MySQL Conference & Expo

Presented by,
MySQL & O'Reilly Media, Inc.

Exploring Amazon EC2 for Scale-out Applications

Morgan Tocker, MySQL Canada

Carl Mercier, Defensio

Introduction

- Defensio is a spam filtering web service for blogs and other social web applications.
- Powered exclusively by Amazon EC2.
- Ruby, Rails, C, MySQL 5.0 (and a few more things).

EC2: Elastic Compute Cloud

- Virtual machines running on Xen.

  These VMs are called “instances”.

- Pay only for what you use.

- On demand scaling - controlled with an API.

- Instances are “disposable”.

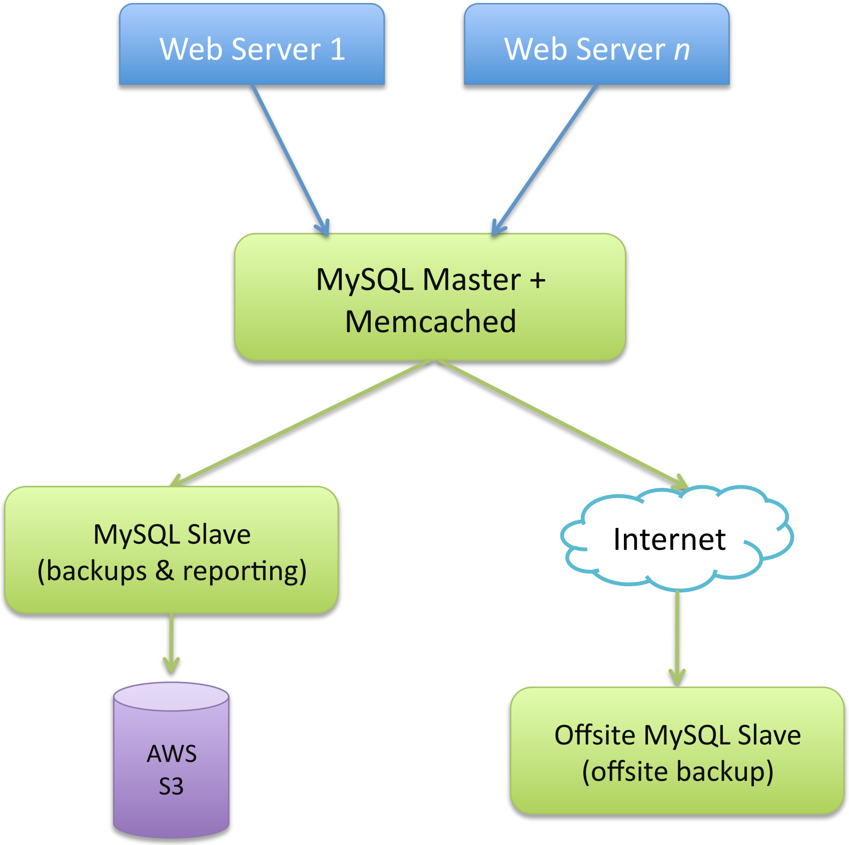
Instance Types

	RAM	CPU	STORAGE	IO	$
SMALL (1)	1.7 GB	1 virtual core 1 CU (32 bit)	160 GB	“moderate”	$0.10 / hour (~ $72 / mo)
LARGE (4) 64 bit	7.5 GB	2 virtual cores 2 CU each (64 bit)	850 GB (2 x 420 GB)	“high”	$0.40 / hour (~ $288 / mo)
XLARGE (8) 64 bit	15 GB	4 virtual cores 2 CU each (64 bit)	1.7 TB (4 x 420 GB)	“high”	$0.80 / hour (~ $576 / mo)

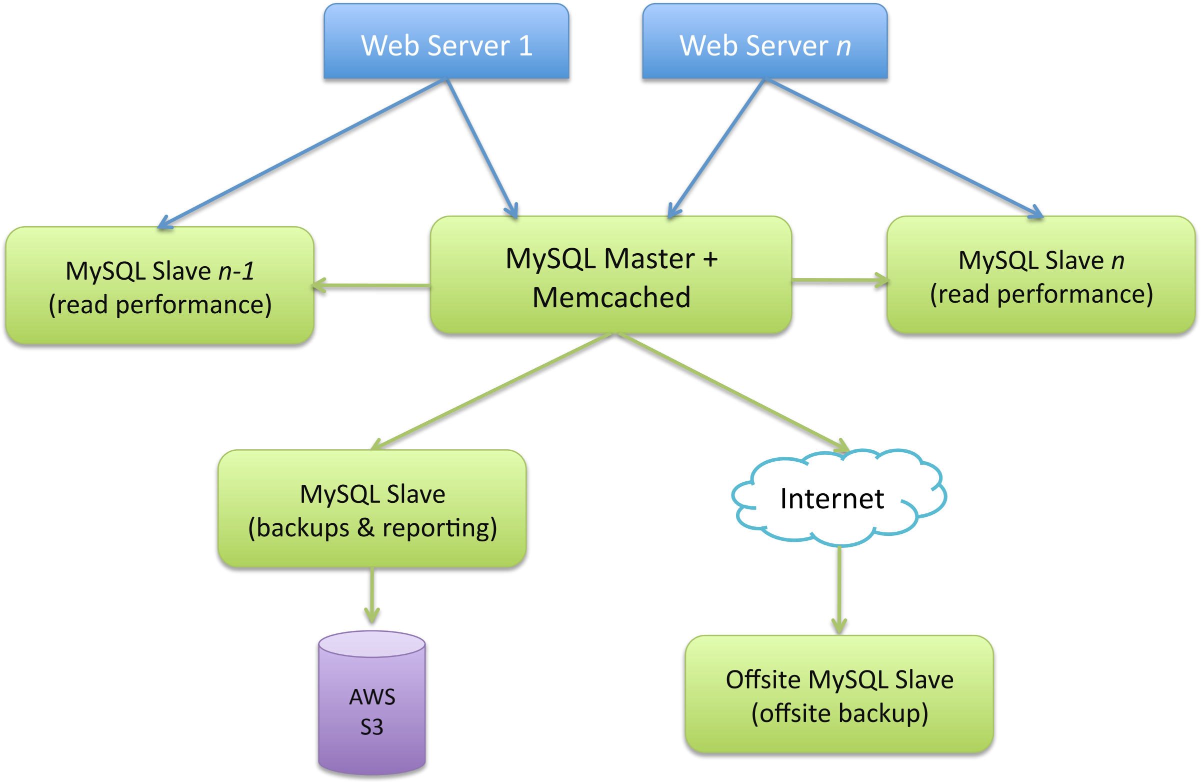
* One EC2 Compute Unit provides the equivalent CPU capacity of a 1.0-1.2 GHz 2007 Opteron or 2007 Xeon processor. This is also the equivalent to an early-2006 1.7 GHz Xeon.

Topologies

In the beginning....


Web Server 1
Web Server *n*
MySQL Master + Memcached
MySQL Slave (backups & reporting)
Internet
AWS S3
Offsite MySQL Slave (offsite backup)

Adding some room to grow…



Web Server 1

Web Server *n*

MySQL Slave *n-1*
(read performance)

MySQL Master +
Memcached

MySQL Slave *n*
(read performance)

MySQL Slave
(backups & reporting)

Internet

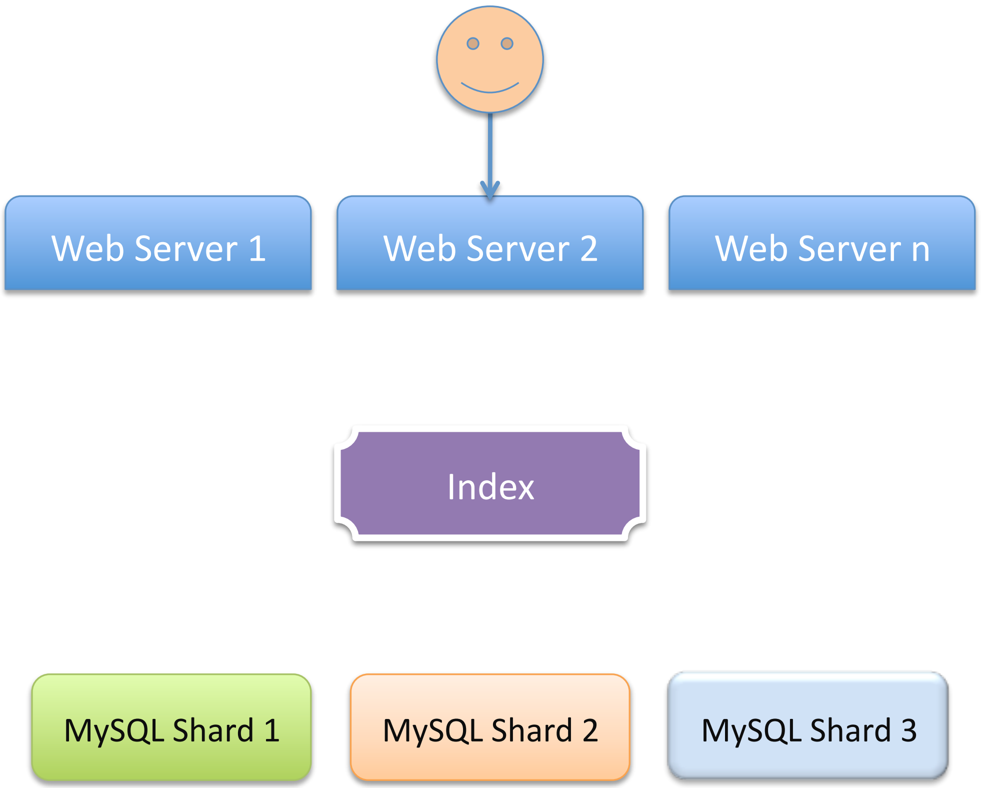
AWS
S3

Offsite MySQL Slave
(offsite backup)

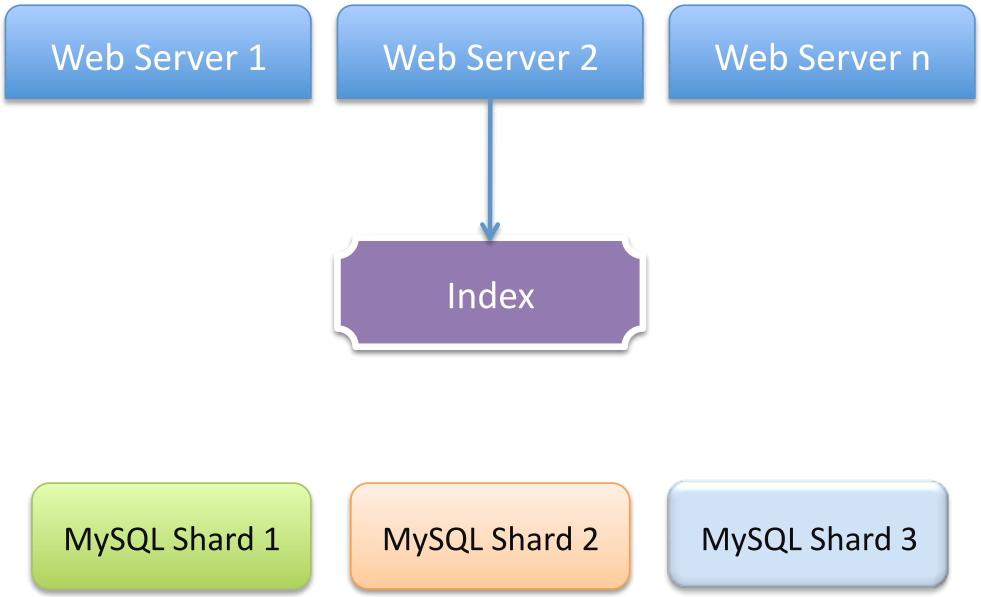
Sharding

- The Defensio application "shards" quite well.
- Expecting to be able to split up customers across many servers.
- Customers are not created equally. Some servers may only have a few with heavy load, others will have thousands.
- Will need to write scripts to "rebalance" the customers - EC2 makes this easy.

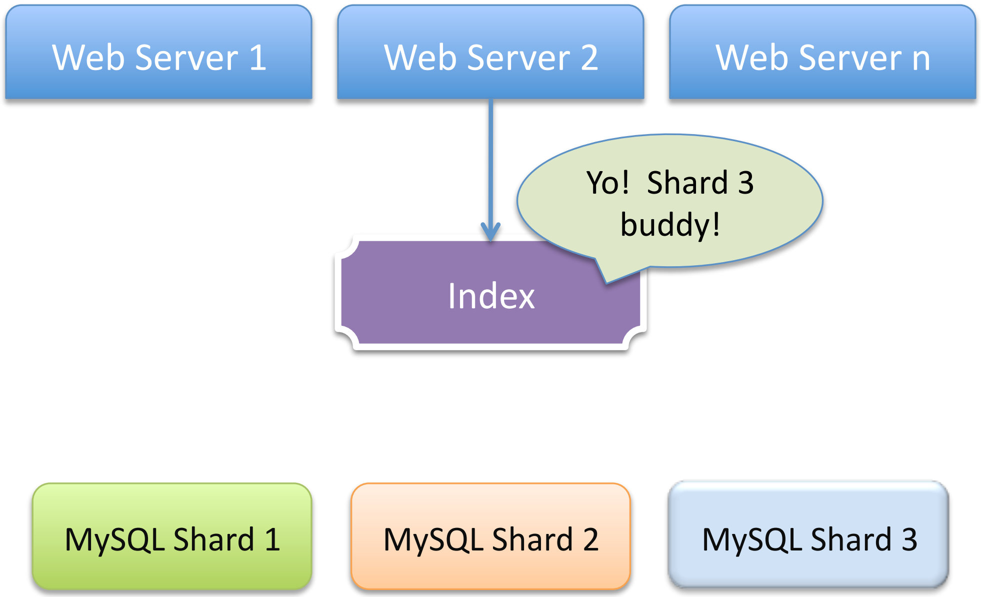
Stepping it up a notch w/ Sharding



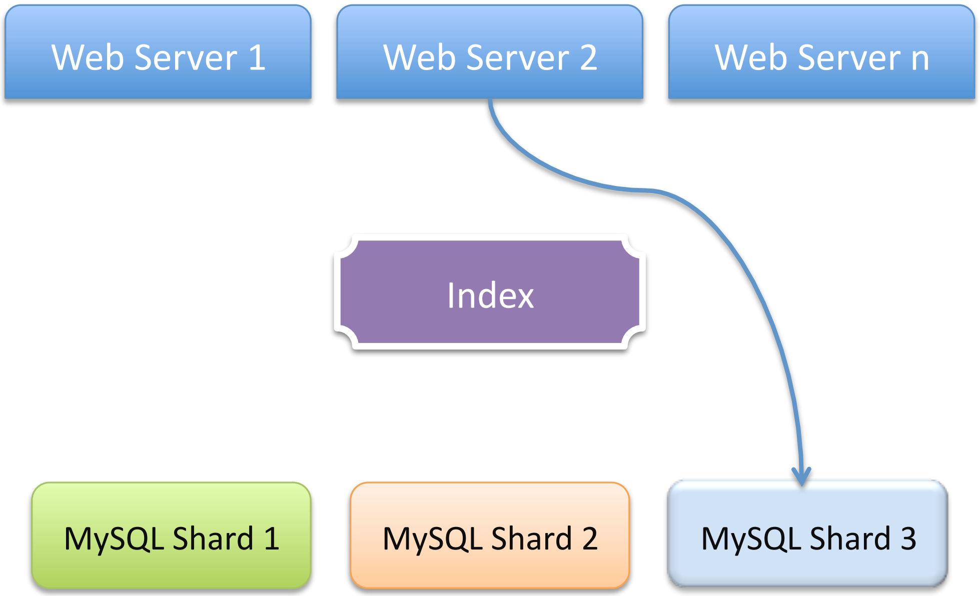
Stepping it up a notch w/ Sharding



Stepping it up a notch w/ Sharding



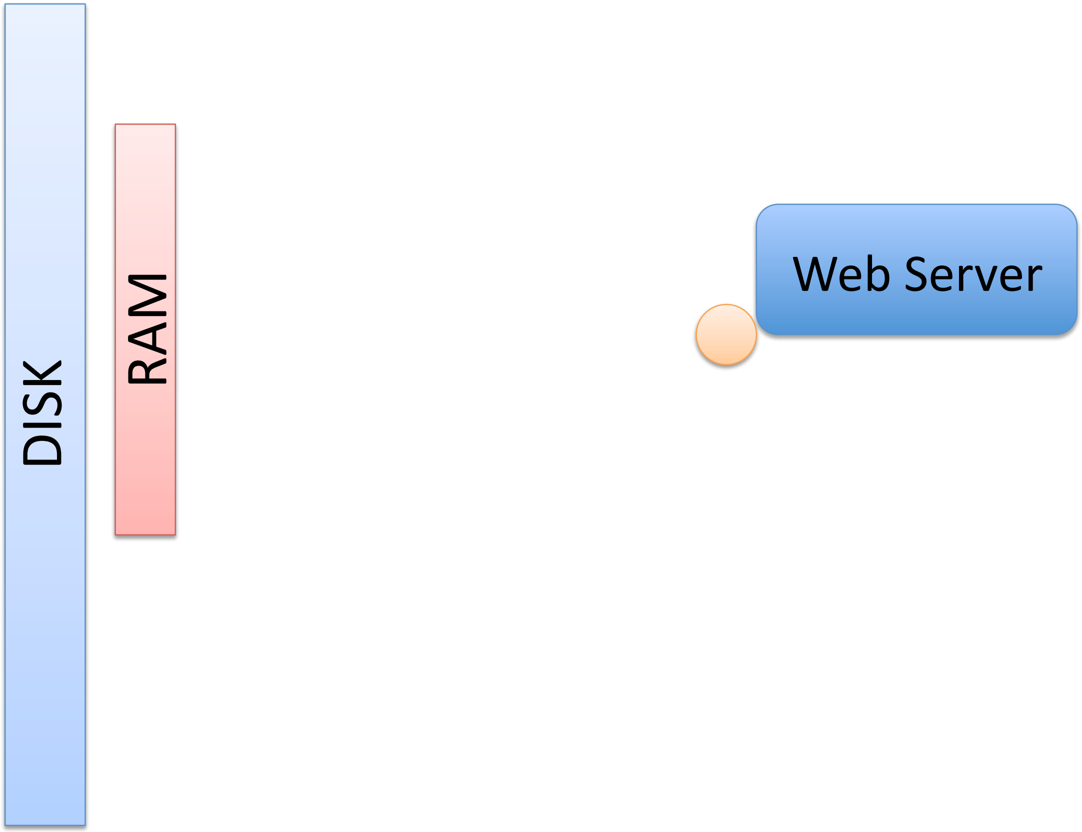
Stepping it up a notch w/ Sharding

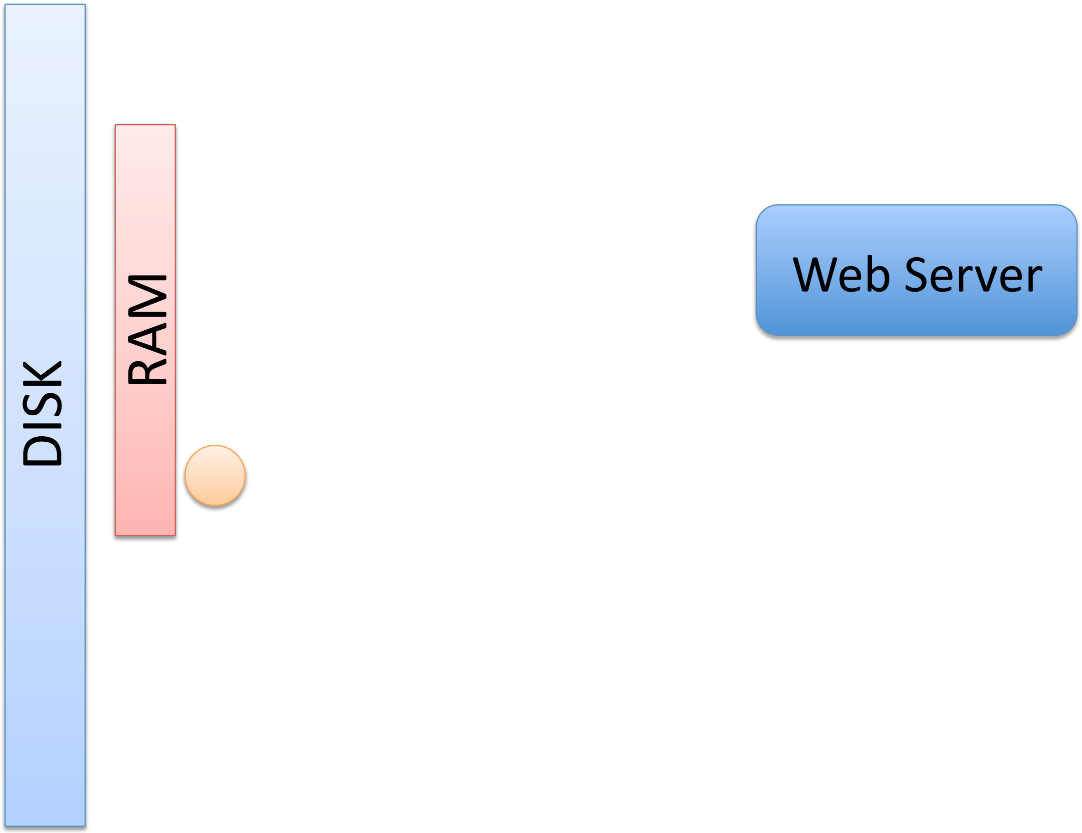


Performance

EC2 Performance

- Databases are bound by disk performance.
- Initial suspicions of EC2 under performing were confirmed.
- It wasn’t that straight forward though.


DISK
RAM
Web Server


DISK
RAM
Web Server

The most basic test

```
$ cd /mnt
$ dd if=/dev/zero of=my50Gfile bs=1024M count=50
```

- A single consumer drive should offer at least 50M/s.
- We’re just writing 50G of nothing.
- It tests sequential I/O, with a small filesystem overhead.

