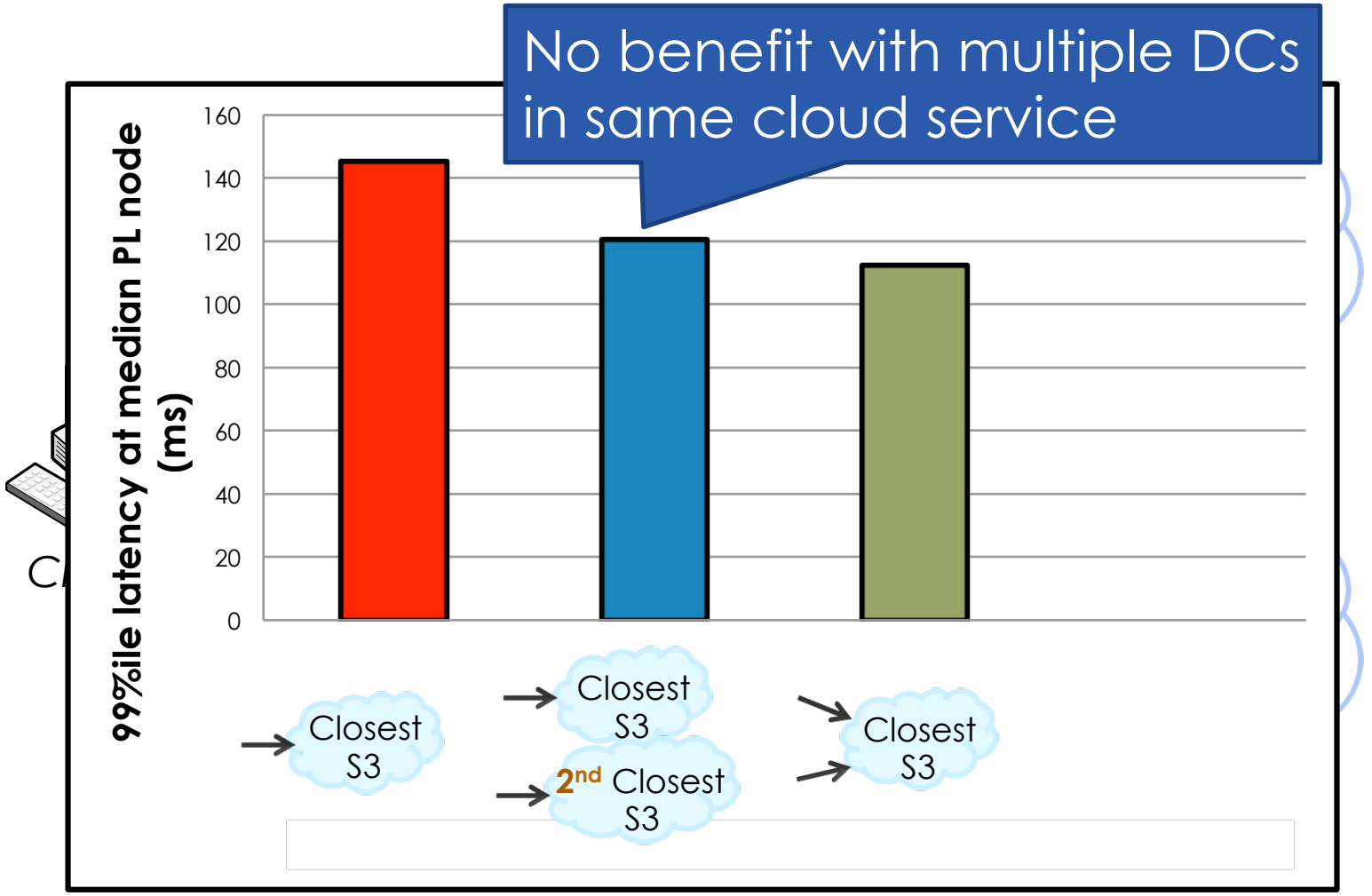
Use Multiple Data Centers



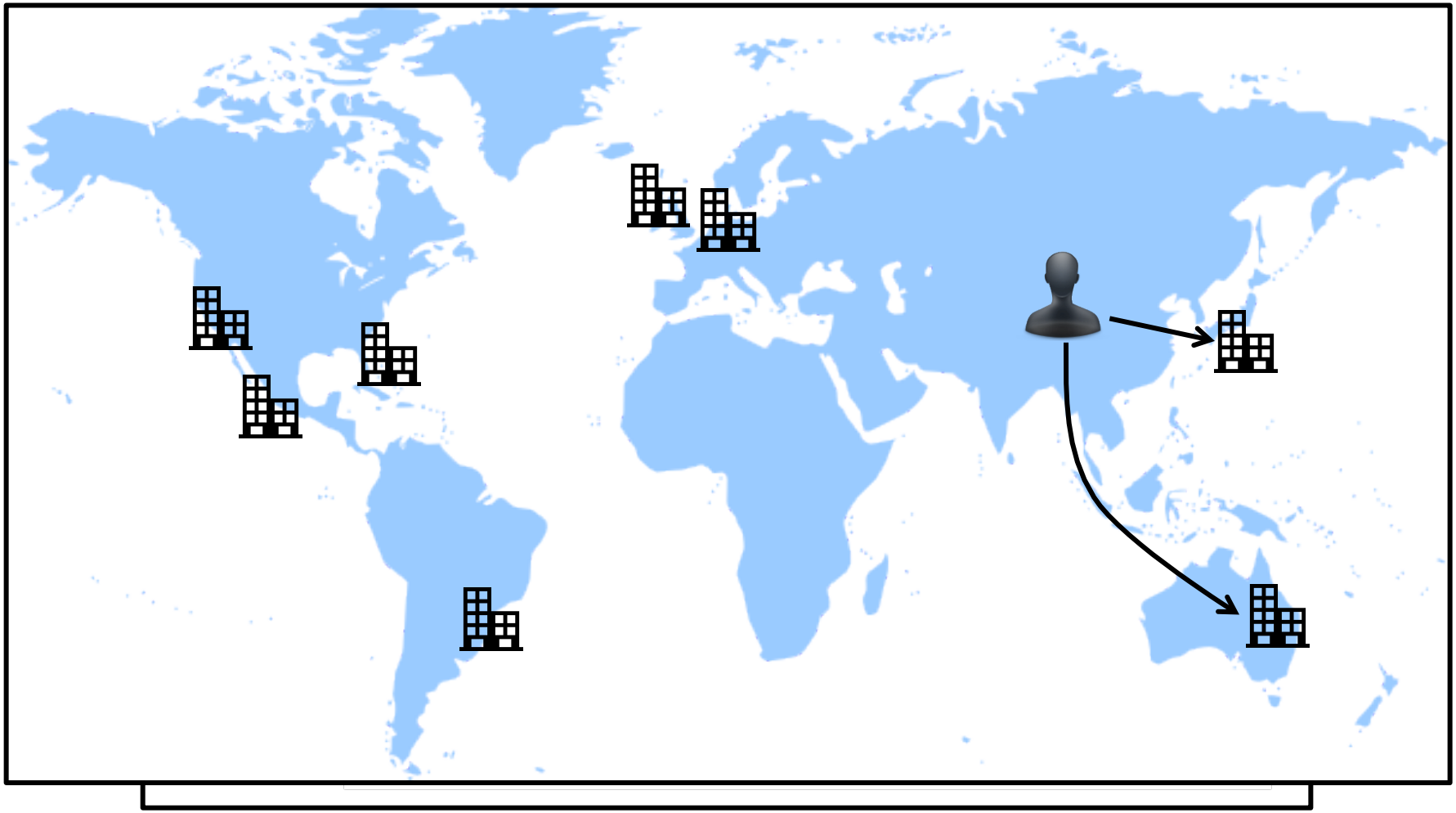
Use Multiple Data Centers



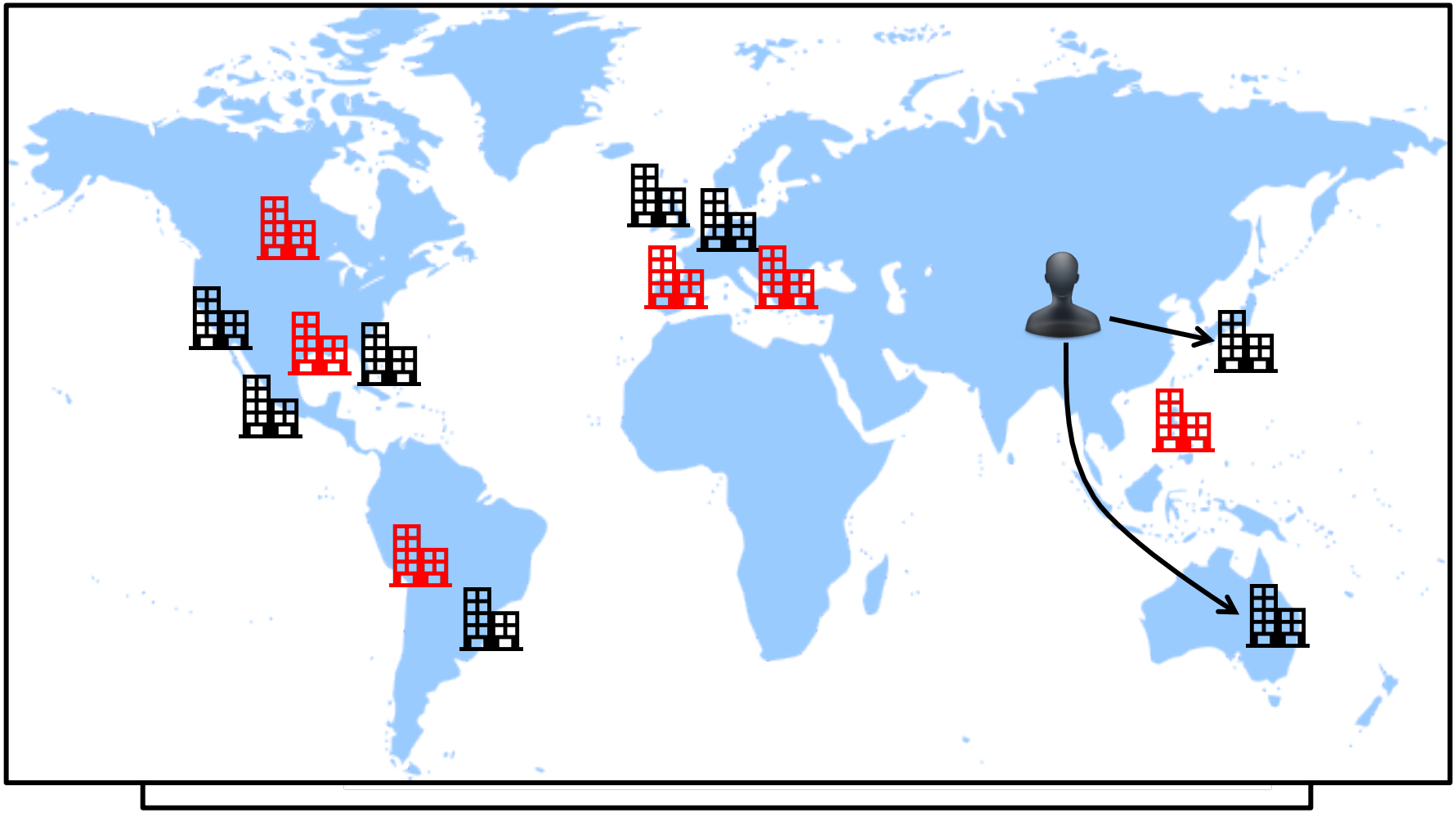
Use Multiple Data Centers



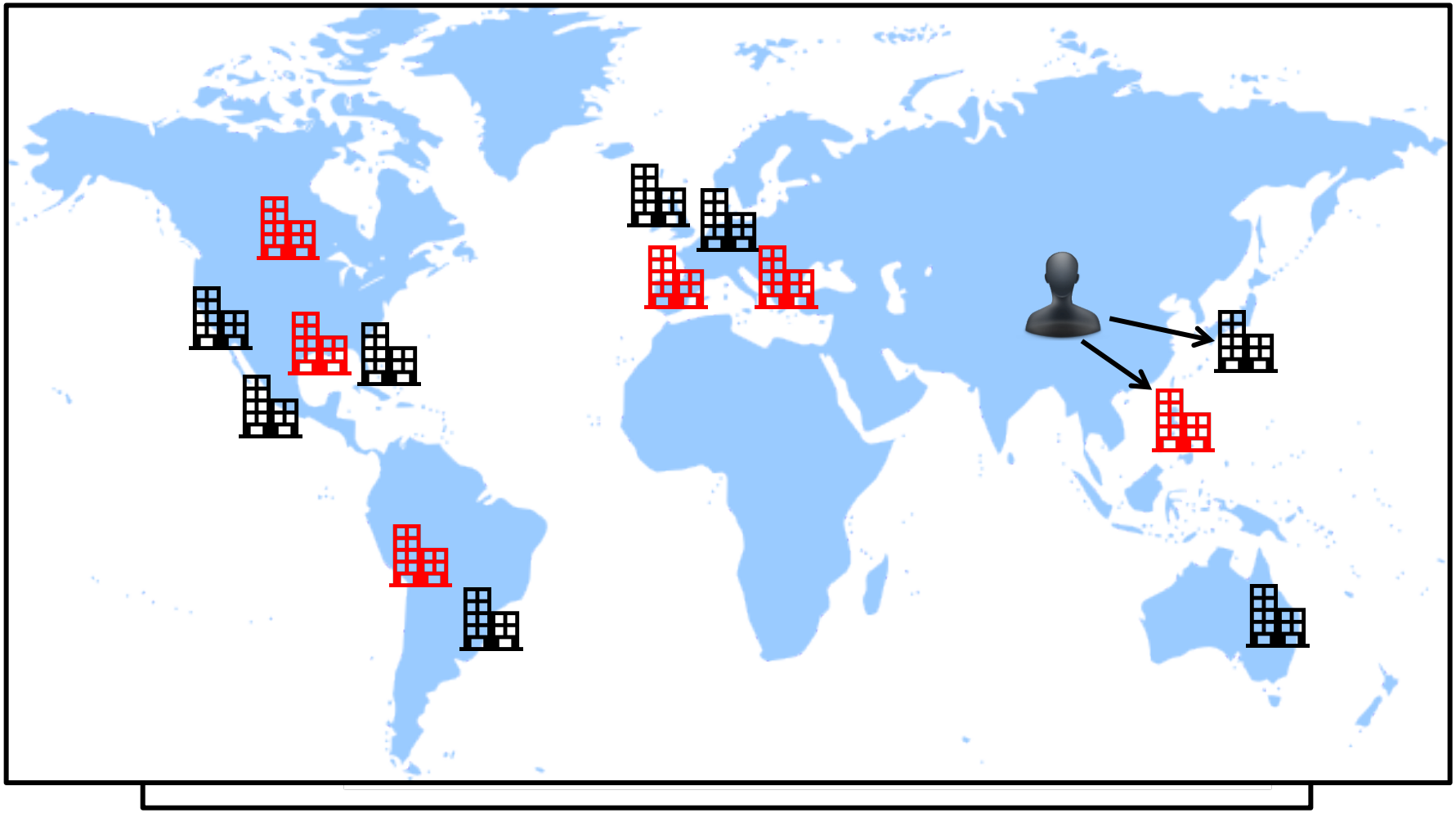
Use Multiple Data Centers



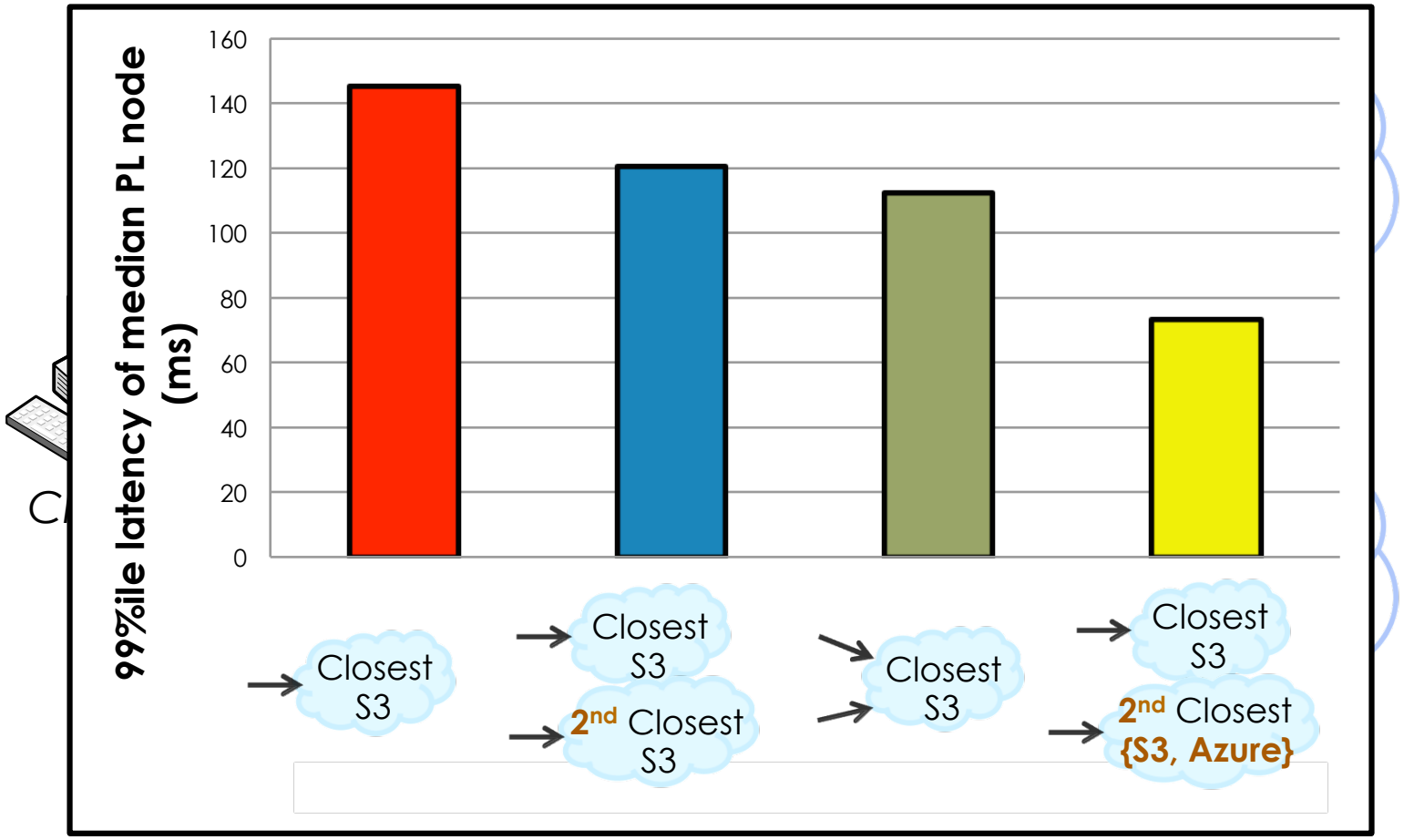
Use Multiple Clouds



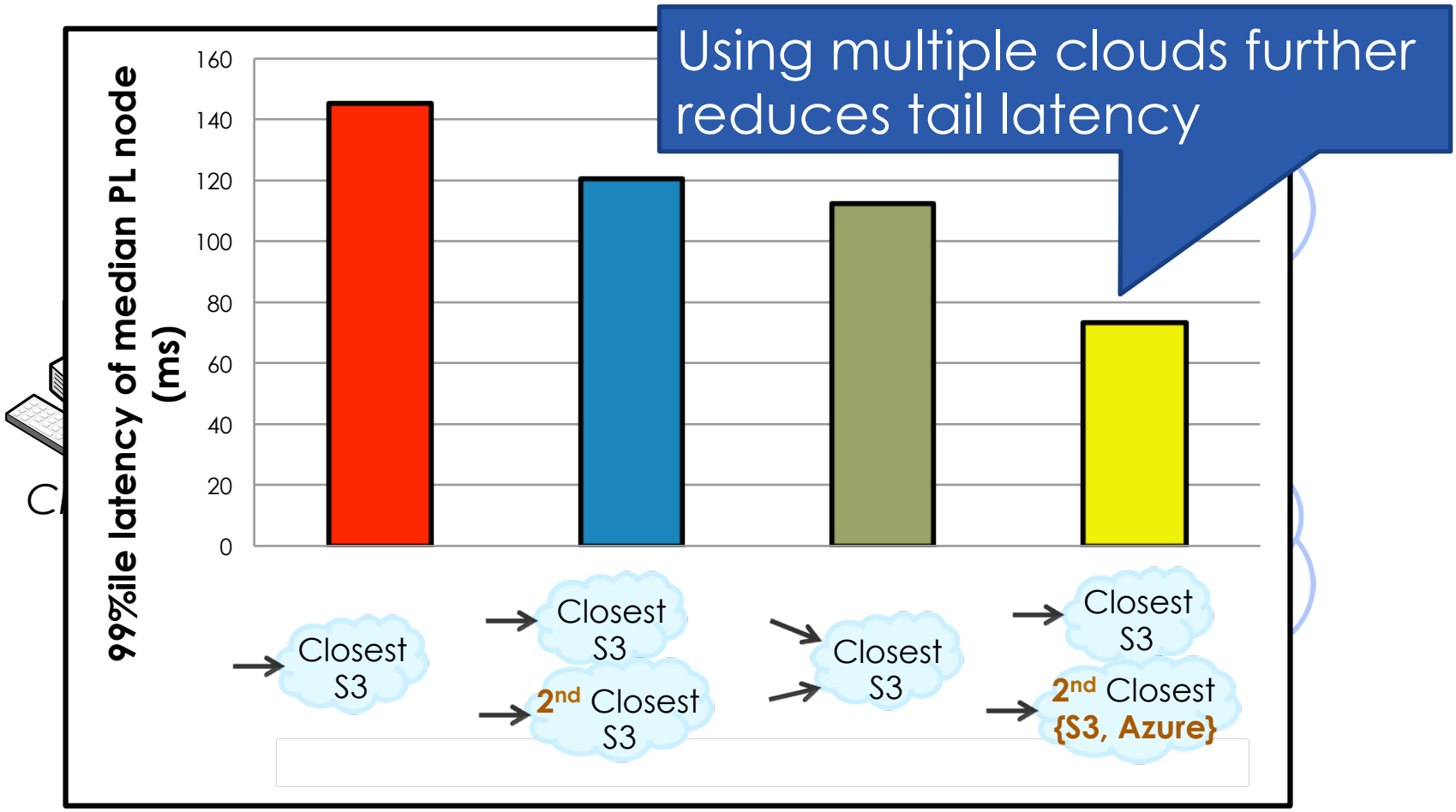
Use Multiple Clouds



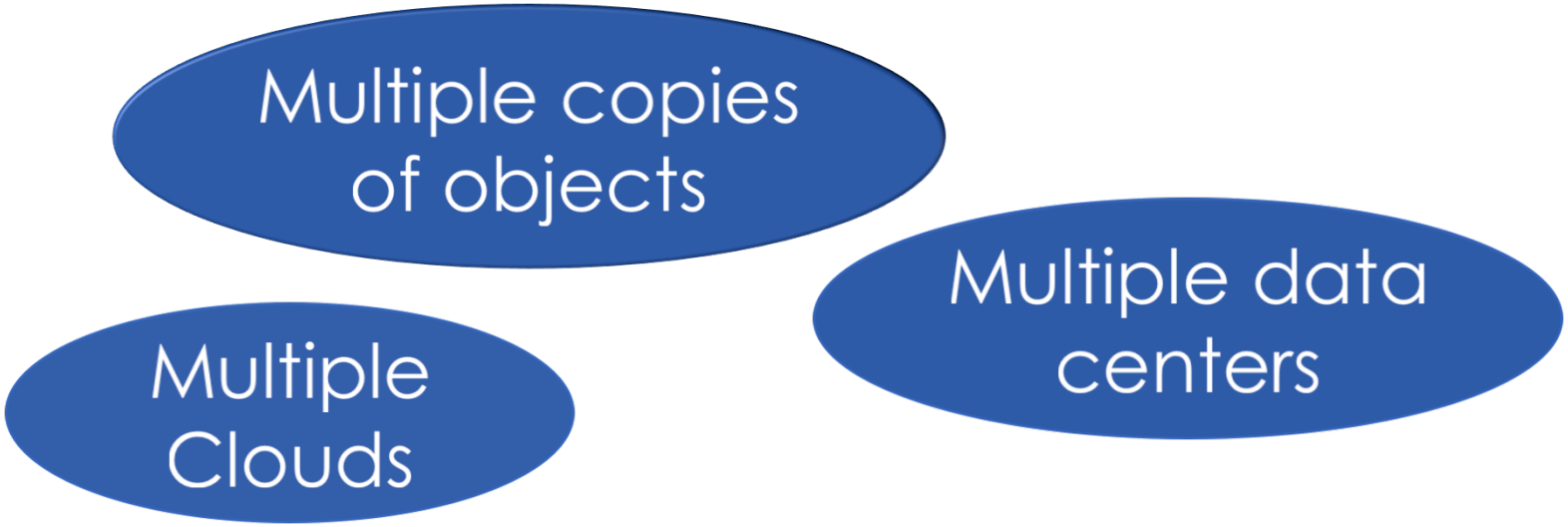
Use Multiple Clouds



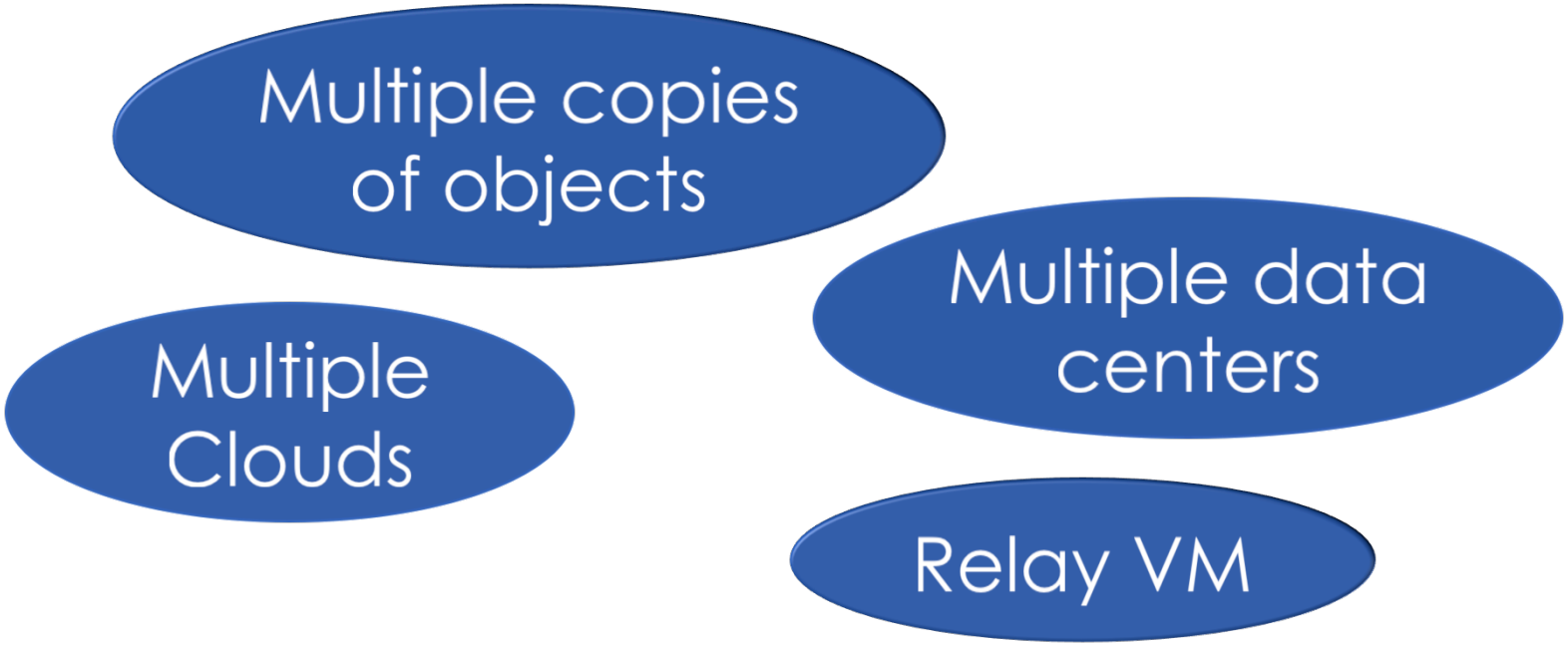
