

---

# Scheduling Pitfalls for I/O-intensive Guests

---

Diego Ongaro, Alan L. Cox, and **Scott Rixner**

Rice University

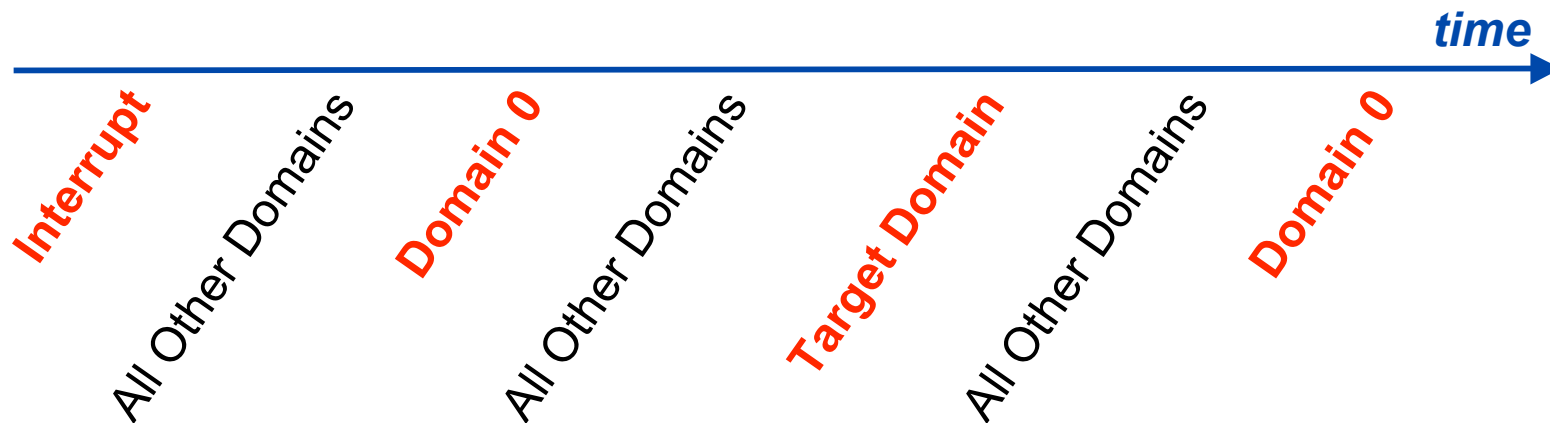
November 15, 2007



# I/O Performance

- Initial observations

- Ping latency to a guest was way too high on a heavily loaded system (>30 seconds!!!)
- Theoretical “worst” case (<1 second):



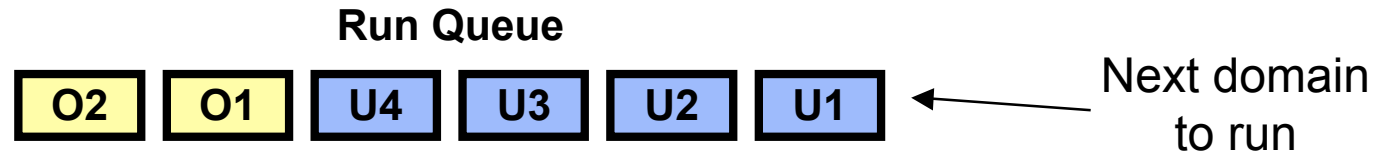
- This led us to explore the impact of Xen’s scheduler on I/O performance

---

# The Credit Scheduler

- Each domain is assigned “credits”
  - Approximates the fraction of processor resources each domain will receive
  - *Does not* indicate *when* a domain will receive that fraction
  
- Scheduler increments/decrements credits
  - Periodically deduct credits from running domain
  - Add credits when majority of credits in the system have been consumed

# Scheduler Operation

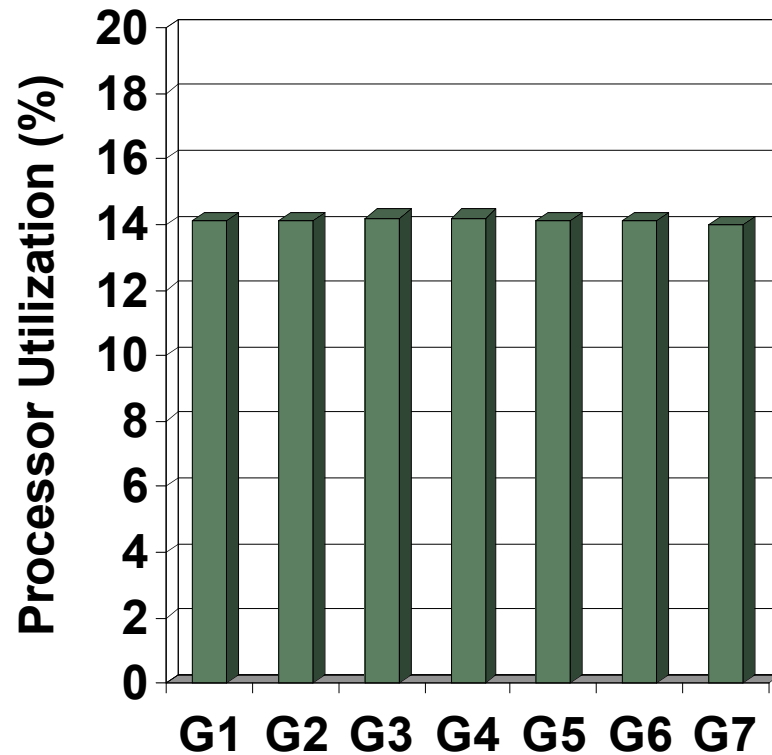


- Domain states
  - Under: domain has credits remaining
  - Over: domain is over its credit allowance
- Domains are run in FIFO order by state
  - “Over” domains only run if no “under” domains
  - Domain may run for up to 30ms if it has enough credits
- After running, return to the run queue by state
  - Behind all other domains in the same state
  - Regardless of runtime or remaining credits

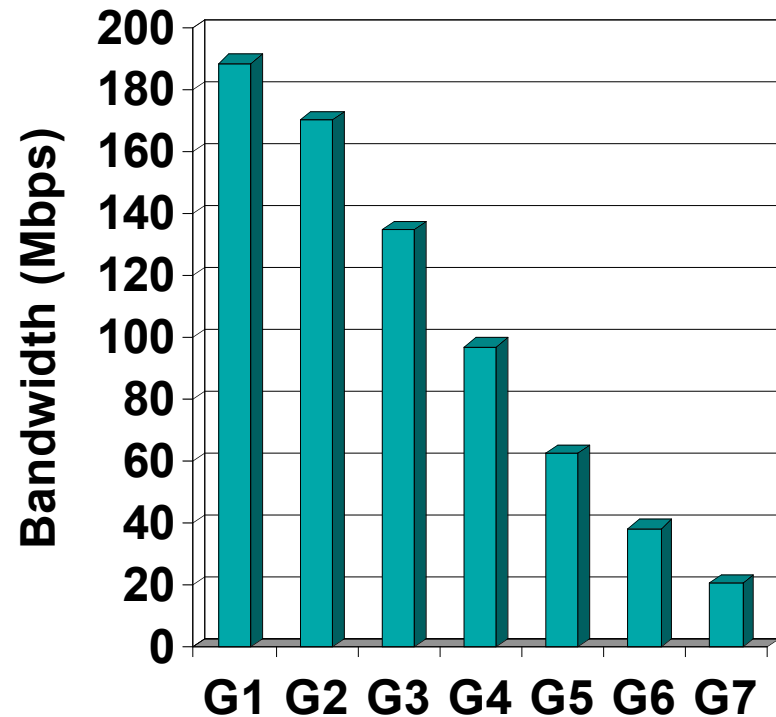
*This approach is biased towards computation*

# Xen 3 Unstable Performance

## Computation Guests



## Streaming Guests



Ping additional idle domain (G8):

Ping latency to G8: 36.6 seconds!

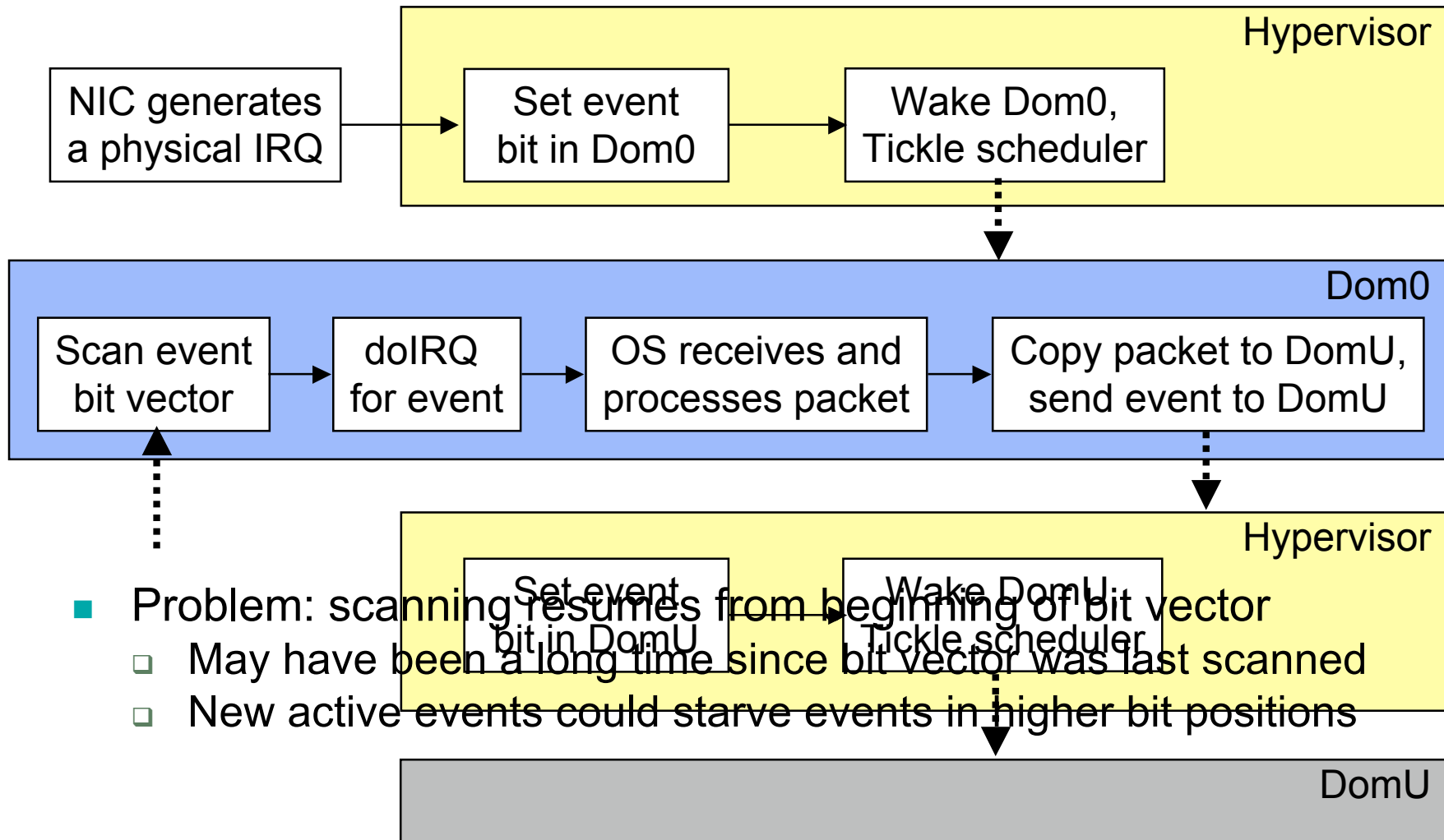
Ping latency to G8: 67.2ms

---

# Event Channel Notification

- Problematic interaction between event channel notification and the scheduler
- Event channels
  - Used for inter-domain communication (physical/virtual interrupts)
- Pending event channel notifications
  - Stored as a bit vector
  - Events are mapped to bits as they are initialized
    - Ordering is basically arbitrary

