Bobtail: Avoiding Long Tails in the Cloud

Yunjing Xu, Zachary Musgrave, Brian Noble, Michael Bailey
University of Michigan
Public clouds are used everywhere

Amazon’s Cloud

1/3 of daily users
One third of all Internet users will access an Amazon AWS cloud site on average at least once a day.

1% of Internet traffic
One percent of all Internet consumer traffic on average is coming or going to Amazon managed infrastructure.

4th largest CDN
Amazon’s growing CloudFront and S3 traffic volumes recently made it the fourth largest CDN after Akamai, Limelight and Level3.

Craig Labovitz, Deep Field, April 2012
Challenges to large applications

• User-facing applications have tight deadlines
  – E.g., 300ms for end-to-end response time

• Large applications have more moving parts
  – A single Amazon page requires over 150 services
  – 10ms budget for individual services

• The tail latency matters, not just the average
  – Easy to miss deadline if any service becomes slow
  – More critical as you scale out further
Agenda

• EC2 has long tail latency problem
  – Much worse than that in dedicated data centers
  – A problem of the node not the network

• Root cause: incompatible sharing
  – Co-scheduling of I/O-bound and CPU-bound VMs

• Solution: Bobtail
  – Pick VMs without long tails
  – A customer-centric approach
Agenda

• EC2 has long tail latency problem
  – Much worse than that in dedicated data centers
  – A problem of the node not the network

• Root cause: incompatible sharing
  – Co-scheduling of I/O-bound and CPU-bound VMs

• Solution: Bobtail
  – Pick VMs without long tails
  – A customer-centric approach
Measurements Setup

• Live measurement in EC2’s east region
  • Use Thrift RPC framework to measure RTT
• EXP1: RTT for 60 RPC servers
  – Measure from designated testing nodes
  – Give an overview of the problem
• EXP2: Pair-wise RTT for 16 RPC servers
  – Measure between each other
  – Study spatial properties
How bad is the latency tail?

99.9th percentile: ~30ms

CDF of RTT vs. RTT (ms)
How bad is the latency tail?

- Median: ~0.6ms
- 99.9th: ~30ms
- Over 10X larger
How does it compare to dedicated DCs?

Over 2 times larger

Measured in dedicated DC[1,29]
Spatial Properties

A property of nodes not the network

Depends only on servers

99.9th (ms)
Agenda

• EC2 has long tail latency problem
  – Worse than that in dedicated data centers
  – A problem of the node not the network

• Root cause: incompatible sharing
  – Co-scheduling of I/O-bound and CPU-bound VMs

• Solution: Bobtail
  – Pick VMs without long tails
  – A customer-centric approach
Root cause analysis

• Virtualization and sharing cause high variance
  – Wang et al., INFOCOM’10
• Does the type of sharing matter? Why?
• Controlled experiments on testbed: Xen 4.1
  – Five VMs share two CPU cores: 40% each
• Vary workload combination
  – CPU-bound workload: ~85% busy
  – I/O-bound workload: same RPC framework
5 I/O-bound VMs on 2 CPU cores
4 I/O-bound VMs on 2 CPU cores
3 I/O-bound VMs on 2 CPU cores
2 I/O-bound VMs on 2 CPU cores
1 I/O-bound VMs on 2 CPU cores
The type of sharing matters

• Incompatible sharing causes the long tail
  – CPU-bound VMs vs. I/O-bound VMs

• Sharing is not always bad
  – Only when # of CPU-bound VMs >= # of CPU cores

• Sharing with I/O-bound VMs is okay
  – Even when there is one CPU-bound VM mixed
Why the type of sharing matters

• What causes large latency
  – CPU-bound VMs occupy CPUs when requests arrive
  – Interrupt processing for I/O-bound VMs is delayed

• Why 30ms for the 99.9\textsuperscript{th} percentile
  – Default time slice
  – Could be reduced at the cost of CPU throughput

• Why the tail only depends on servers
  – Servers may need scheduling when a request comes
Agenda

• EC2 has long tail latency problem
  – Much worse than that in dedicated data centers
  – A problem of the node not the network

• Root cause: incompatible sharing
  – Co-scheduling of I/O-bound and CPU-bound VMs

• Solution: Bobtail
  – Pick VMs without long tails
  – A customer-centric approach
Deal with it

• Key insight
  – CPU-bound VMs delay interrupt processing
  – Any interrupt suffers; not just network I/O
  – Does testing locally using time interrupt

• Bobtail: how it works
  – Launch $N*K$ instances if $N$ needed – one time cost
  – Sleep 0.1ms to check if overslept (e.g., >10ms)
  – Pick the ones that rarely oversleep – kill others
Evaluation

- Bobtail in EC2
  - Using two availability zones in the east region
- Use 40 small instances
  - Bobtail picks 40 out of 160 candidates
  - Baseline: 40 launched directly by EC2
- Partition-Aggregation model [Zats et al., SIGCOMM’12]
  - Call all servers in parallel and wait for the last to respond
  - The slowest determines the transaction response time
- Other benchmarks have better results
  - Refer to the paper
Partition-Aggregation Workflows

10 servers

20 servers

40 servers
Partition-Aggregation Workflows

10 servers

20 servers

40 servers
Discussion

• What if everyone uses Bobtail
  – Not exclusively looking for empty space
  – Seek to share with compatible workloads
  – Emerging partitions – good for everyone

• Amazon could fix the problem
  – Reduce VM sharing – lower VM density
  – Sacrifice CPU efficiency – inherent trade-off
  – Smarter scheduling – do you trust users
Conclusion

• Long tail latency problems EC2
  – Large-scale measurement in real cloud
  – Controlled experiments on local testbed

• Root cause of the long tail problem
  – Co-scheduling of incompatible workloads
  – CPU-bound vs. I/O-bound

• Bobtail: avoid VMs with long tails
  – A scalable approach
  – A user-centric approach without providers’ help
Acknowledgements

The Students and Collaborators from the Networking and Security Research Group:

And of course our research sponsors and collaborators: