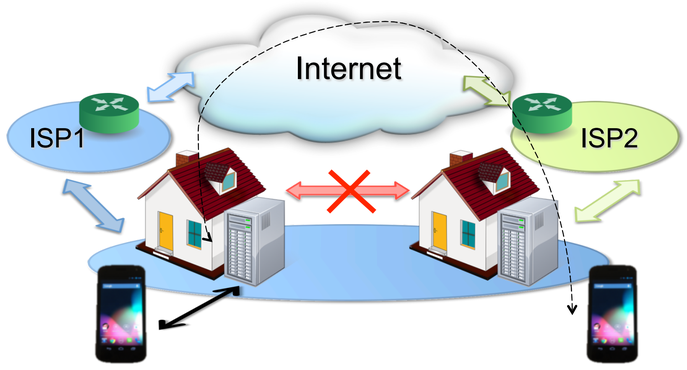
Adaptive VM Handoff Across Cloudlets

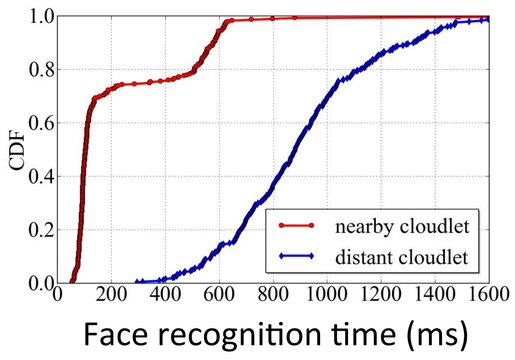
Kiryong Ha, Yoshihisa Abe, Zhuo Chen, Wenlu Hu, Brandon Amos, Padmanabhan Pillai, Mahadev Satyanarayanan
Carnegie Mellon University and †Intel Labs

Cloudlet and User Mobility

- **Cloudlets** bring the Cloud closer to users
  - Second-level data centers dispersed at the edge
  - Low latency, high BW compared to distant Cloud

- Mobility: what if a user moves away from the current cloudlet?
  - As long as network is connected, the application continues to work
  - Interactive response will degrade as the network distance increases





User Mobility can eliminate the benefits of Cloudlets!

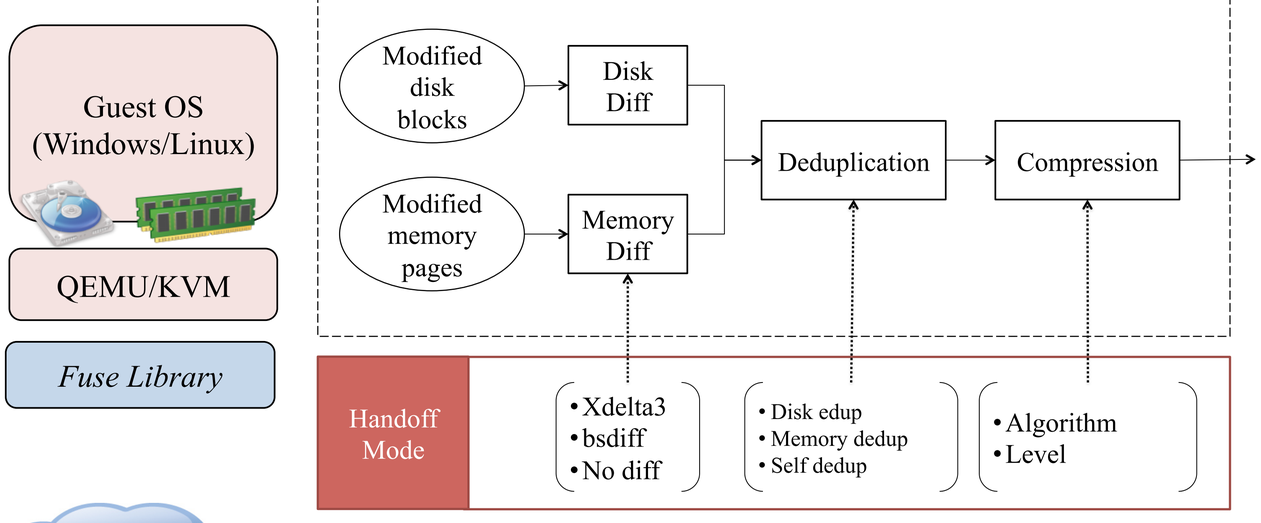
VM Handoff

- Live migration of the backend server across cloudlets
- Maintain network quality by seamlessly migrating the backend
- Different from data center live migration
  1. Optimized for minimizing handoff time (a.k.a migration time)
  2. Migration over WAN
  3. Computation used for handoff can be a bottleneck (cloudlet is much more limited than a cloud datacenter)