# Receiving Network Packets



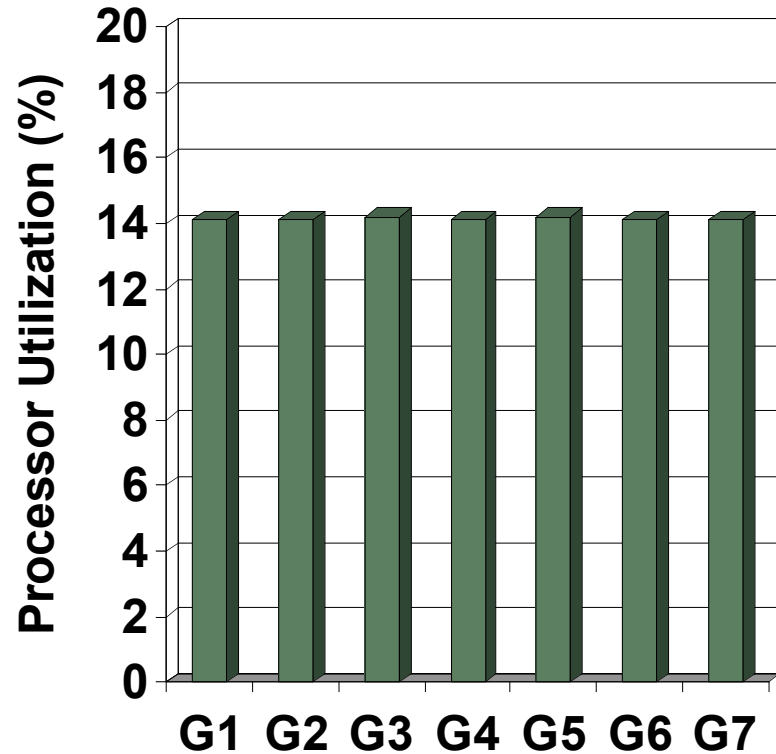
---

# Event Channel Processing Fix

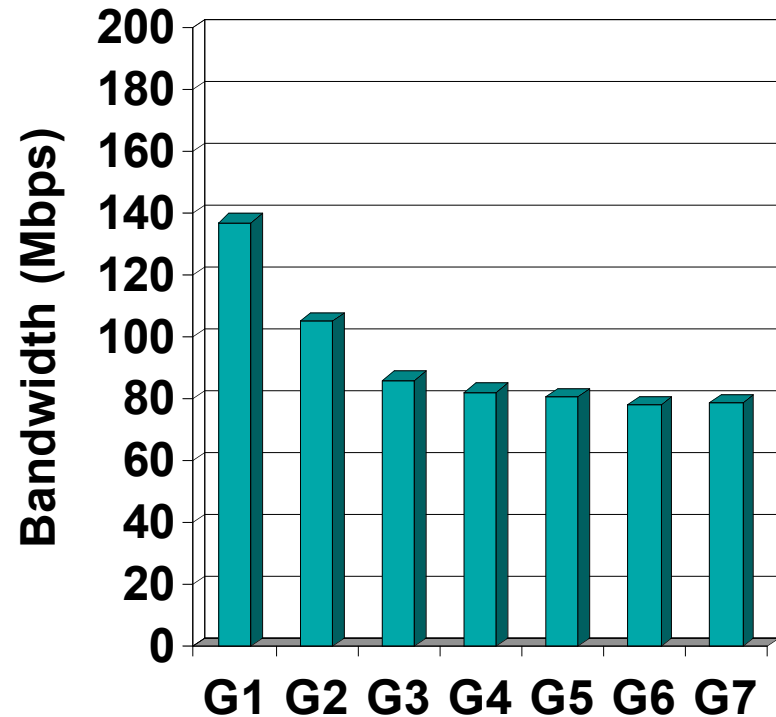
- Priority based on bit order was unintentional
- Process events in strict round-robin order
  - Ensure every event is serviced once before any event is serviced twice
- Worst case returns to “theoretical” worst case
  - Unless domain 0 or the target domain are out of credits...