Use Multiple Clouds



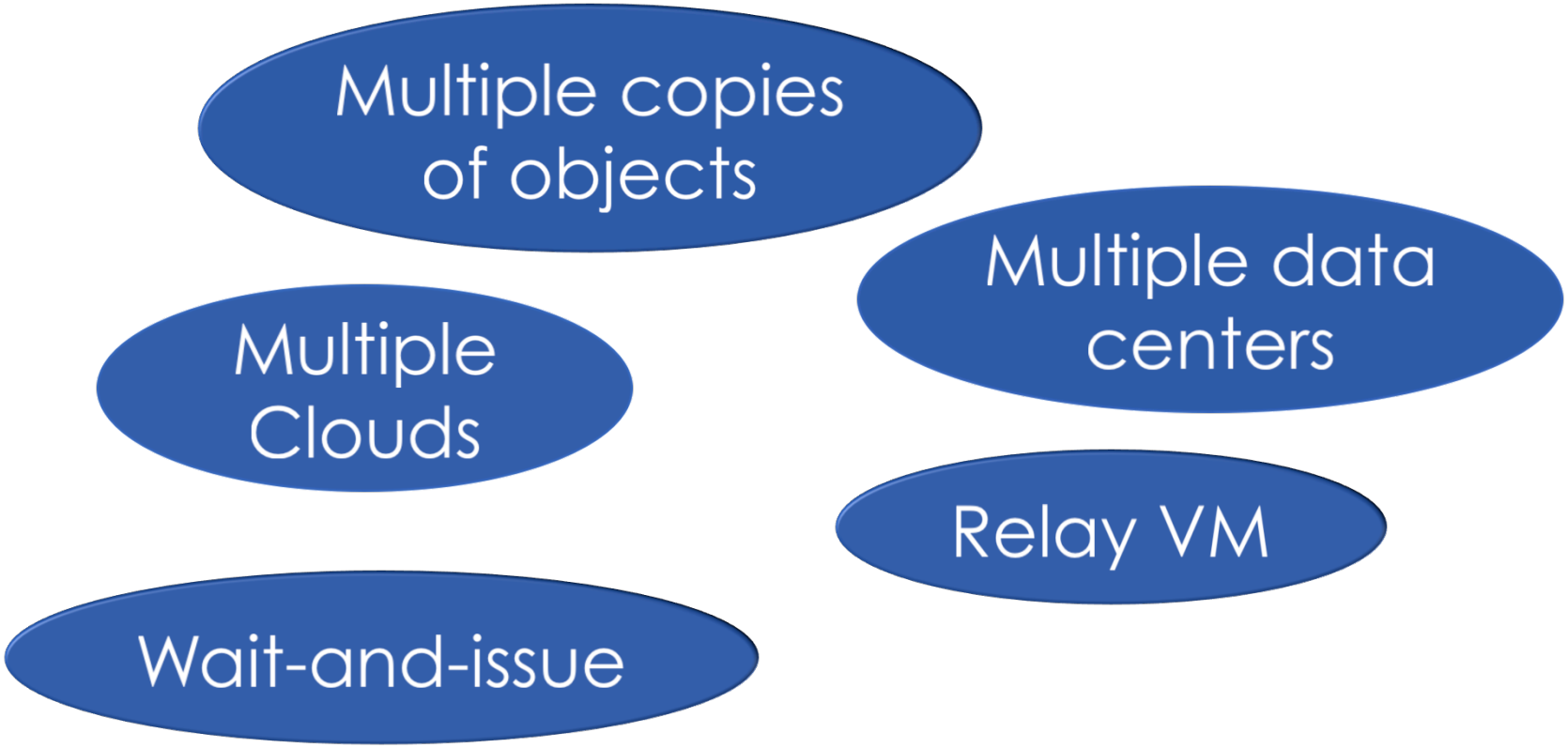
Many Forms of Redundancy



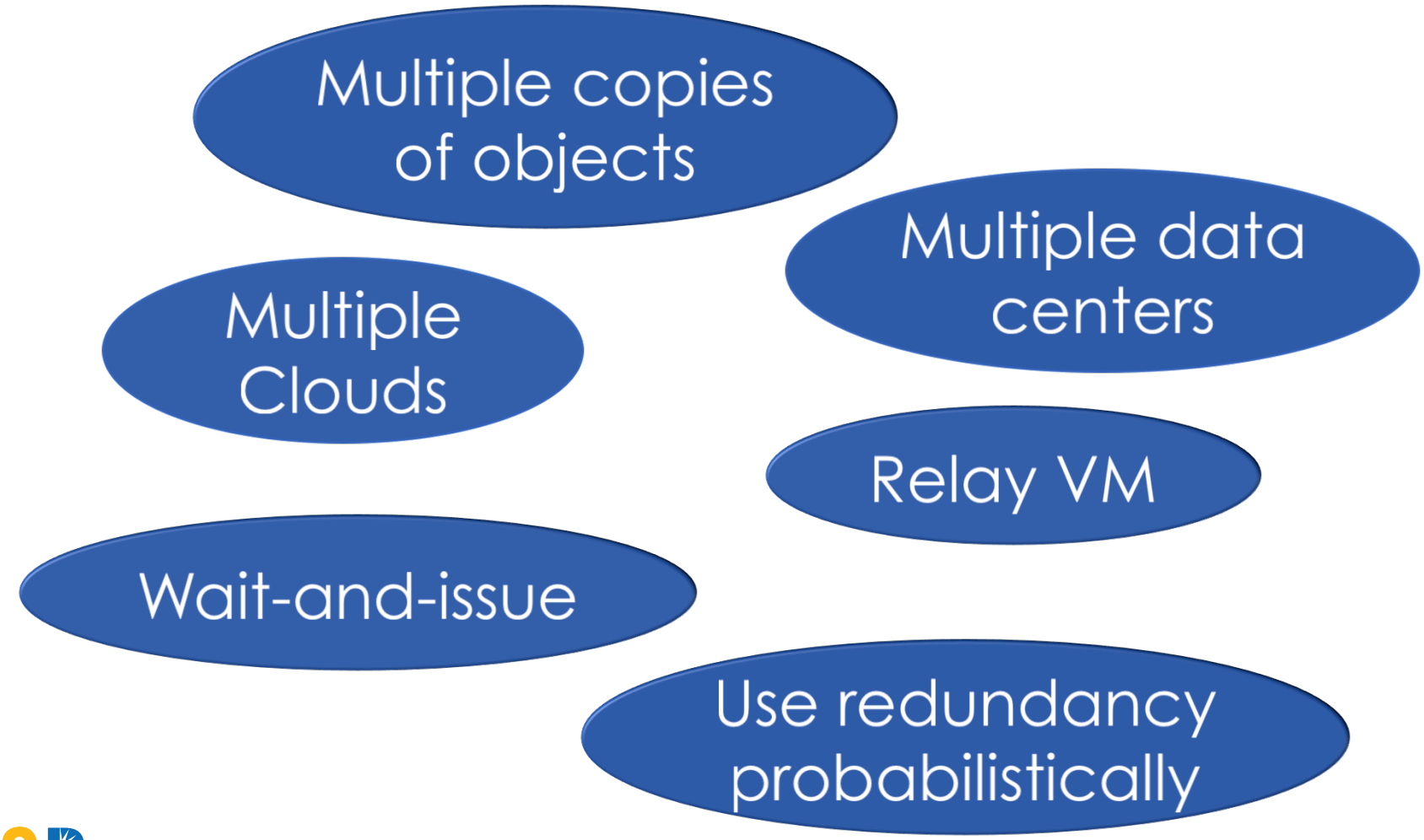
Many Forms of Redundancy



Many Forms of Redundancy

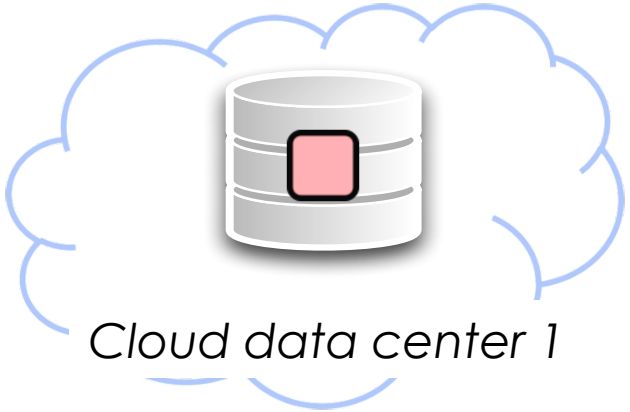


Many Forms of Redundancy

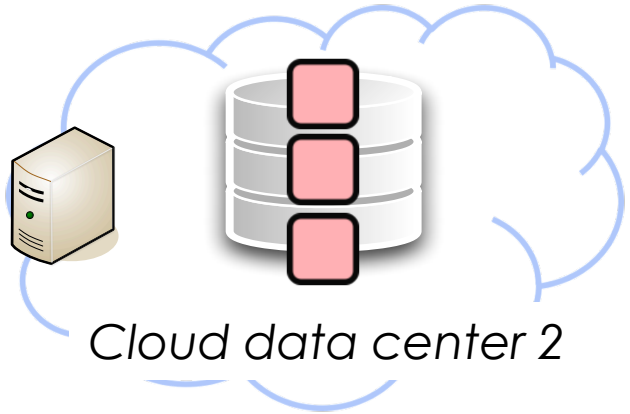


Example Configuration

Client

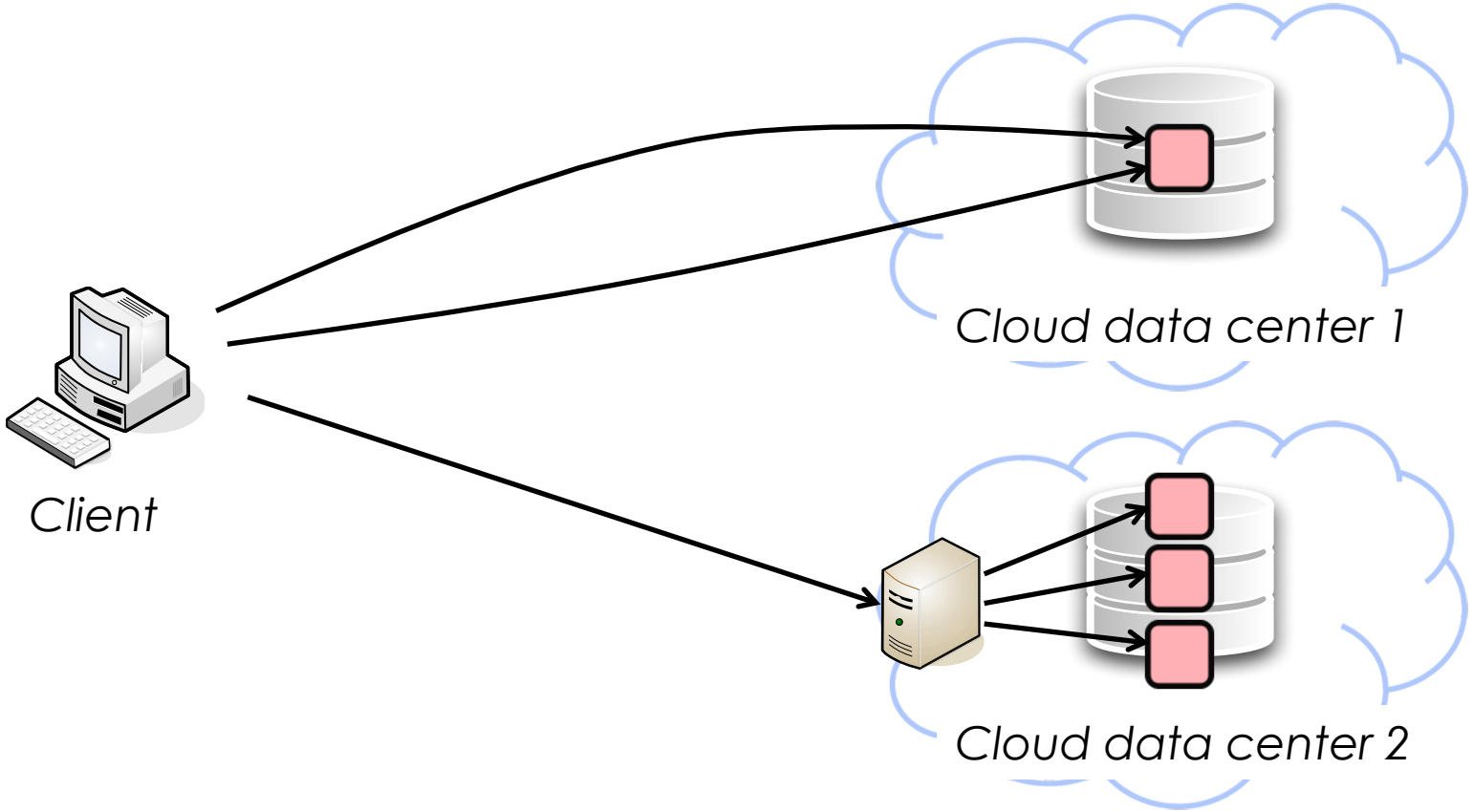


Cloud data center 1



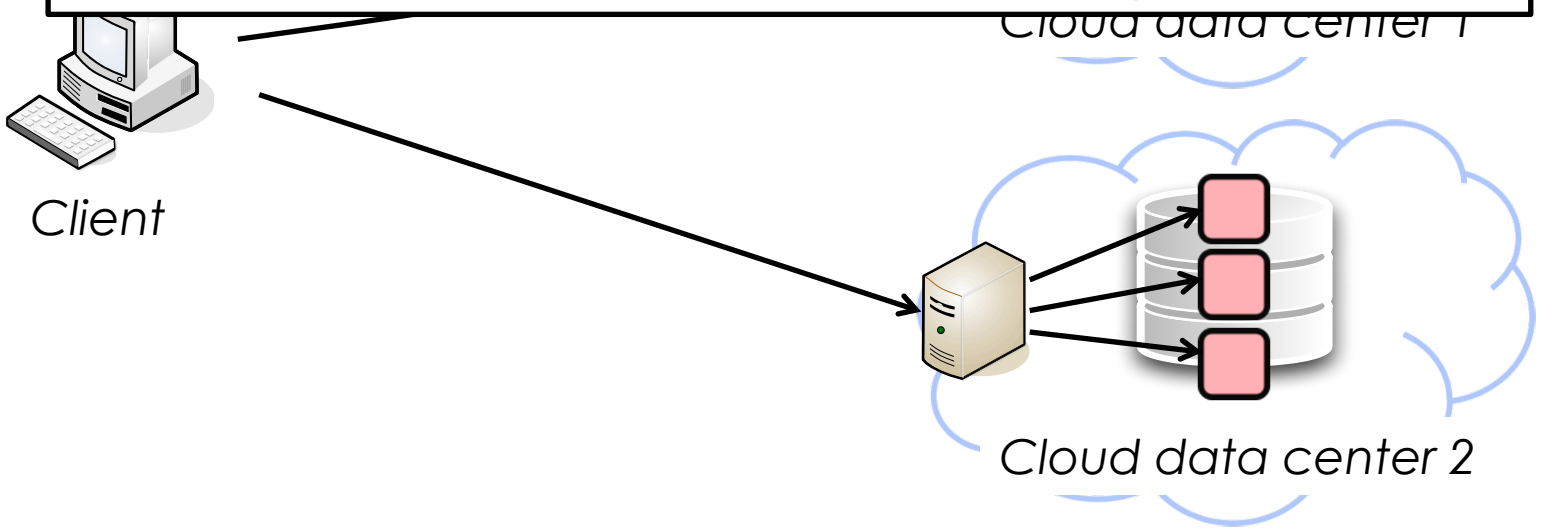
Cloud data center 2

Example Configuration

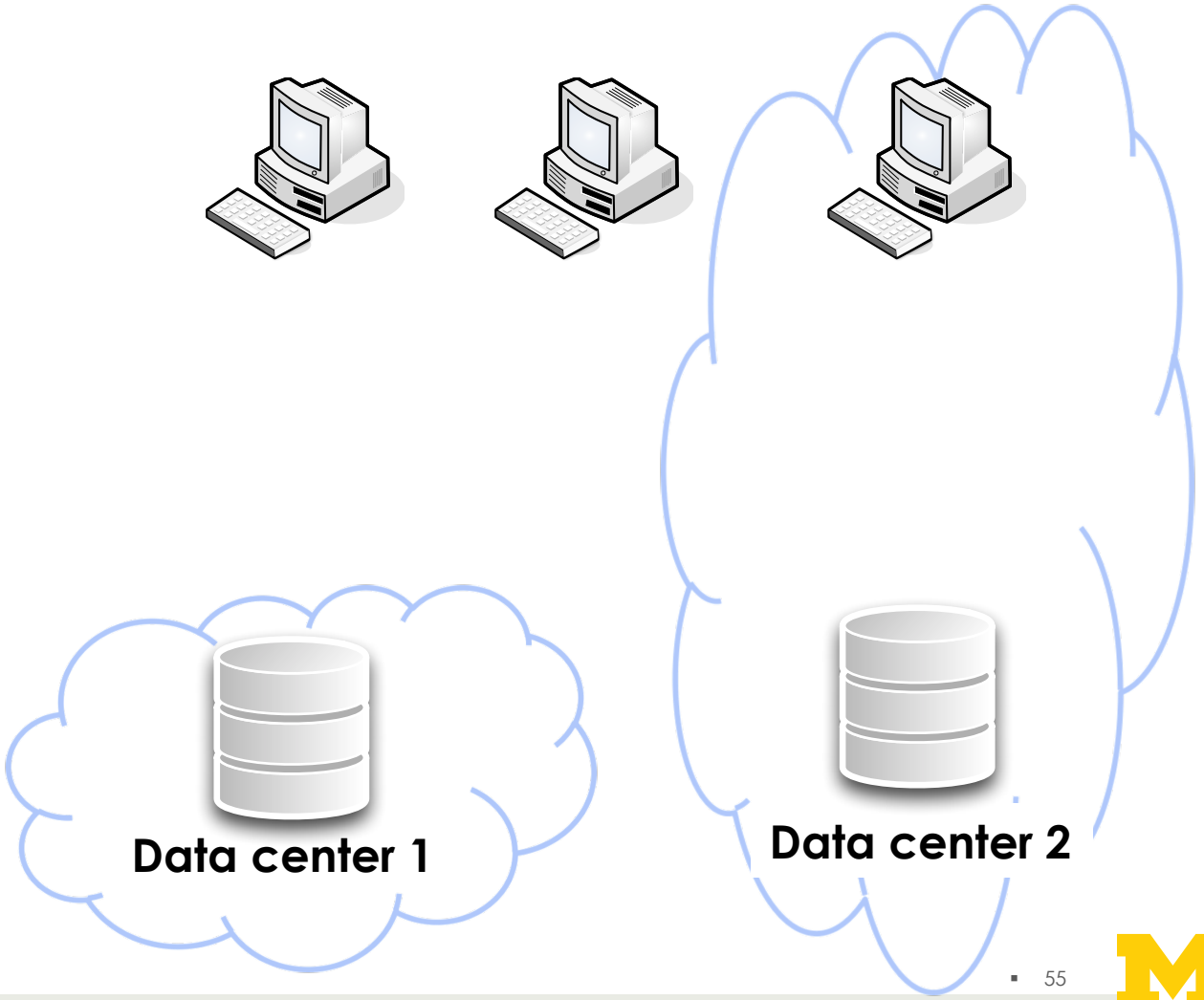


