The Proper Care and Feeding of a MySQL Server for Busy Linux Admins

Dave Stokes
MySQL Community Manager

Email: David.Stokes@Oracle.com
Twiter: @Stoker
Slides: slideshare.net/davestokes

Safe Harbor Agreement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decision. The development, release, and timing of any features or functionality described for Oracle’s products remains at the sole discretion of Oracle.

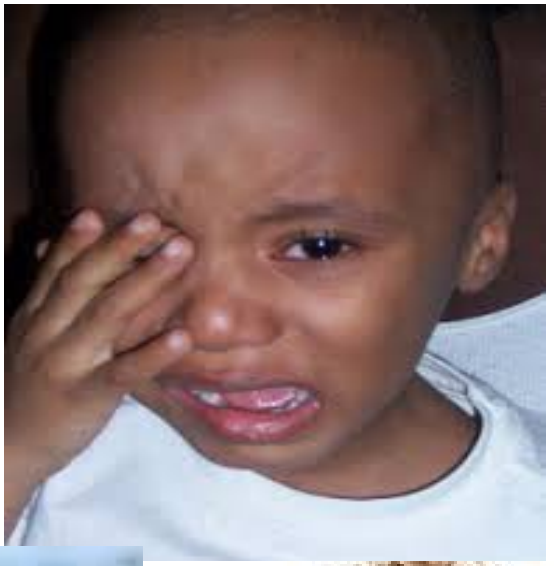
So take anything on future products with a grain of

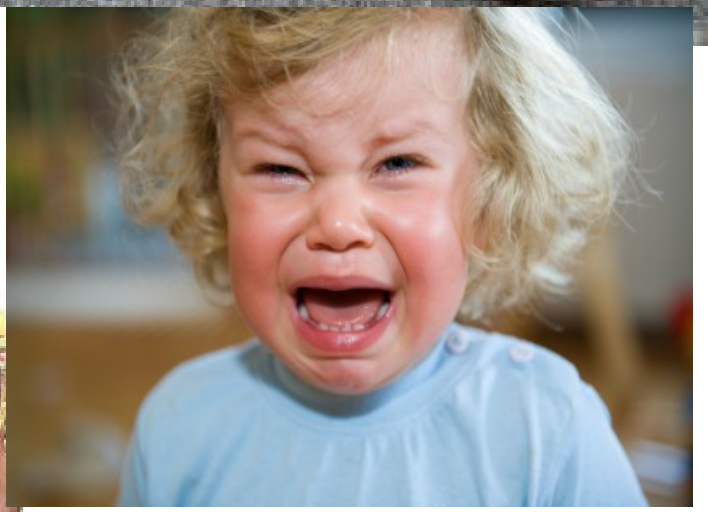
Happy Birthday to MySQL

Databases are

- Selfish
  - Want entire system to self
- Messy
  - Suck up memory, disk space, bandwidth, sanity
- Growing all the time
  - Needs updates
- Suck up a good part of your life

Databases are *nasty* toddlers!!





The previous opinions are

- The views of most admins (and developers) who *also* have DBA duties added to all their other regular tasks to help fill all their lavish spare time*

The previous opinions are

- The views of most admins (and developers) who *also* have DBA duties added to all their other regular tasks to help fill all their lavish spare time*

* This is the opinion of their bosses between rounds of golf, frozen adult beverages, and private jet 'business trip' to tropical locations.

Happy MySQL Databases

- Hardware
- Software
- Backups & Replication
- Tools to make life easier
- Configuration suggestions
- Q&A

Hardware Happiness

- Databases *LOVE* memory
  - Spend money on good memory
  - Lots of it
  - More important than cores

Why memory?

- Reading from memory is 100,000 faster than reading from disk
  - 100K miles if four times plus around the world
  - At one situp a second, it would take 27.78 hours to do 100K

Disks or Disk Like Things

- Move data to separate controller/disk from logs
- Do not log on slow devices
- RAID to your favorite level – RAID 10 Minimum
- DO NOT USE Consumer Grade Disks
  - Use disks that are designed for 7/24/365 operation not a price point
- CACHES – disk and controller
  - Write through or write back caches
    - Both lie
    - Make sure they don't auto tune during production hours
- FusionIO cards
    - Atomic rites = No Double buffering , therefor *SPEED!!*

Network

- Never expose your instances to outside world
- SCRUB *rigorously* all user data
- Keep separate net for replication and/or backup
- MySQL authentications uses host, user & Password
  - Boss@Home May not equal Boss@Work
  - Overly enthusiastic, first match in table gets in!!!
  - Set switch to not do lookup in case DNS overloaded

Slave Servers

- Slaves need to have bigger/badder hardware than master
  - Do more work
  - Use MySQL Utilities to clone masters, set up slaves
  - Dedicated network to avoid network contention

Software

- Run the latest greatest version of MySQL you can
  - Performance
  - Bug fixes
  - Features
- Keep MySQL by itself
  - *Databases do not play well with other services*
  - Contention for resources
  - Swapping
  - Maybe a caching layer but *watch* memory use!!