# Performance with Event Channel Fix

## Computation Guests



## Streaming Guests



Ping additional idle domain (G8):

Ping latency to G8: 259.2ms

Ping latency to G8: 6.6ms

---

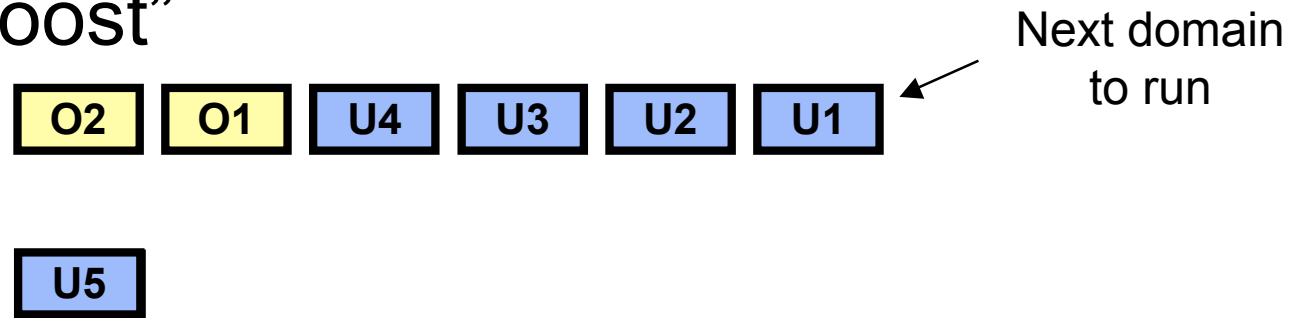
# Scheduler “Optimizations” for I/O

- Boosting domains
  - Increasing priority of I/O domains
- Tickling the scheduler
  - Invoking the scheduler during event channel notifications

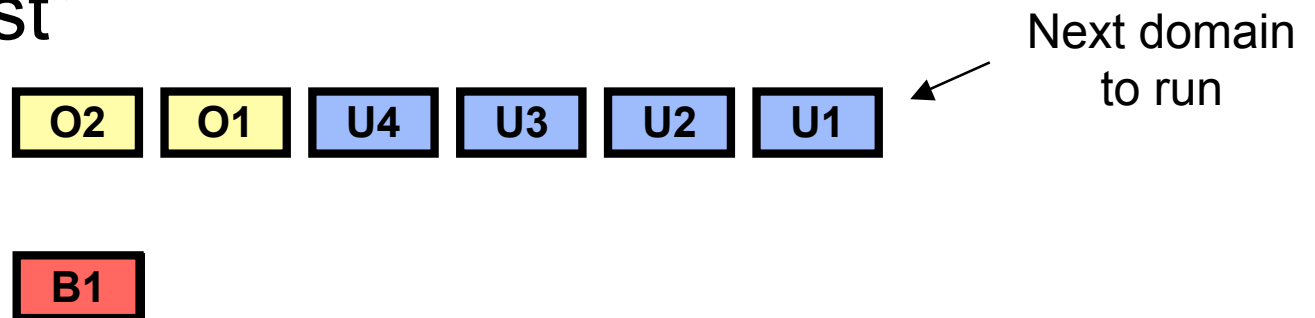
# Boosting Idle Domains

- Additional scheduling state (“boost”)
  - Higher priority than “under”
  - Used when idle domains are targets of events

- Without “boost”



- With “boost”



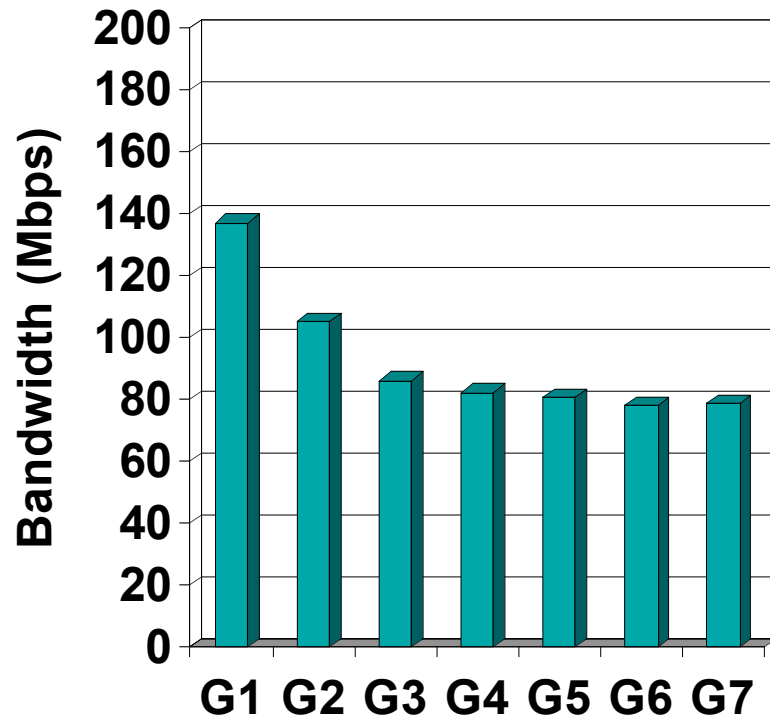
---

# Tickling the Scheduler

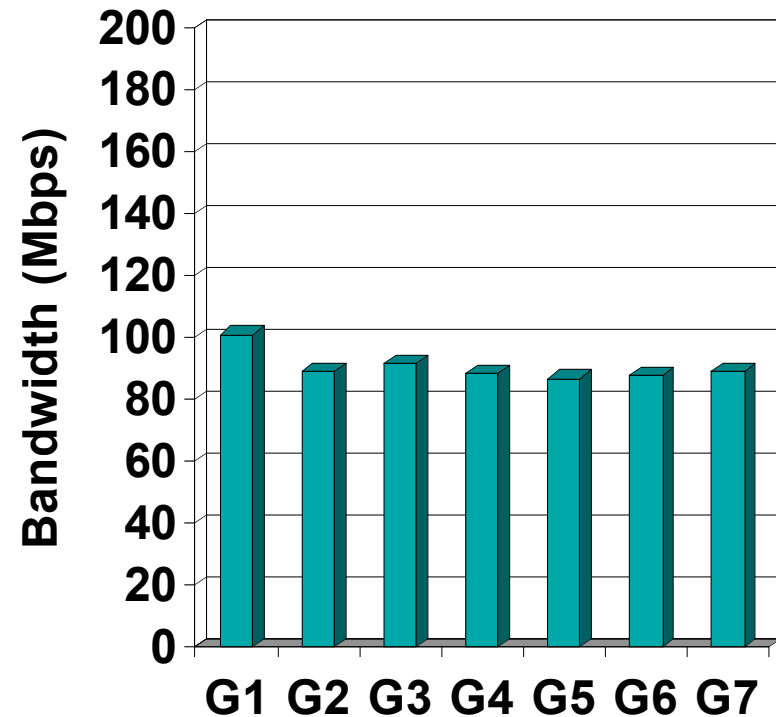
- Tickling the scheduler on every event channel notification may be a mistake
  - Scheduler only knows about the one domain that received an event
- Waiting until *all* events have been processed allows the scheduler to decide which domain has highest priority and should run next

# Impact of Tickling

## Boost/Tickle



## Boost/No Tickle



Ping additional idle domain (G8):

Ping latency to G8: 6.6ms

Ping latency to G8: 5.1ms

---

# Summary

- Scheduling has a large impact on I/O fairness
  - Can cause exorbitant latency
  - Can cause bandwidth inequity
- Credit scheduler is not necessarily fair for I/O
  - Does well with compute domains
  - No notion of timeliness, which is needed for I/O
- We are exploring other scheduler behaviors and optimizations for I/O
  - Paper to appear in VEE 2008