Example Configuration

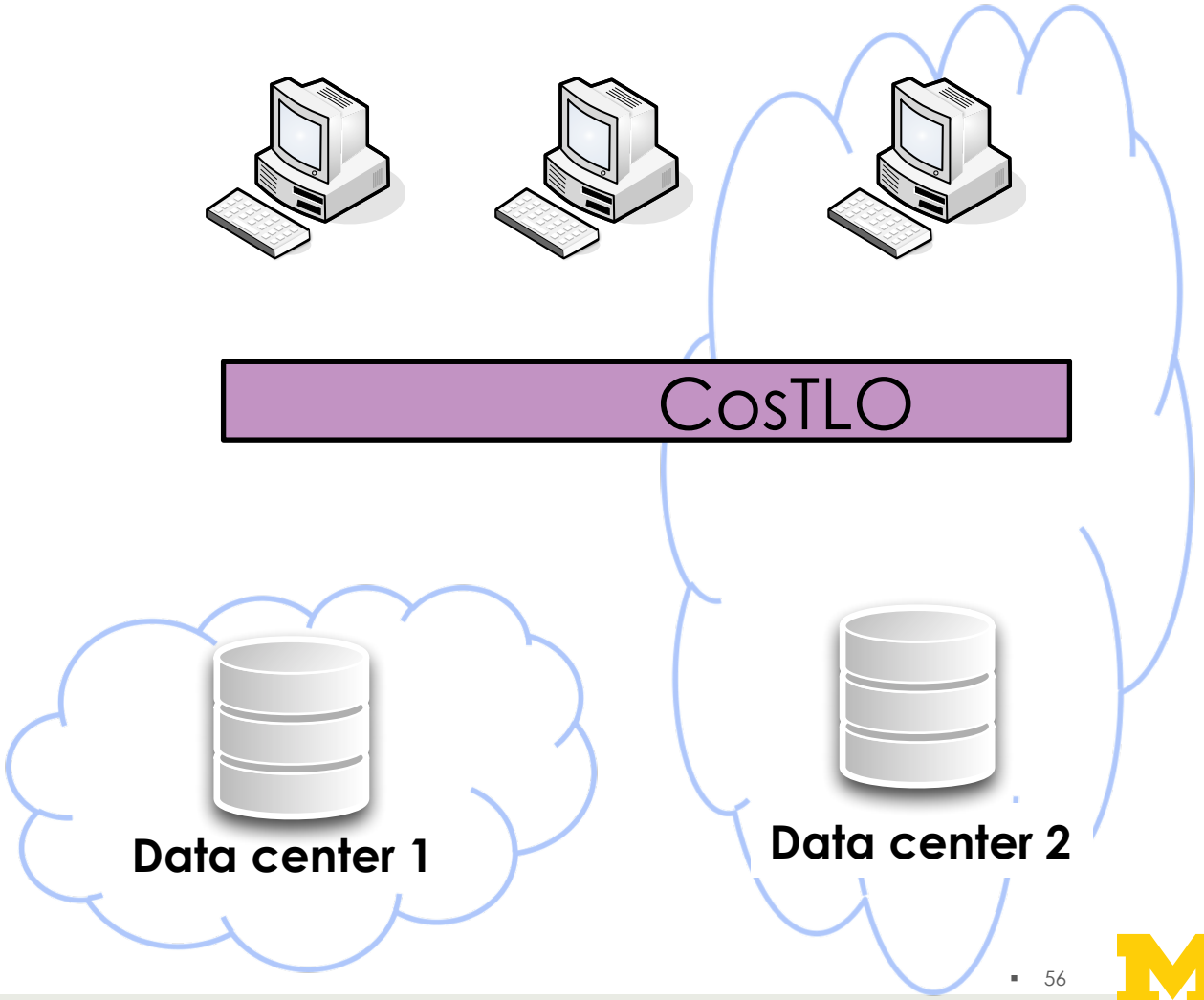
Challenge: How to pick from the unbounded set of configurations to satisfy application goal?



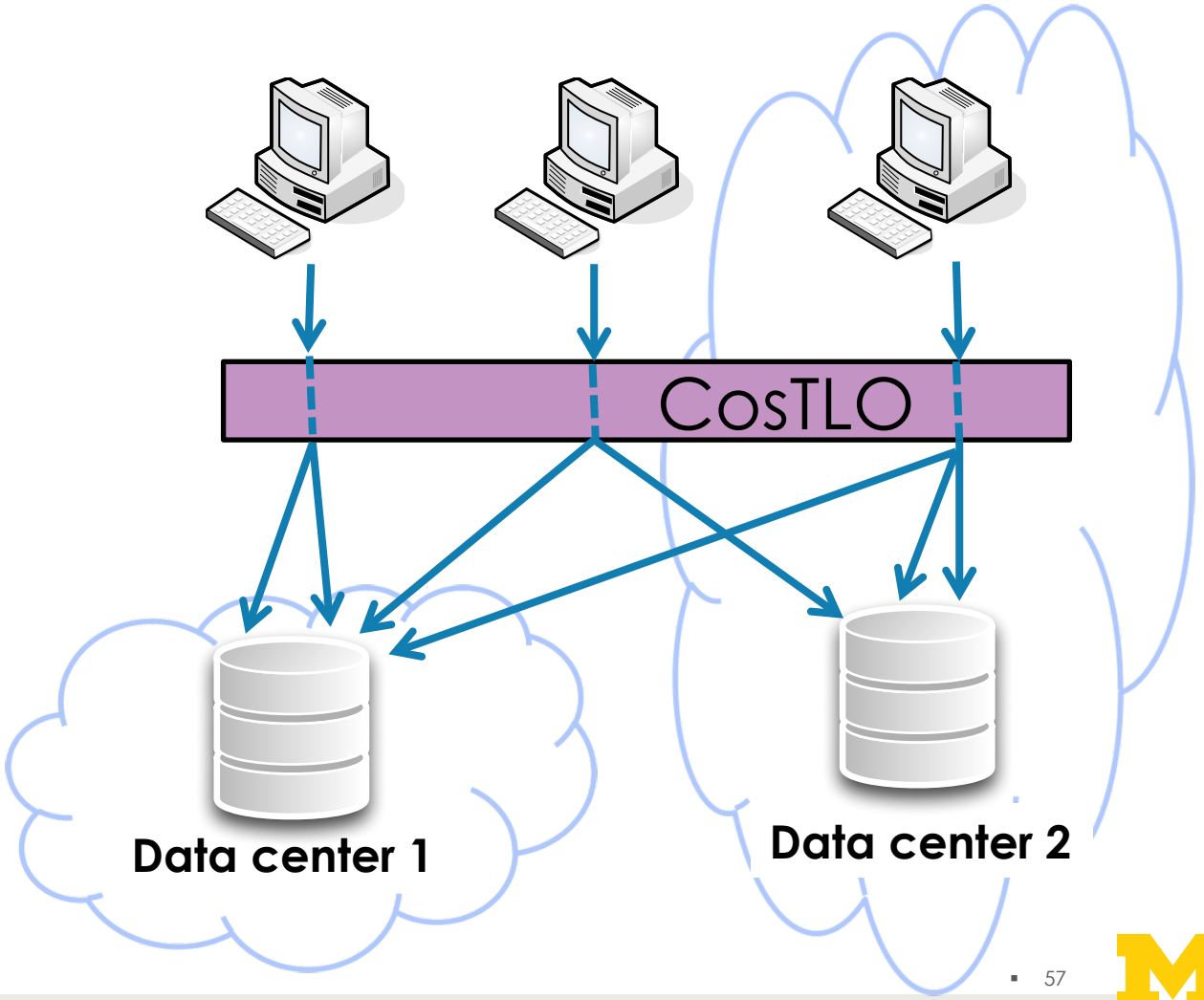
CosTLO: Cost-Effective Tail Latency Optimizer



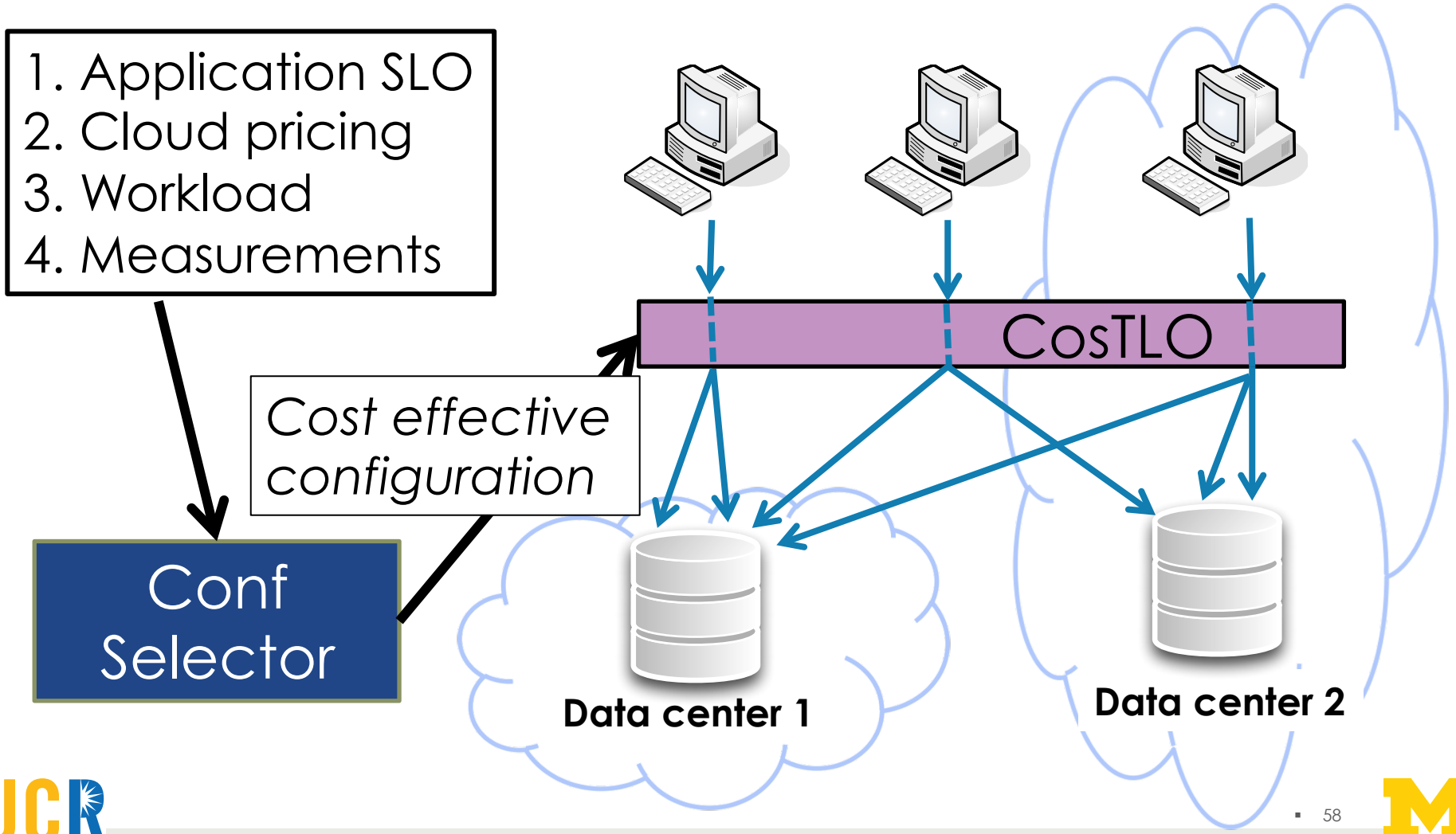
CosTLO: Cost-Effective Tail Latency Optimizer



CosTLO: Cost-Effective Tail Latency Optimizer



CosTLO: Cost-Effective Tail Latency Optimizer



Challenges

- How to search configuration space to find cost effective configuration?