The most basic test (cont.)

```
$ time dd if=/dev/zero of=my50Gfile bs=1024M count=50

50+0 records in
50+0 records out
53687091200 bytes (54 GB) copied, 2309.87 seconds, 23.2 MB/s
real    38m30.377s
user    0m0.000s
sys     0m57.560s
```

Damn, that sucks.

...what’s one of the first rules of benchmarking?

```
$ time dd if=/dev/zero of=my50Gfile bs=1024M count=50

50+0 records in
50+0 records out
53687091200 bytes (54 GB) copied, 504.717 seconds, 106 MB/s

real    8m24.982s
user    0m0.000s
sys     1m24.790s
```

Explanation

- There’s a first write penalty for EC2.
- It is a limitation in EC2s architecture - all subsequent writes are **much** faster.
- There’s no documentation on this **anywhere**.
- Deletes are also slow.

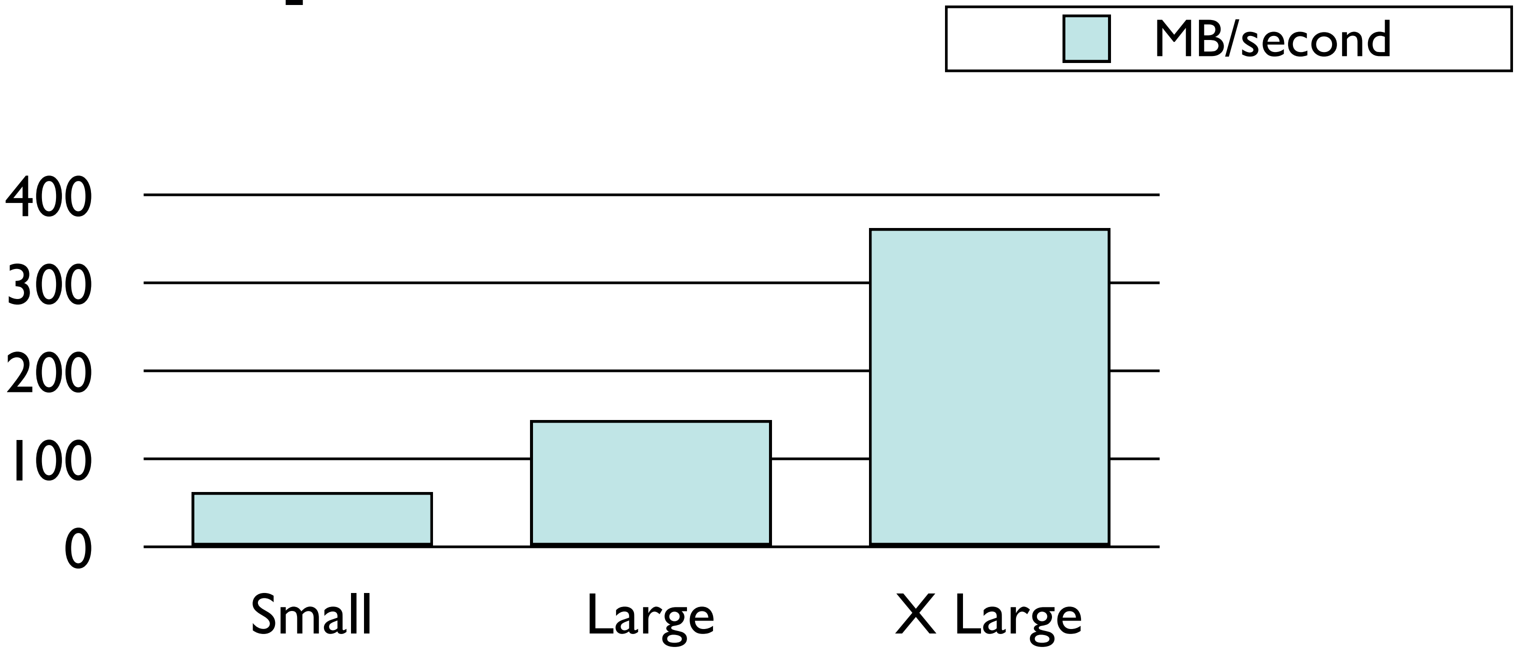
Workaround

```
$ dd if=/dev/zero of=diskfiller.tmpfile bs=1000M count=99999999
```

- This takes just over 5 hours for a 400G stripe on 2 drives.