Backups & Replication

- Thou shalt make backups!!!
  - Make sure you backup your data so frequently it verges on being painful and then look for ways to increase *that!*
  - Know how to restore entire instances, entire databases, or a table. Save views, functions, etc. And others on staff need to be able to do this too (cross train)
  - Keep off site backups off site and test randomly
    - Make sure multiple people can get to off site backups
- Nobody ever got fired for doing too many backups – Paranoia should be your friend!!!!

Replication

- MySQL replications is easy to set up and misunderstand
  - Two types of replications
    - Async – slave grabs copy of changes from master and applies them to own set of data, master unaware of what slave is doing
    - Semi-sync – master waits for acknowledgment from at least one slave before proceeding
  - Three forms – Statement, Row, and Mixed
- Single threaded before 5.6, multi threaded for different databases in 5,6, multi infra databases threaded for 5.7

Replication Filters

- Do not need to replicate everything
  - Check churn of data, maybe 1x day backup
- Filter tables
- Can change filters on the fly with 5.7
  - “Something going on in manufacturing, can we get ALL of their data copied someplace?'

Global Transaction IDs

- Each transaction has unique GTID starting 5.6
  - Easy for slaves to get caught up to master
  - No longer have to look at file offsets on master and slave to get start position
    - Saves time and $ and ***sanity***
- Storing replication data in InnoDB tables plus adding check sums make crash safe
- Row based can exploit only sending key and changed items, not entire row of data

Multimaster and Multisource

- Multimaster
  - Not recommended but many do it
  - System A auto_increment odd numbers and System B auto_increment even numbers
    - Needs to be watched
- Multi source – MySQL 5.7
  - Multiple masters send data to one slave for master backup
    - Make sure sharded data does not overlap

Oracle Database Shops!

IF you also have a big Oracle DB shop:

- You can backup to the big Oracle STB backup devices
  - Great if you are in an Oracle shop
- MySQL can use oracle Database Firewall & Audit Vault
- Enterprise Customers
  - Audit Vault

Replication for backup

- Replication uses three threads
  - Master to slave
  - Slave to log
  - Log to data
- Shut down log to data thread, run backup, then restart log to data
  - Data from master still stored but not written during backup but applied when backup is done

Tools to make life easier

- There are lots of tools to make life easier for DBA chores
  - Monitoring
    - Yes, you *need* to monitor
  - Administration
    - Yes, you can type everything by hand on the command line but don't you have better things to do!?
    - Documentation of instances
    - Backup

Monitoring

- Active
  - Watches instances and send alerts
    - MySQL Enterprise Monitor (supported customers)
    - Nagios, cacti, etc.
      - Percona has plugins
    - Your favorite that is not mentioned
    - Helps to be able to comb historical data
- Semi-active
  - MySQL Workbench
    - Dashboard & SYS Schema
    - PhpMyAdmin
    - Your favorite tool that is not mentioned

You can't the **full size** of a problem at first glance!!

MySQL Workbench

- Query tool
  - Visual Explain to aid in optimization
- Admin tool
  - Users, backup, imports, change settings
  - No more fat finger 'UPDATE user set 'SELECT_PRIV='Y',....
- Dashboard and System Monitoring
  - Sys Schema
- Entity Relationship Mapper
- Migration tool
- And more!

MySQL Utilities

- Written in Python, easy to extend
- Setup replication and automatic fail over
- Copy user settings
- Copy data
- Look for bad processed and kill 'em
- Move binary logs
- Grep for a columng
- And much more

And more

- Percona tool kit
- Toad for MySQL from Dell
- Your favorite tool that is not mentioned

Config *Suggestions*

- Turn off DNS lookups – zone transfer dies
  - Use skip-name-resolve
- Save/Load statistics
  - Use innodb_stats_persistent
  - See 1**4.13.16.1 Configuring Persistent Optimizer Statistics Parameters** in the MySQL Manual
  - innodb_buffer_pool_dump=ON
  - innodb_buffer_pool_dump_at_shutdown=ON & innodb_buffer_pool_load_at_startup=ON

Config continued

- Tune log level (5.7)
  - log_error_verbosity – errors, errors & warnings, E&W + notes
  - Send to SYSLOG
- Turn off query cache (5.7 Default)
  - Single threaded, use memcached/redis
  - Free up memory
- InnoDB buffer pool size
  - 75-80% of RAM

Big Hint #1

- BE *DAMN* STINGY with permissions & grants
  - Easier to say no than to constantly be restoring
  - --safe-updates or –i-am-a-dummy
    - No more 'opps, I forgot the where clause'

Big Hint #2

- Sys_schema – please use
  - Views, functions, and procedures on top of the Performance_schema and Information_schema
    - Who is hogging resources
    - Indexes not being used
    - Problematic queries
    - Other routine PITAs

Big Hint #3

- 5.7 Security
  - Secure install becomes the default
    - Forced root password
    - No anonymous account, no test DB
  - Password rotation
  - Configure rules
    - Length, characters
  - mysql_config_editor (5.6.6)
    - Store encrypted auth credentials (no clear text)
    - Use mysql --login-path=finance

Q&A

- Slides: slideshare.net/davestokes
- Twitter: @Stoker
- Email: David.Stokes@Oracle.com
- Blog: OpenSourceDBA.wordpress.com