- How to estimate latency distribution for any configuration?
- How to guarantee data consistency despite concurrent PUTs?

Challenges

- How to search configuration space to find cost effective configuration?
- How to estimate latency distribution for any configuration?
- How to guarantee data consistency despite concurrent PUTs?

Challenge: Latency Estimation

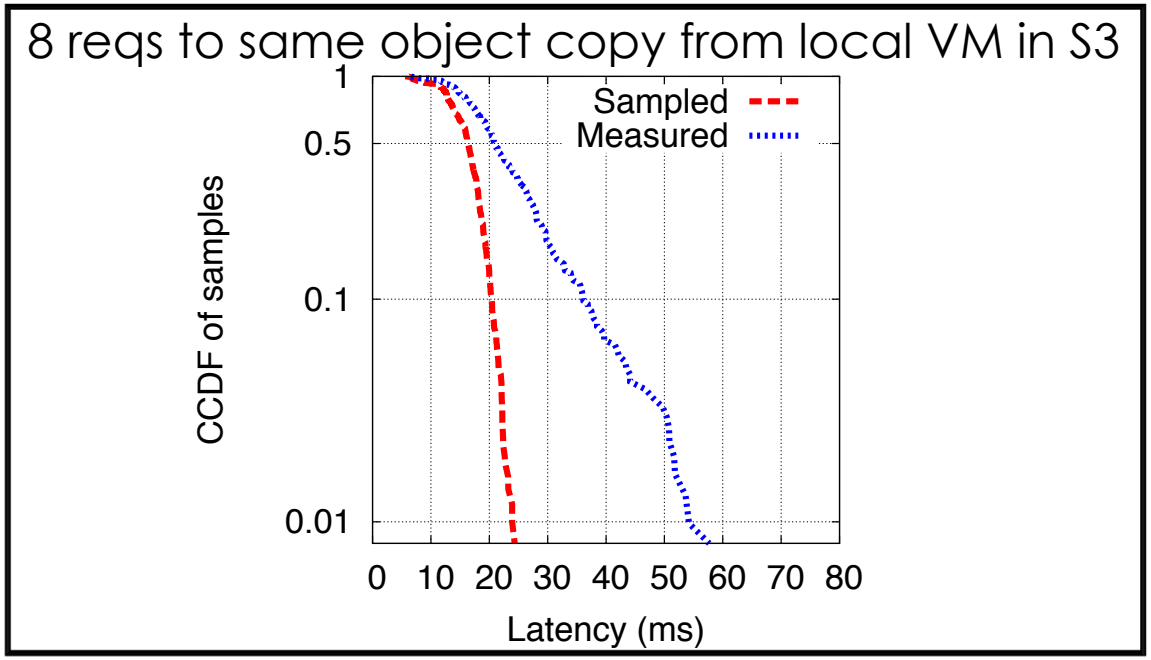
- How to estimate, rather than measure, the latency with any particular configuration?