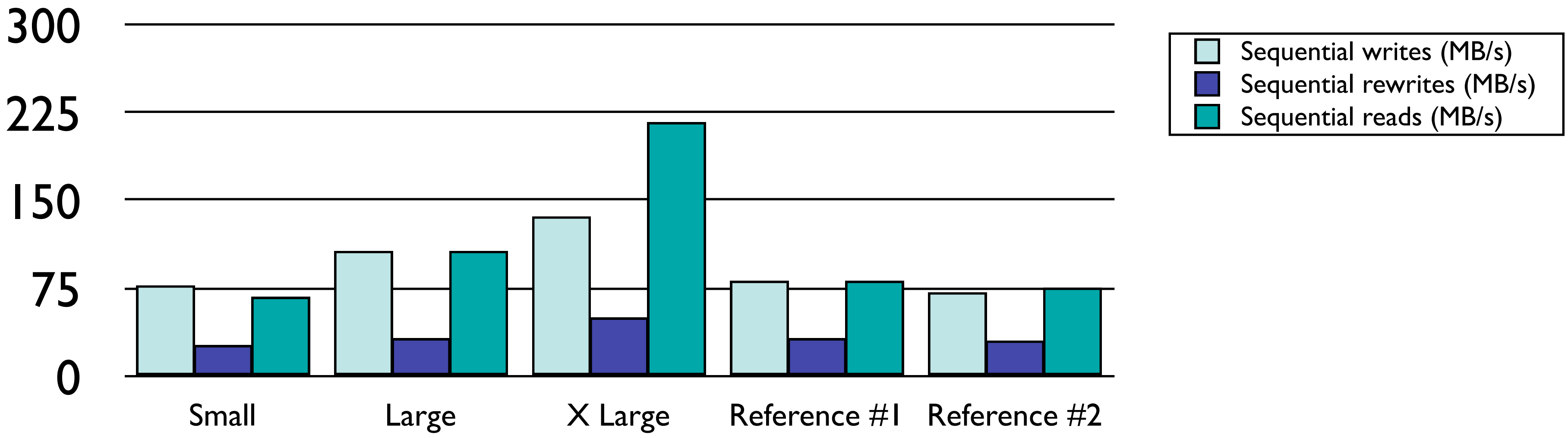
Now we’re past that false start, let’s start again!

Raw Disk Speed



- The average speed in writing an 11GB file to /mnt. Large and XLarge instances used software RAID (striping). Full disclosure TBA on http://tocker.id.au/ in April 08.