Completely different use case from live migration in data centers!

System Overview

- Minimize transfer size: efficiently find/encode modified regions for transfer across slow WAN
- Adaptive system: dynamic tuning of parameters to balance CPU and network transfer times
- Utilize VM overlays, Delta-encoding, Deduplication, Compression



Adaptive System

- Motivation for dynamic adaptation
  1. Unpredictable network (WAN) between cloudlets
  2. Network fluctuation throughput over time
  3. Varying workload (CPU utilization) at the cloudlet

- System bottlenecks: 1) Processing 2) Transfer time
  More compression to reduce migration size → processing bound
  Fast speed to maximize network utilization → transfer bound

$$Thru_{system} = min(Thru_{processing}, Thru_{network})$$

- Estimate system throughput, which is determined by choice of algorithms
  - Idea: two algorithms differ in compressibility, but their relative performance will be similar across workloads
  - Use a profile created using a test workload at offline

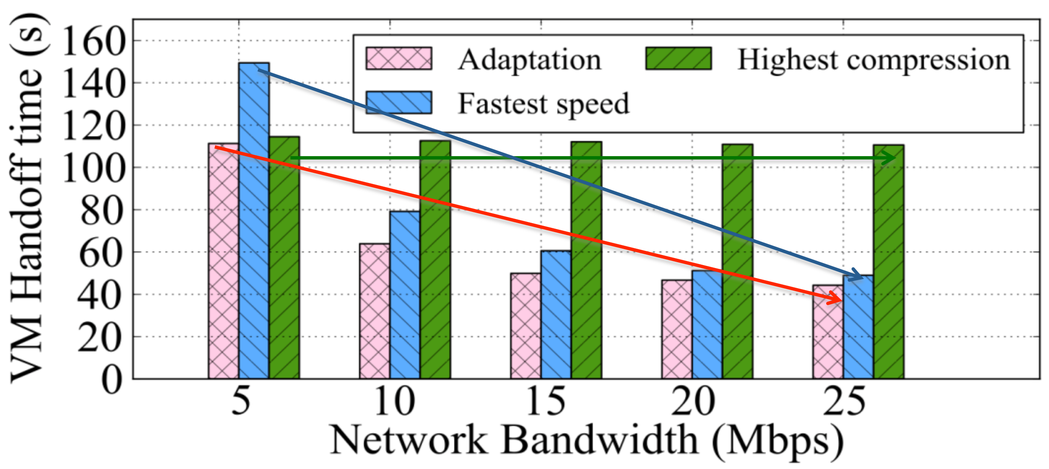
VM Handoff is adaptive to both network BW and computation

Evaluation

- Performance comparison with datacenter live migration (10Mbps network, QEMU/KVM, 8GB disk and 1Gb memory)

Application	Method	Handoff Time	VM downtime
OBJECT (Linux)	VM Handoff	**1 min**	5.50 s
	KVM (no-share)	**127 min**	1.45 s
	KVM (incremental)	**12 min**	1.54 s
MAR (Windows)	VM Handoff	**4.2 min**	12.6 s
	KVM (no-share)	**159 min**	7.44 s
	KVM (incremental)	**52 min**	7.63 s

- Comparison with static operating modes
  - *Fastest speed*: less compute/larger data → Network bound
  - *Highest comp*: small data/more process time → CPU bound



Order of magnitude improvement in migration time!

Intel Science & Technology Center for Cloud Computing

Carnegie Mellon University · Georgia Tech · intel · Princeton University · UC Berkeley · University of Washington