Challenge: Latency Estimation

- How to estimate, rather than measure, the latency with any particular configuration?
- Simplest way: sample from single request distribution independently, and take the min

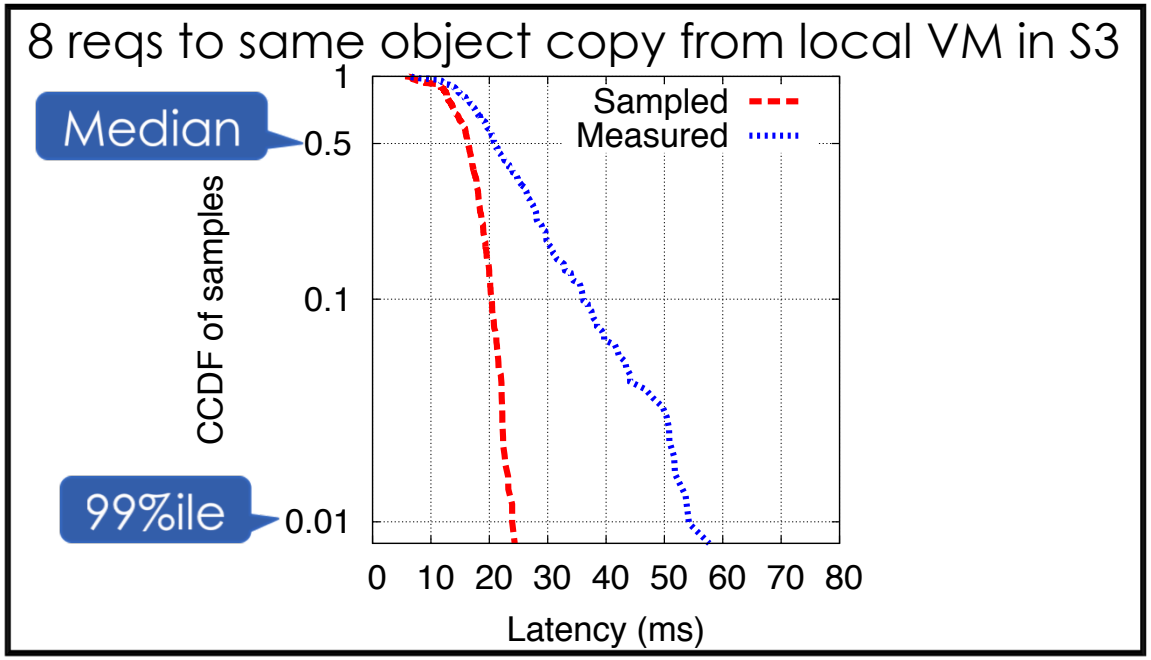
Challenge: Latency Estimation

- How to estimate, rather than measure, the latency with any particular configuration?
- Simplest way: sample from single request distribution independently, and take the min



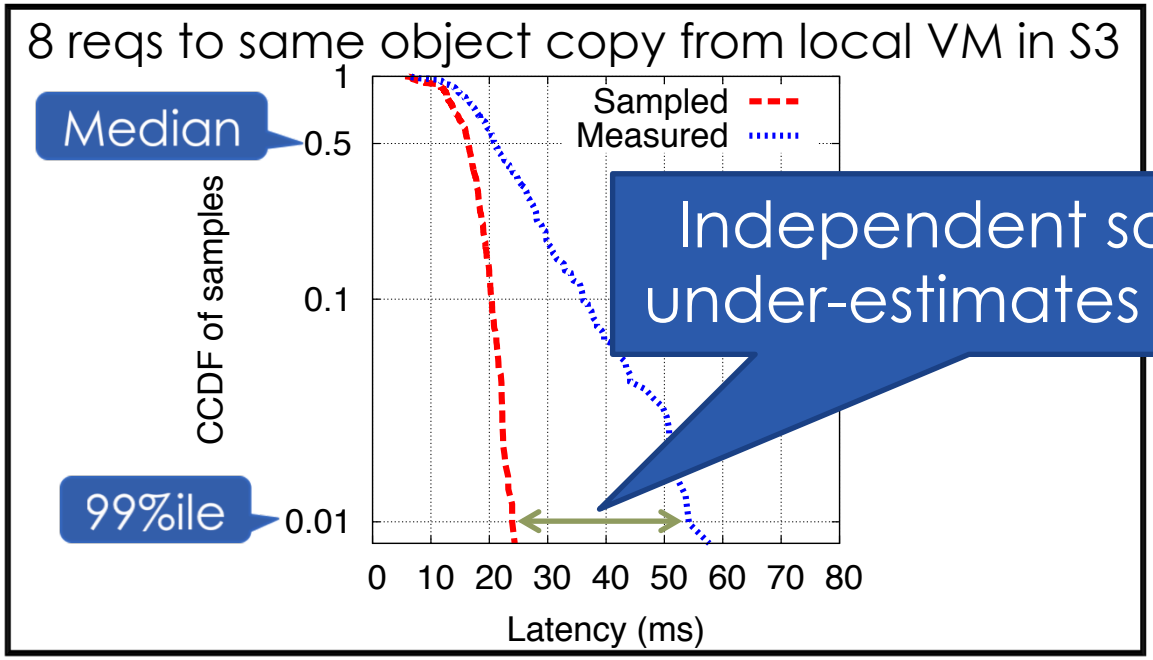
Challenge: Latency Estimation

- How to estimate, rather than measure, the latency with any particular configuration?
- Simplest way: sample from single request distribution independently, and take the min

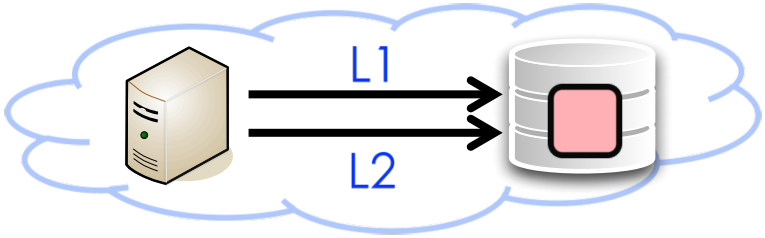


Challenge: Latency Estimation

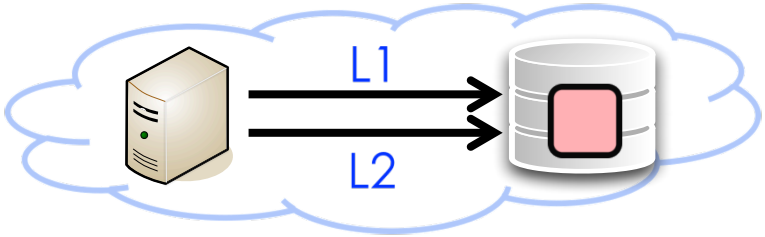
- How to estimate, rather than measure, the latency with any particular configuration?
- Simplest way: sample from single request distribution independently, and take the min



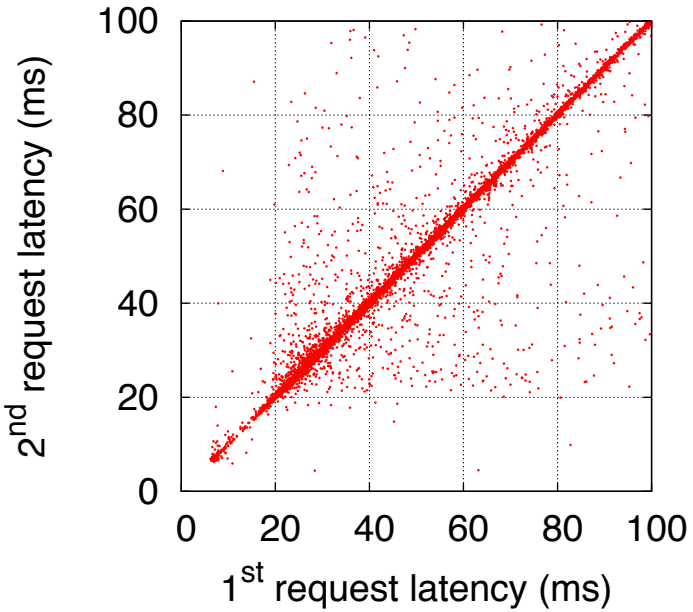
Problem: Inter-Request Dependency



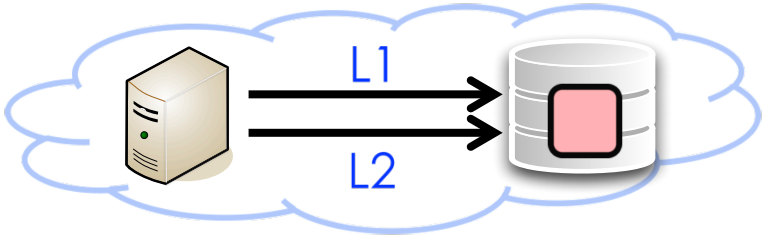
Problem: Inter-Request Dependency



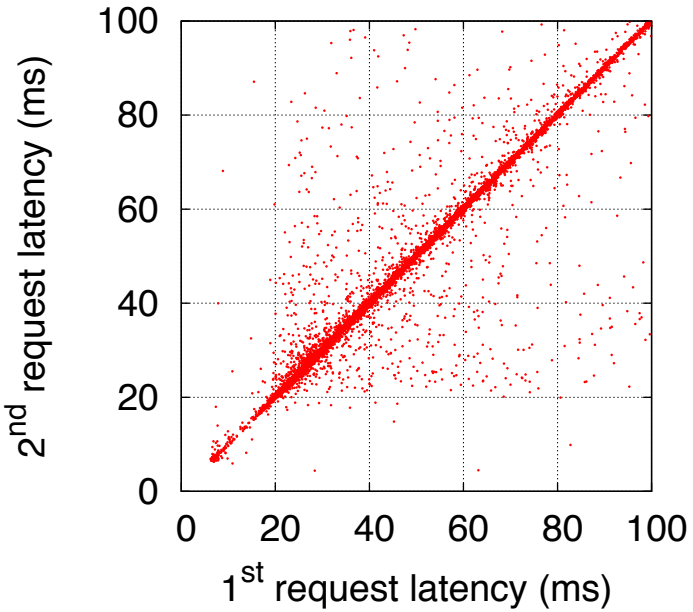
2 requests to same object in Azure



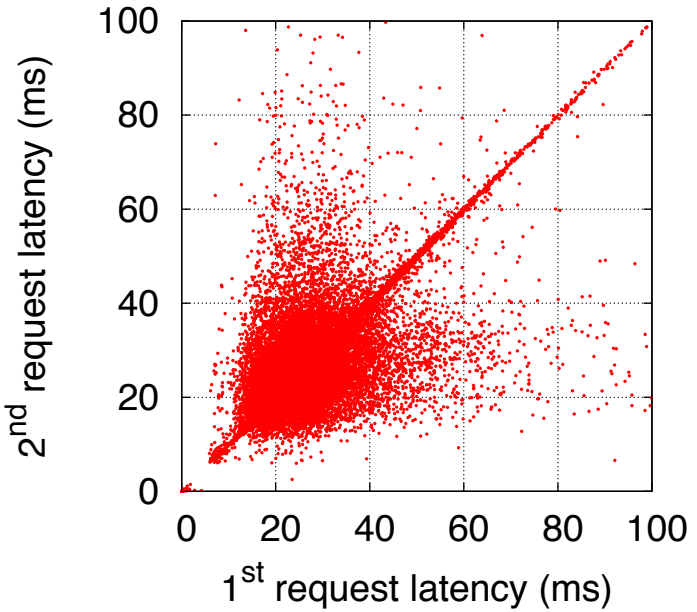
Problem: Inter-Request Dependency



2 requests to same object in Azure



2 requests to same object in S3

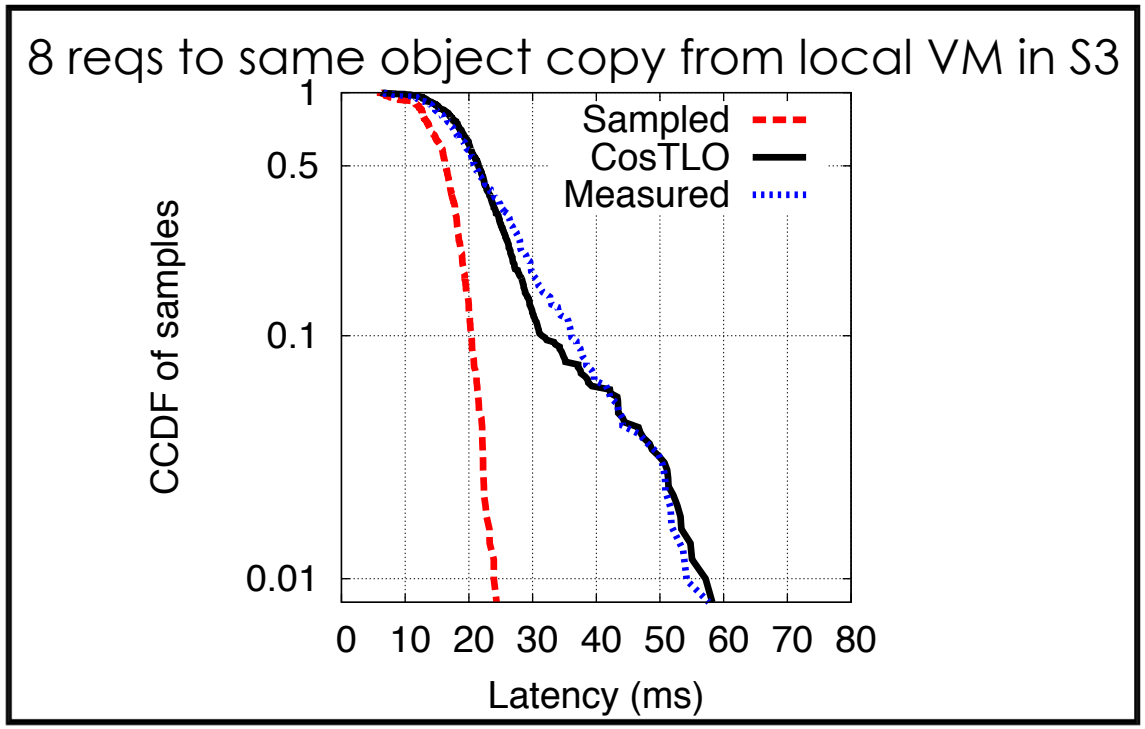


CosTLO Latency Estimation

- **Explicitly model sources of dependency**
  - Concurrent requests hit same replica
  - Concurrent requests take same network path