Bonnie++ Tests

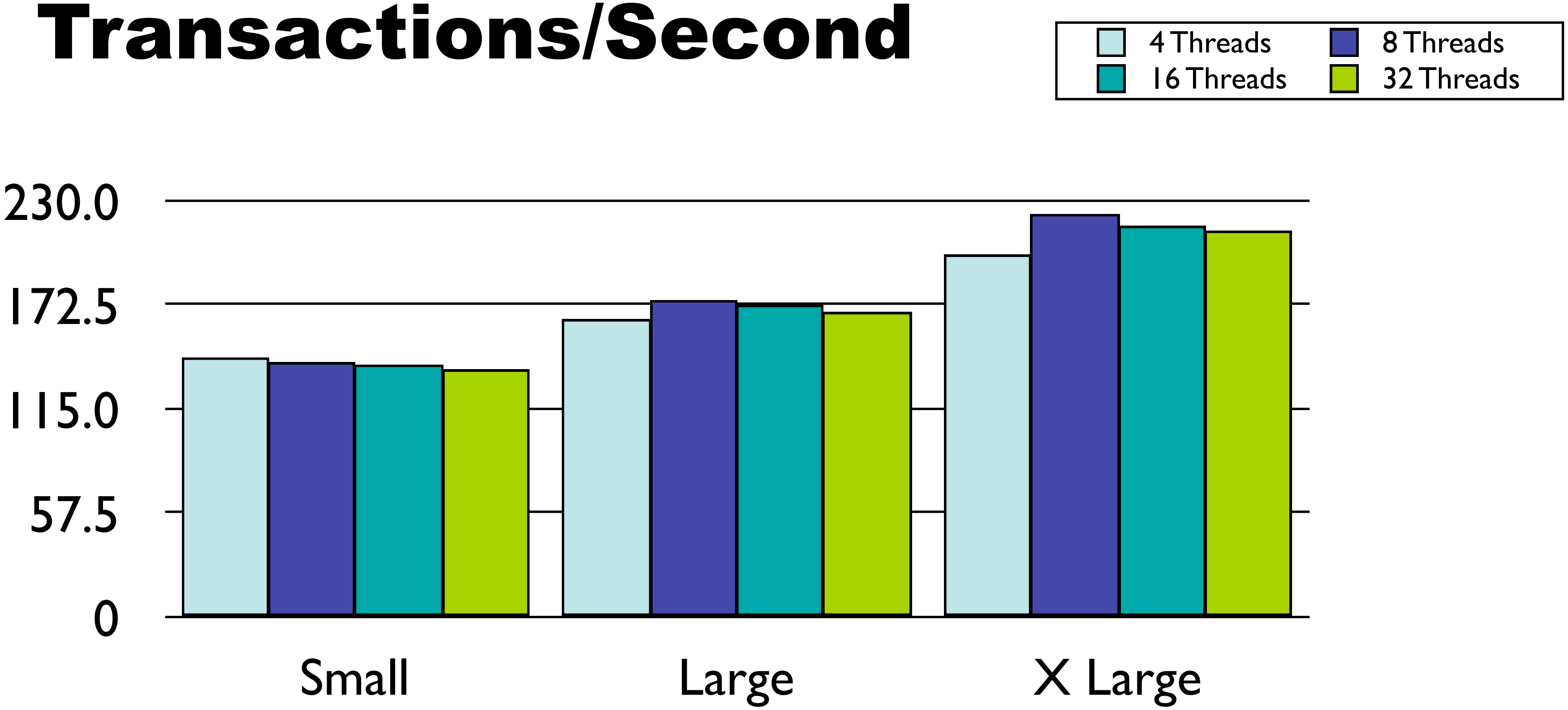


- All tests performed on /mnt, with software RAID. Reference #1 system was an Athlon XP 2500+ with a single 10,000 RPM SATA disk. Reference #2 was the same system with a single 7200RPM disk. Full disclosure will be on http://tocker.id.au/ in April 08.

Benchmarks

- It’s easy to lie.
- Next step after confirming raw performance is consistent, is to benchmark inside MySQL.
- This can be done with Sysbench **http://sysbench.sourceforge.net/**

Transactions/Second



Test was OLTP complex and InnoDB tables.

Benchmark Conclusions

- Comparable performance to non-virtual-machines.
- RAID0 or RAID0+1 with software RAID.

  Increased risk of failure with more spindles.
- Other sources have benchmarked CPU and network performance.

Limitations “Yesterday”

Limitations

- I had wanted to use DRBD / Heartbeat to reduce impact of failure. Can't do it because of possibility of split brains.

- Can't swap in another disk subsystem. No Hardware RAID or BBU.

- Not so much documentation on hardware.

Limitations (cont.)

- Wanted to know if writes persist on disk - not every possible to tell.
- For Defensio losing a few rows is annoying, but you would be insane if this were a financial application.

Limitations (cont.)

- Amazon seems “reasonably friendly” about giving impending failure notice.
- If a disk in your software raid dies or the network card dies, they’re going to make you move off to fix it.
- This is going to annoy you.

Limitations (cont.)

- S3 has too high latency to mount in FUSE and try and use for persistent storage (not designed for this either).
- Planning to use S3 on slave/reports server and push snapshots to it.
- Can’t increase the size of an instance.

Yeah, “Yesterday”

Amazon's Announcements

- Persistent Storage for Amazon EC2 http://www.allthingsdistributed.com/2008/04/persistent_storage_for_amazon.html

- On the Road to Highly Available EC2 Applications http://www.allthingsdistributed.com/2008/03/on_the_road_to_highly_availabl.html

Limitations (cont.)

- No higher availability instances.
- No way to a la carte add storage.
- No way to quickly migrate and recover from instance failure.
- Not easy to guarantee that an instance was on a different physical node than another instance.

Limitations (cont.)

- ~~No higher availability instances.~~
- ~~No way to a la carte add storage.~~
- ~~No way to quickly migrate and recover from instance failure.~~
- ~~Not easy to guarantee that an instance was on a different physical node than another instance.~~

War Stories

War Stories

- Possible bug in Replication (BUG #26489).
- Possible memory leak in MySQL (was reasonably elegant just to upgrade and shutdown - can’t do with other hosting).

War Stories (cont.)

- Degraded Nodes x2.
- The usual “Replication is asynchronous” (plan accordingly) dilemmas.

Conclusions

Conclusions

- Good value in Small and Large instances.
- For Defensio’s architecture might buy two Large rather than one XL machine.
- $600/month for XL is quite expensive.

Conclusions (cont.)

- Failure rate has been unusually high - probably bad luck.

- Might not suit people who fill up the disks on X.Large completely due to time to restore.

Conclusions (cont.)

- Does not offer **durability** (non-acid compliant).
- ~~Wish there were more instance types or a way to order features "a la carte".~~
- ~~Wish migrating data off a failed node was easier.~~

MySQL®
Conference & Expo

Presented by,
MySQL & O'Reilly Media, Inc.

The End.

Questions?

morgan@mysql.com
carl@defensio.com