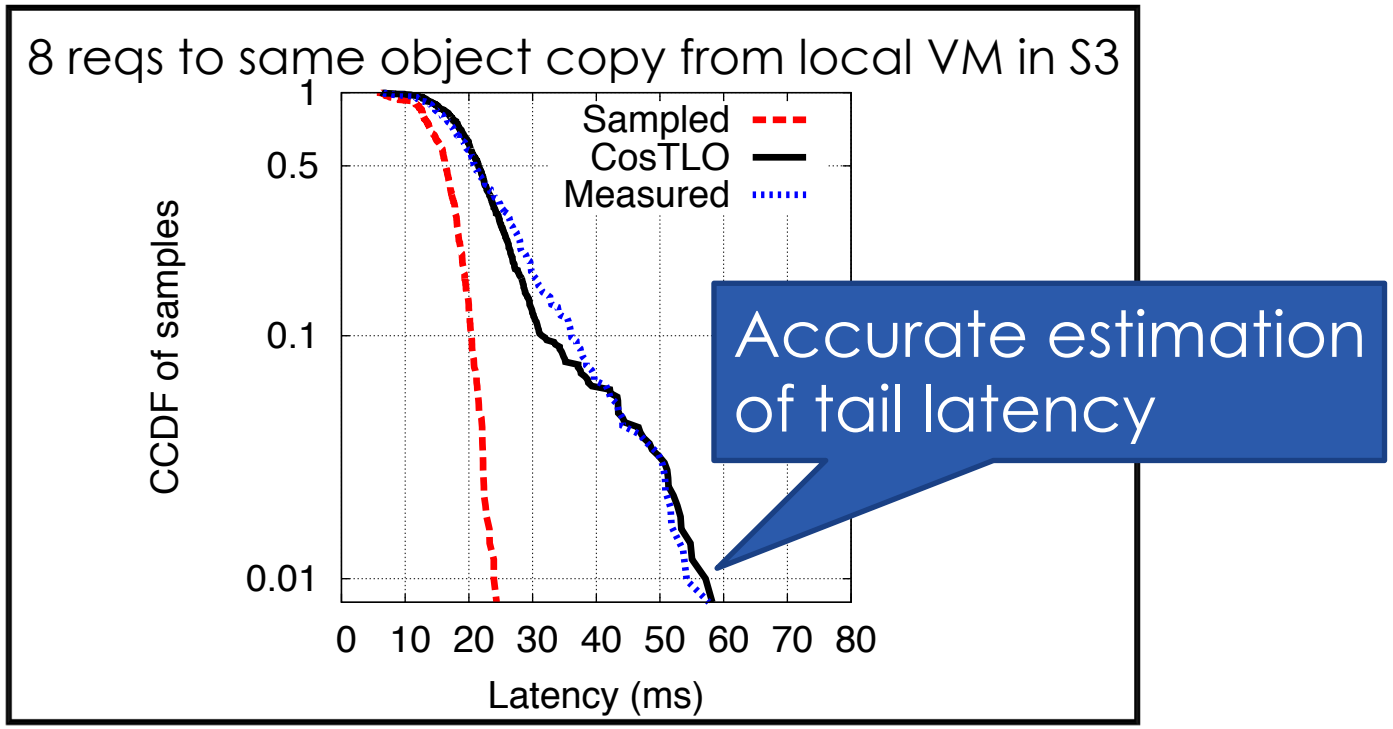
CosTLO Latency Estimation

- **Explicitly model sources of dependency**
  - Concurrent requests hit same replica
  - Concurrent requests take same network path



CosTLO Latency Estimation

- **Explicitly model sources of dependency**
  - Concurrent requests hit same replica
  - Concurrent requests take same network path



Evaluation

- **Questions**
  - Can CosTLO meet SLOs?
  - How useful are different forms of redundancy?
  - How much cost overhead does CosTLO incur?

Evaluation

- **Questions**
  - Can CosTLO meet SLOs?
  - How useful are different forms of redundancy?
  - How much cost overhead does CosTLO incur?

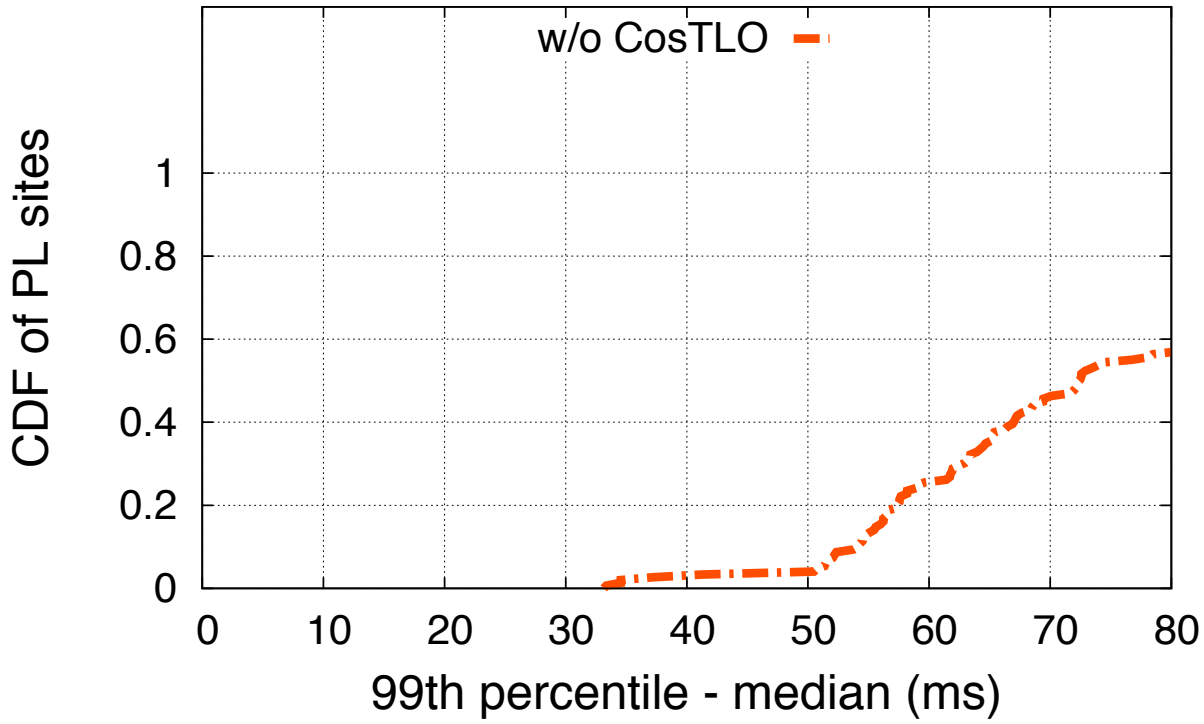
Evaluation

- **Questions**
  - Can CosTLO meet SLOs?
  - How useful are different forms of redundancy?
  - How much cost overhead does CosTLO incur?
- **Experiment setup**
  - Application is deployed on Amazon AWS
  - CosTLO is deployed on S3 and Azure
  - 120 PlanetLab nodes as clients
  - 1 week long Wikipedia workload

CosTLO Satisfies SLOs

Single request SLO

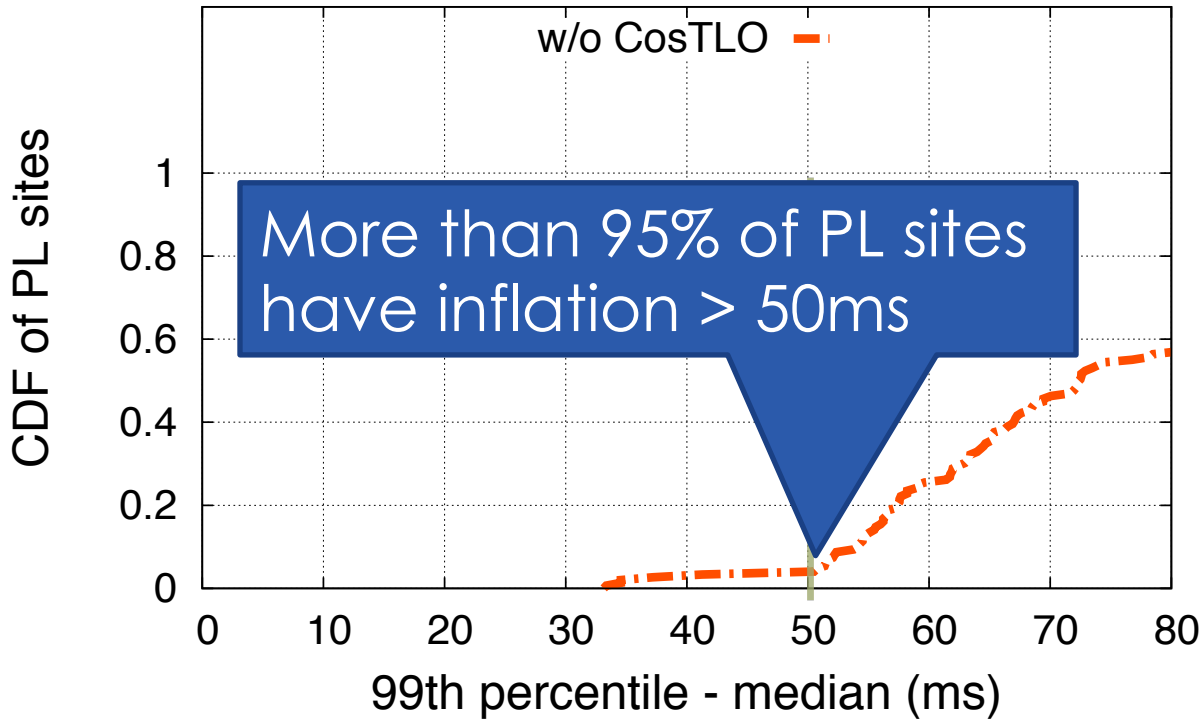
99%ile – median <= X ms



CosTLO Satisfies SLOs

Single request SLO

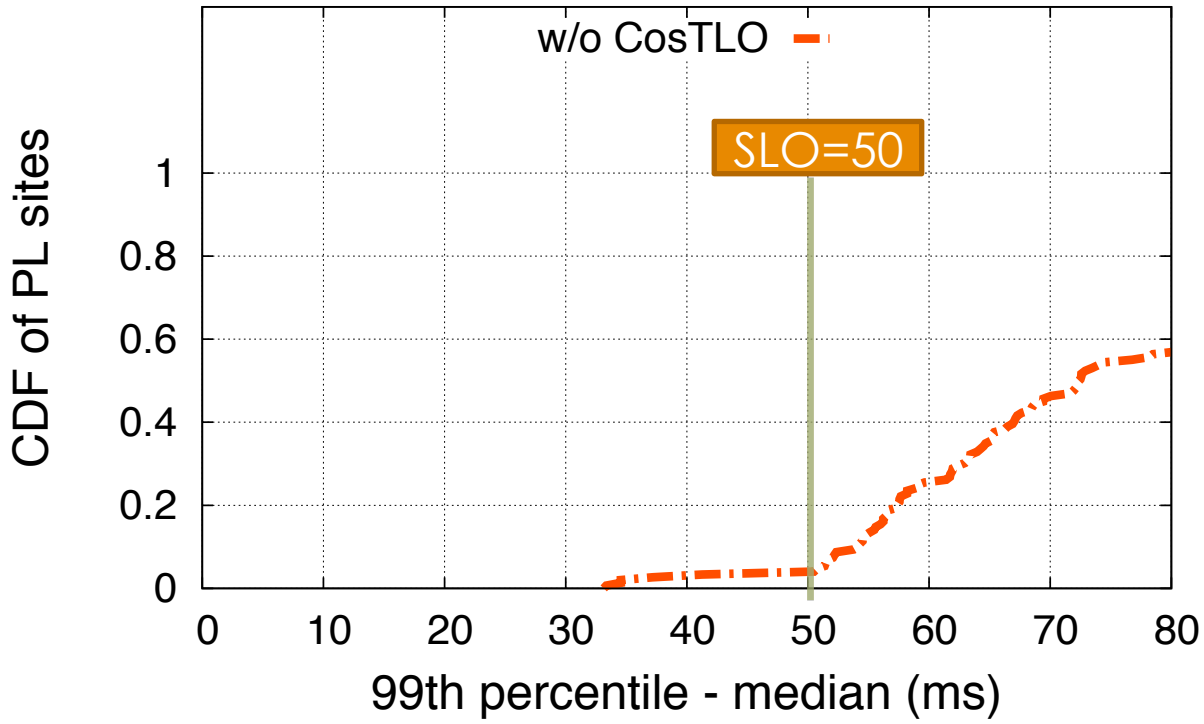
99%ile – median <= X ms



CosTLO Satisfies SLOs

Single request SLO

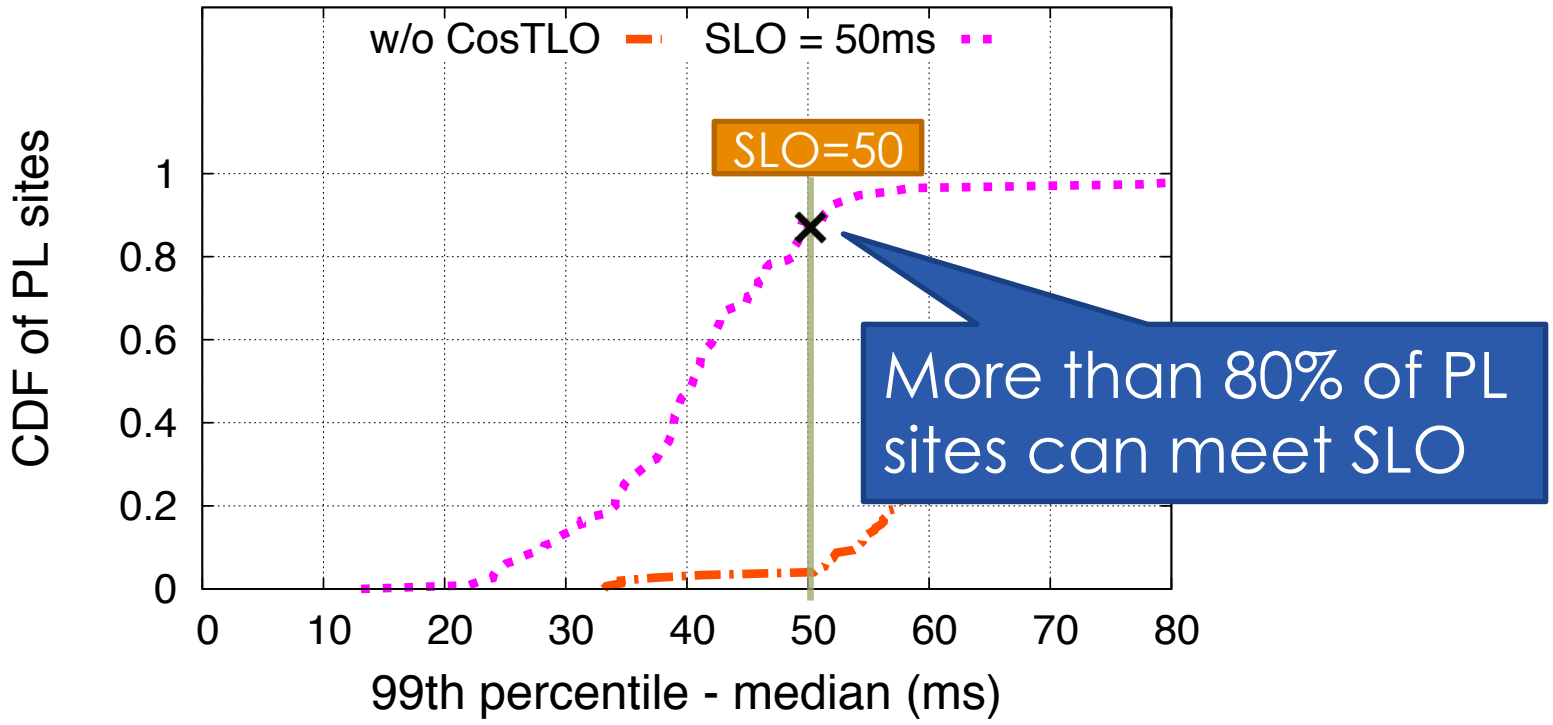
99%ile – median <= X ms



CosTLO Satisfies SLOs

Single request SLO

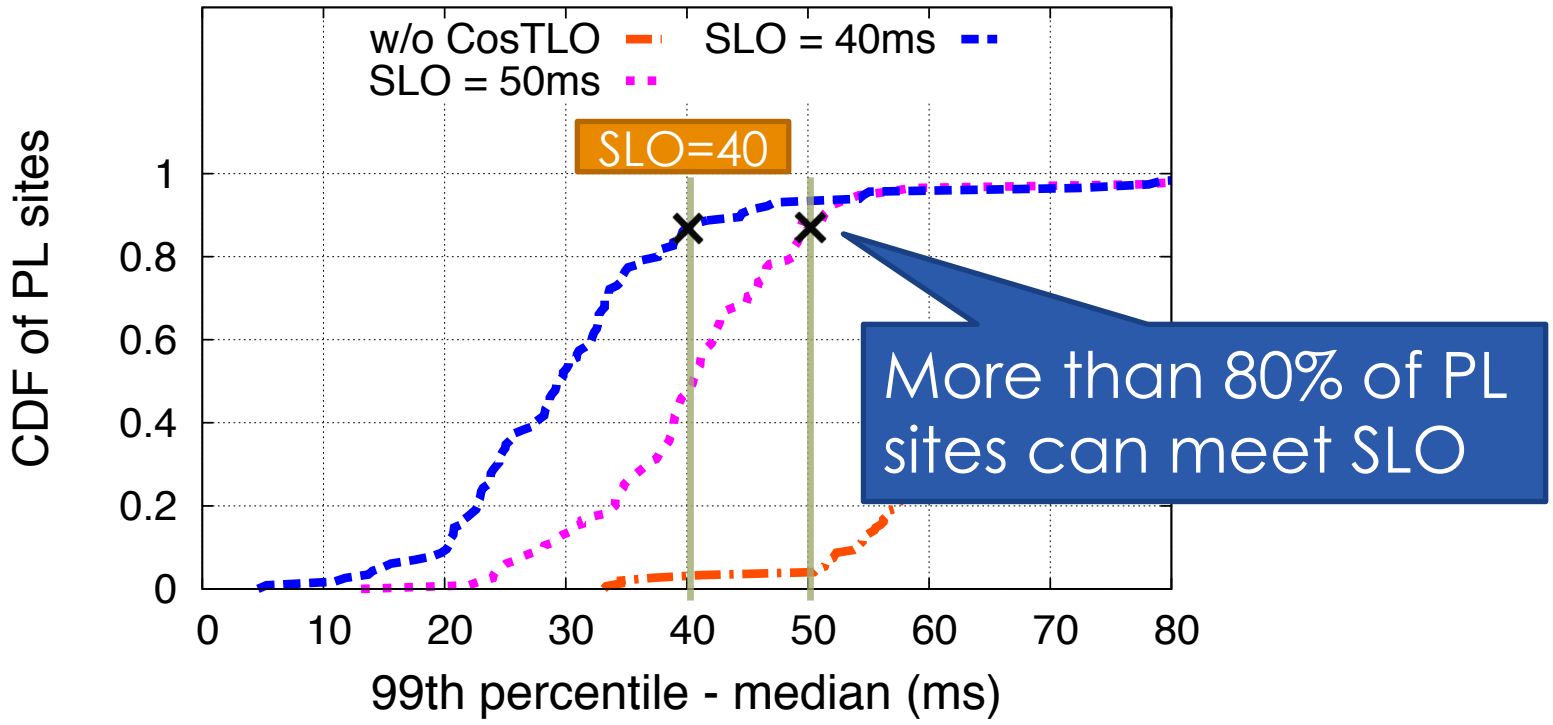
99%ile – median <= X ms



CosTLO Satisfies SLOs

Single request SLO

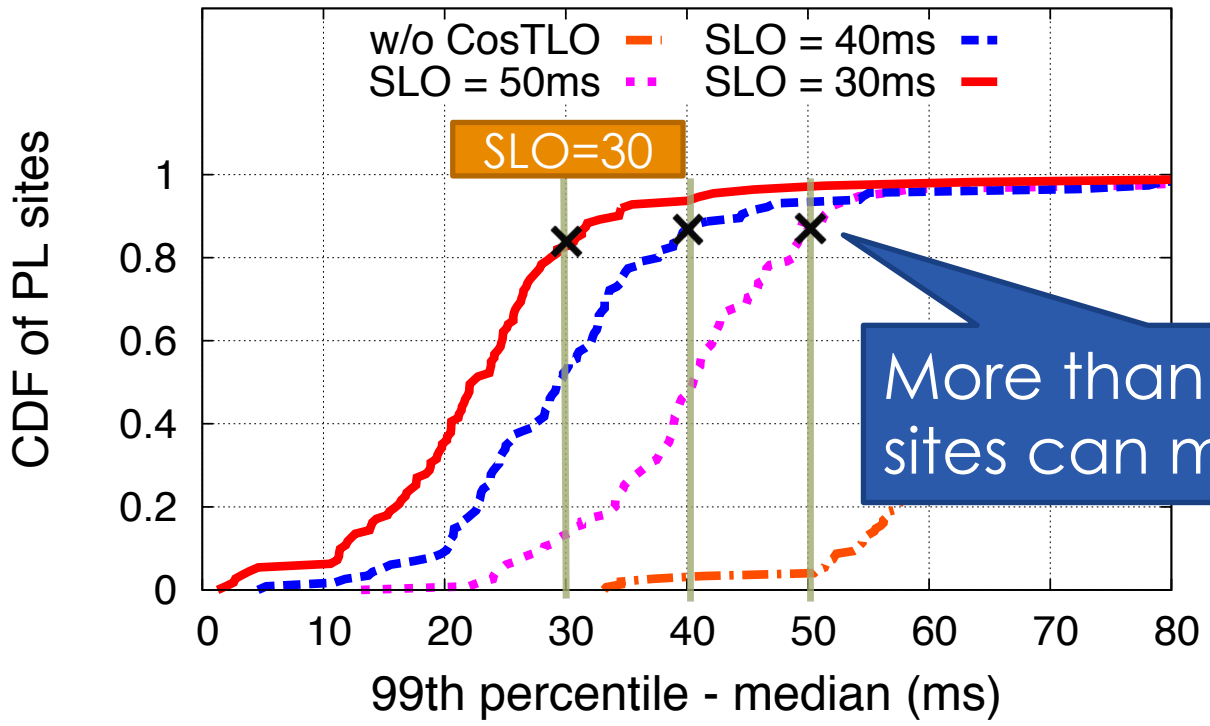
99%ile – median <= X ms



CosTLO Satisfies SLOs

Single request SLO

99%ile – median <= X ms

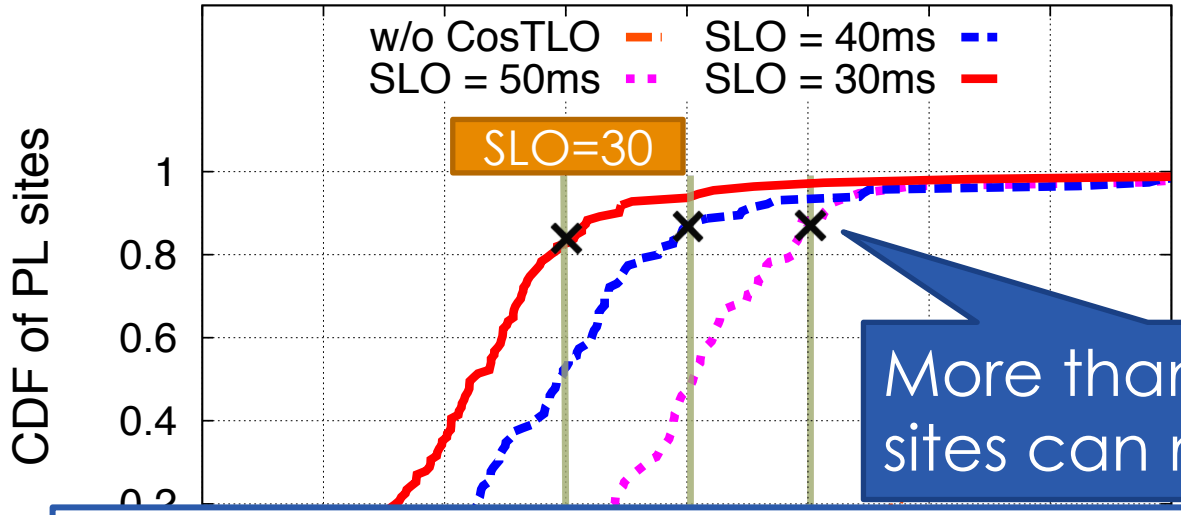


More than 80% of PL sites can meet SLO

CosTLO Satisfies SLOs

Single request SLO

99%ile – median <= X ms



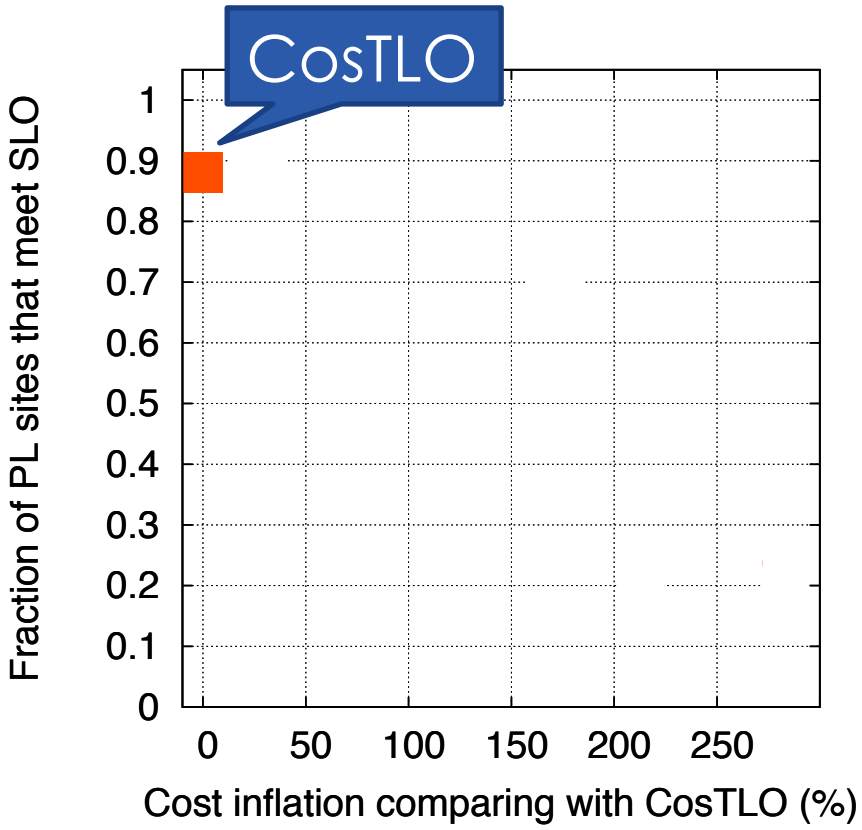
More than 80% of PL sites can meet SLO

Can also satisfy application-specific SLOs

- *Bound median webpage load time*
- *Bound median sync completion time*

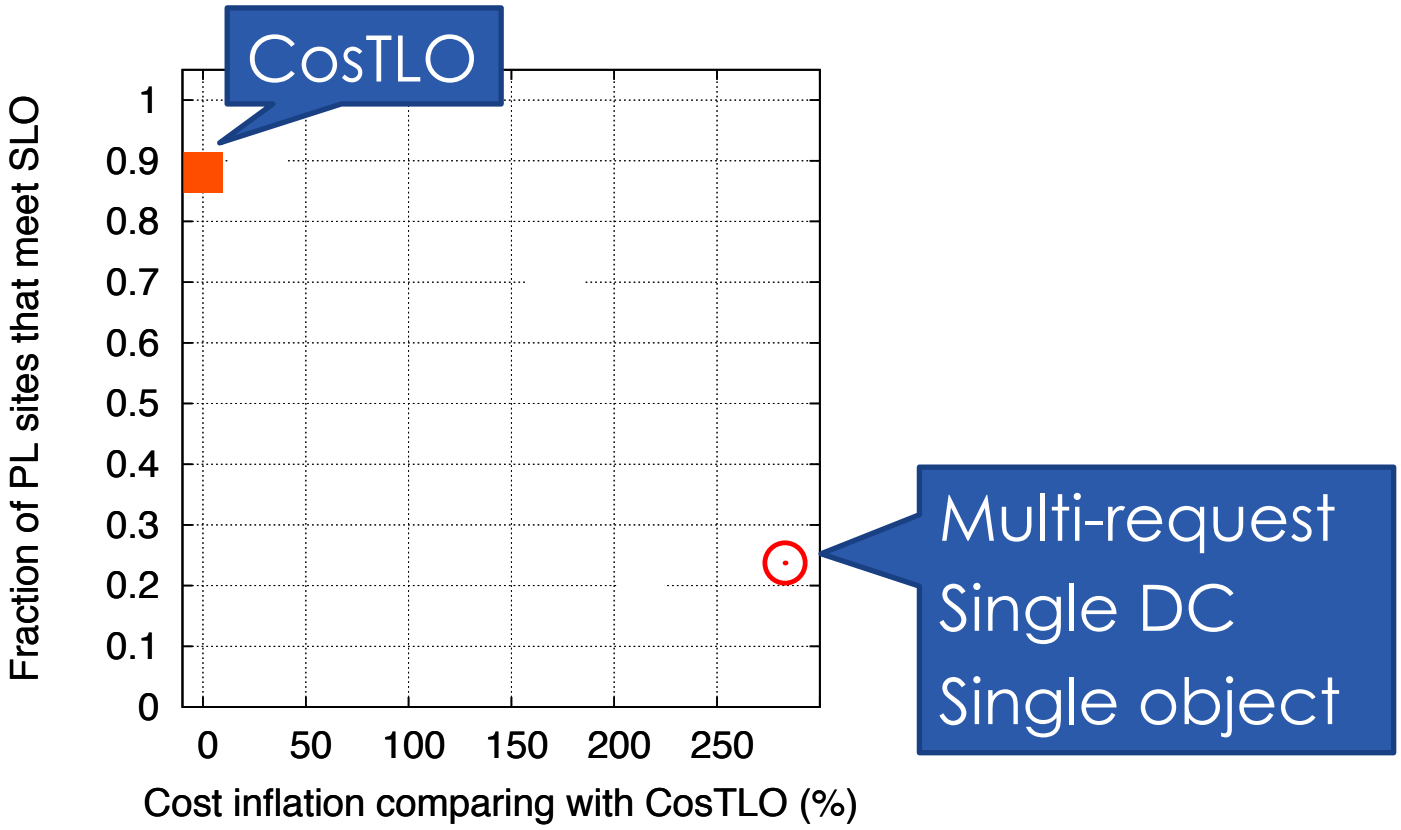
Important to combine forms of redundancy

SLO: GET tail latency inflation < 40ms



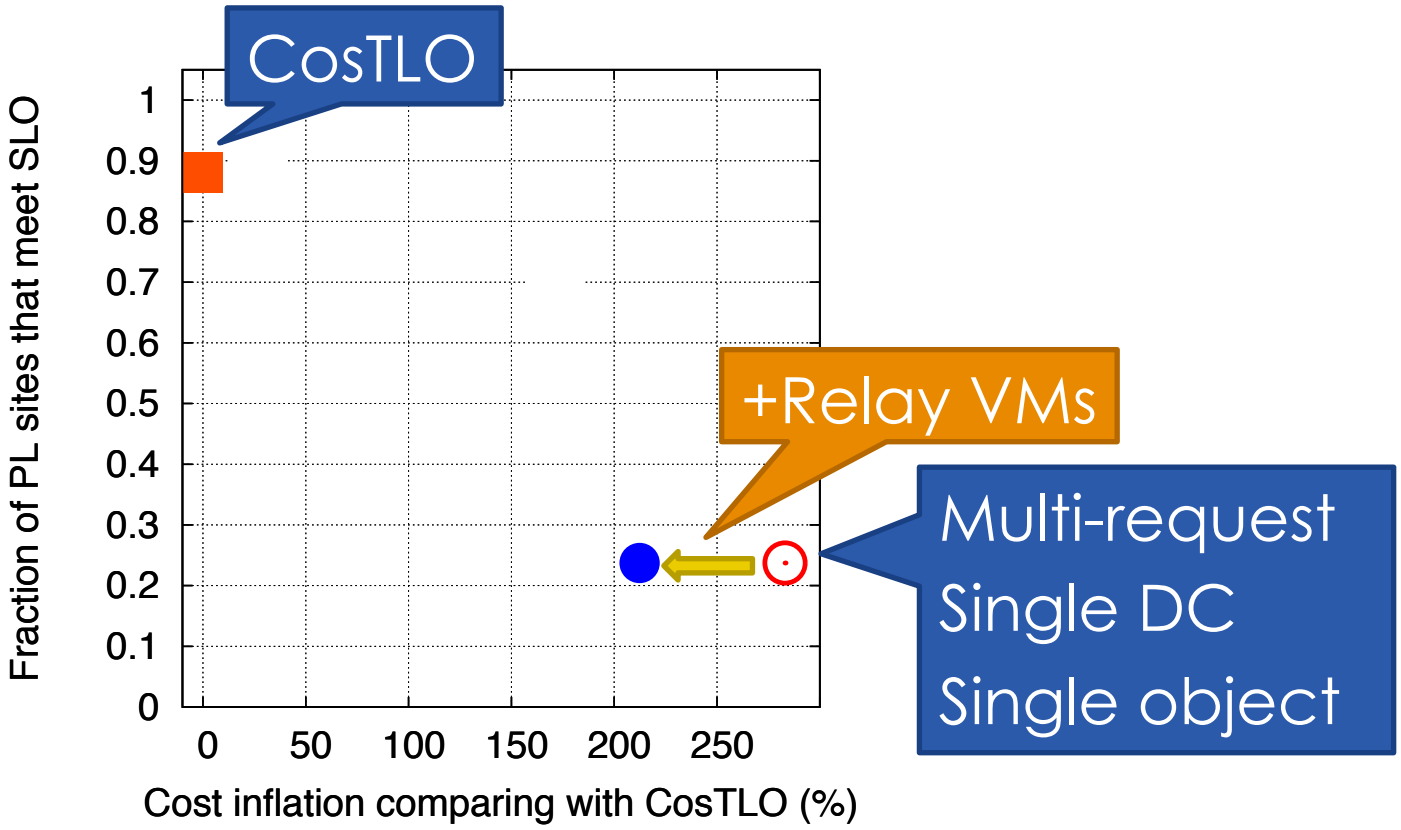
Important to combine forms of redundancy

SLO: GET tail latency inflation < 40ms



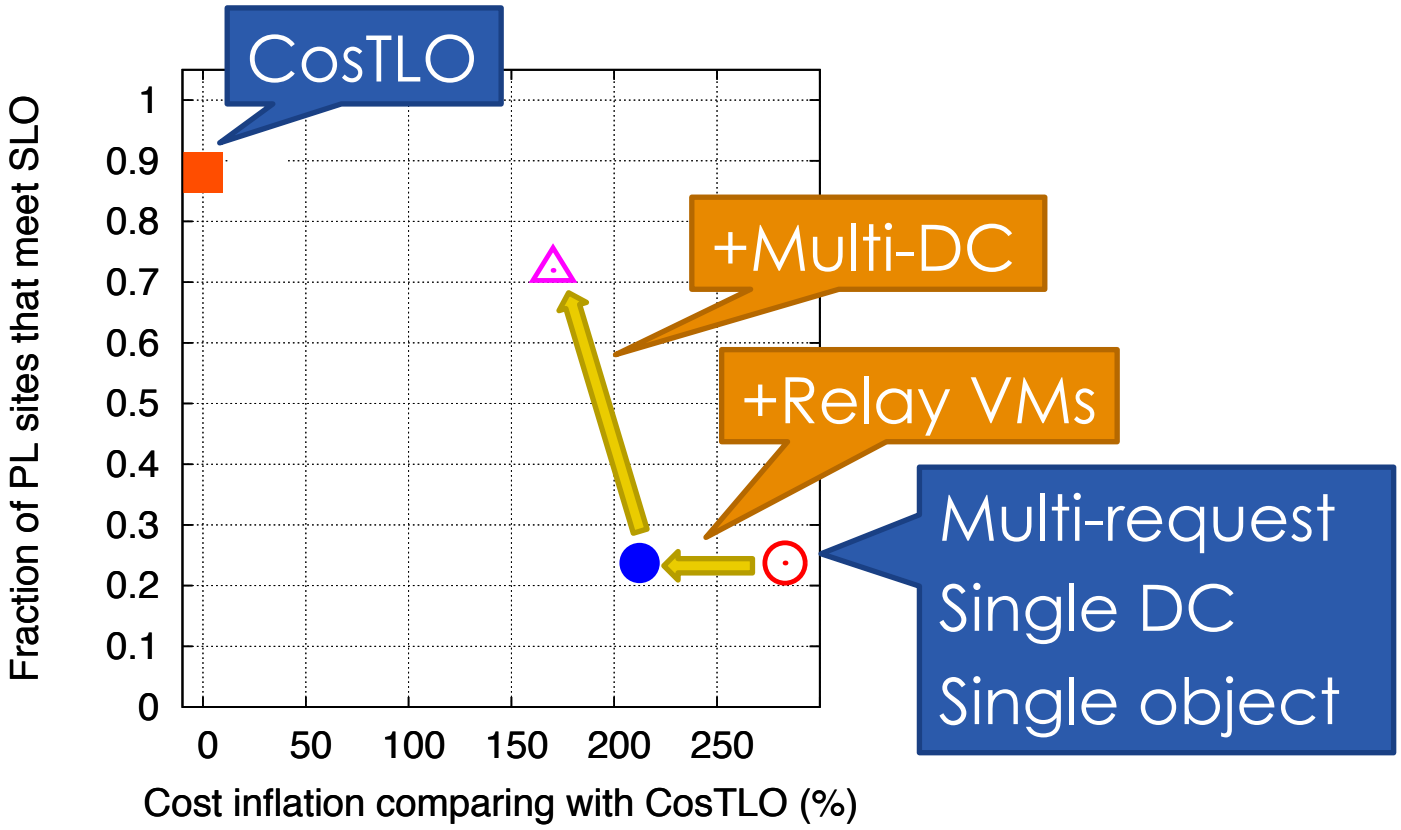
Important to combine forms of redundancy

SLO: GET tail latency inflation < 40ms



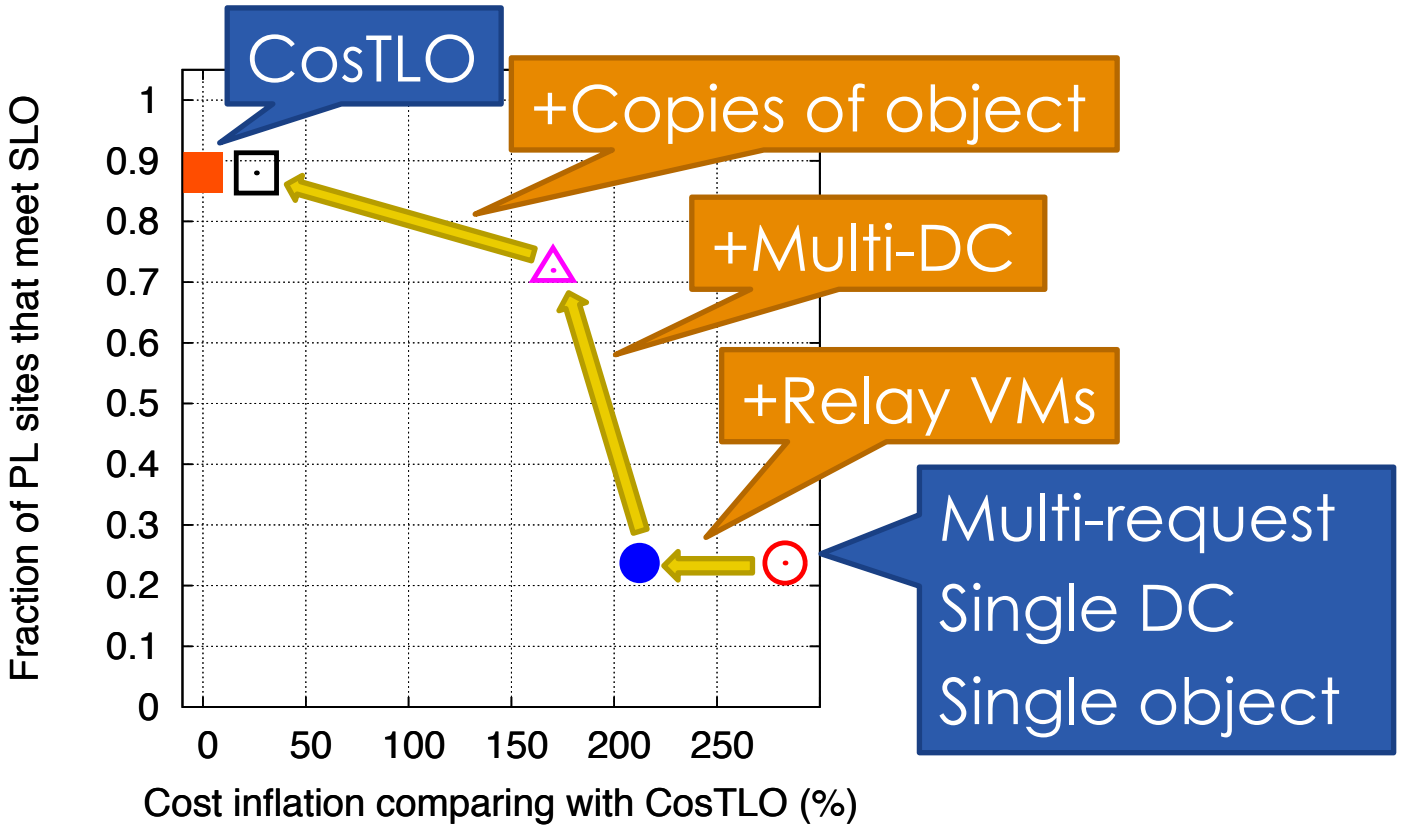
Important to combine forms of redundancy

SLO: GET tail latency inflation < 40ms



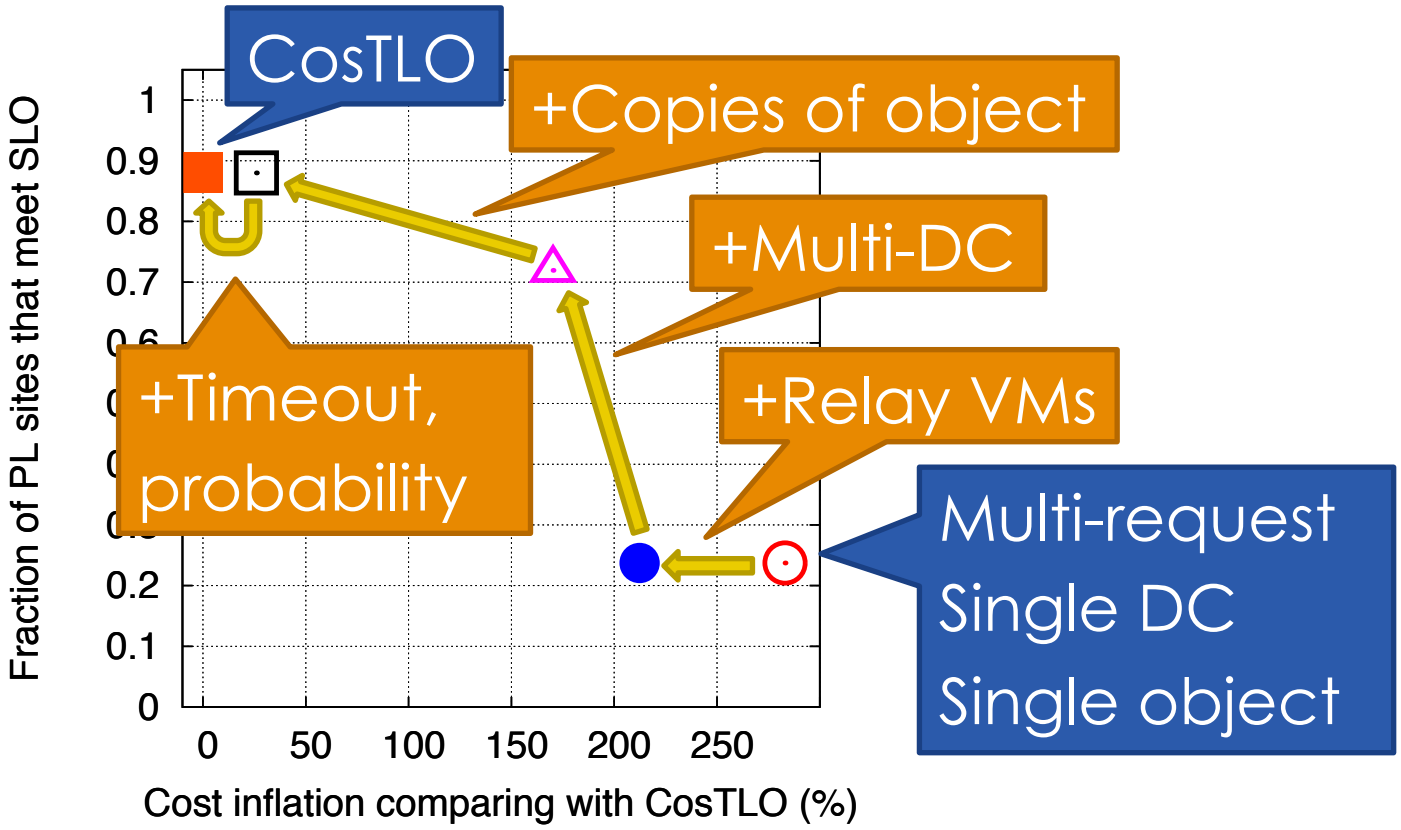
Important to combine forms of redundancy

SLO: GET tail latency inflation < 40ms



Important to combine forms of redundancy

SLO: GET tail latency inflation < 40ms



Conclusions

- Current cloud storage services have high latency variance and unpredictable performance
- CosTLO
  - Reduce tail latency using redundant requests
  - Judiciously combine forms of redundancy
  - Satisfy SLOs with low additional cost

Thank you

http://zwu.me/costlo.html

wuzhe@umich.edu

Questions?