# Backing Up MySQL

**http://bit.ly/mysqlbackups**


Sheeri Cabral

Senior DB Admin/Architect, Mozilla

@sheeri  www.sheeri.com

# Who am I?

Oracle ACE Director for MySQL

Worked at Pythian for a few years

Currently working at Mozilla

On the IOUG MySQL Council
www.ioug.org/mysql_council

# Defining Terminology


DILBERT ©1998 Scott Adams.

# Defining Terminology

Logical

Physical

Consistent

Inconsistent

# Defining Terminology

Cold

Hot

Warm

# Is a Slave a Backup?

You can only get “now”

Delayed replication

- Percona toolkit, MySQL 5.6

How long do you delay?

Who here has valid backups?

There are no valid backups,
only valid restores.

# Schrodinger's Backup

# Why Make Backups?

Database Migration/Upgrade

Analysis/Reporting

Archives

Recovery

# What Kind of Recovery?

One user's data

Data corruption

Data loss

How much data can you lose?

# Who Is Responsible?

Ultimately, head of IT

DBA?

Sysadmin?

Backup Service?

Operators?

Developers for development machines?

# What do you backup?

Some or all data

Slave position

Binary log position

Logs containing commands

These get easier in MySQL 5.6!

# Expectations

Is dev backed up?

- Do you make that public?

Will folks play fast and loose with their data if backups are easy?

Automatic refreshes?

Manual, but does not need you?

# Where Are Backups Taken?

Master

Slave

Non-end-user affecting slave:

- Reporting slave
- Admin slave
- Backup slave

# When Are Backups Taken?

Where they are taken matters

If backup slave, any time!

Static data backed up less frequently?

# Backups for Disaster Recovery

What kinds of disasters?

- Member deletes data/pay status expires
- Dev/admin drops table/database
- Server has disk corruption
- ~~Server has network problems~~
- Data center blows up

Non-disaster

- Migrate data (e.g. reporting server)

Backup use determines backup method

# Backups for Disaster Recovery

What kinds of recovery?

- Member deletes data/pay status expires
  - Logical export so you can grep
- Migrate data
  - Logical export for flexibility
- Dev/admin drops table/database
  - Logical export for InnoDB
  - Physical export for MyISAM

# Backups for Disaster Recovery

What kinds of recovery?

- Full server recovery
  - Physical backup
  - Config file(s)
- Point-in-time recovery (PITR)
  - Physical backup
  - Config file(s)
  - Binary logs

# Logical Backups

SELECT … INTO

CSV table

mysqldump

# SELECT...INTO

- DUMPFILE
  - No formatting
  - One row only
- OUTFILE
  - FIELDS ESCAPED BY
  - FIELDS [OPTIONALLY] ENCLOSED BY
  - LINES TERMINATED BY

http://dev.mysql.com/doc/refman/5.5/en/select-into.html

# Examples for SELECT...INTO

```
mysql> SELECT * INTO DUMPFILE
  '/tmp/backup_actor.txt' FROM sakila.actor;

ERROR 1172 (42000): Result consisted of more than
  one row

mysql> \! rm /tmp/backup_actor.txt
```

# Examples for SELECT...INTO

```
mysql> SELECT * INTO DUMPFILE
  '/tmp/backup_actor.txt' FROM sakila.actor limit 1;

Query OK, 1 row affected (0.00 sec)


$ cat /tmp/backup_actor.txt
1PENELOPEGUINESS2006-02-15 04:34:33


mysql> SELECT * FROM sakila.actor limit 1\G
*************************** 1. row ***************************
   actor_id: 1
 first_name: PENELOPE
  last_name: GUINESS
last_update: 2006-02-15 04:34:33
1 row in set (0.00 sec)
```

# Examples for SELECT...INTO

```
mysql> SELECT * INTO OUTFILE '/tmp/backup_actor.tab'
  FROM sakila.actor;
Query OK, 200 rows affected (0.00 sec)

$ head -5 /tmp/backup_actor.tab
1       PENELOPE        GUINESS 2006-02-15 04:34:33
2       NICK    WAHLBERG        2006-02-15 04:34:33
3       ED      CHASE   2006-02-15 04:34:33
4       JENNIFER        DAVIS   2006-02-15 04:34:33
5       JOHNNY  LOLLOBRIGIDA    2006-02-15 04:34:33
```

# When to use SELECT...INTO

You only want certain fields

You want to join tables

Usually for reporting migrations

SELECT...INTO DUMPFILE not used much

# Problems with SELECT...INTO

Only one query at at time

Need to lock tables if you want consistency

Does not backup binlog position

Does not backup slave status

Restore can take a long time

SELECT makes a shared lock

- Writes cannot happen

# Restoring from SELECT...INTO

...or any other data file

LOAD DATA INFILE

- FIELDS ESCAPED BY
- FIELDS [OPTIONALLY] ENCLOSED BY
- LINES TERMINATED BY

# CSV Table

```
mysql>  CREATE TABLE test_csv
(name VARCHAR(50) NOT NULL,
claim_to_fame VARCHAR(50) NOT NULL) ENGINE=CSV;
Query OK, 0 rows affected (0.03 sec)

mysql> INSERT INTO test_csv (name,claim_to_fame)
 VALUES('Sheeri Cabral','this talk'),
('Neil Patrick Harris','actor/magician'),
('Gerry Narvaja','oursql podcast co-host');
Query OK, 3 rows affected (0.00 sec)
Records: 3  Duplicates: 0  Warnings: 0
```

# CSV Table

```
localhost:test root# file test_csv.*
test_csv.CSM: MySQL table definition file Version 0
test_csv.CSV: ASCII text
test_csv.frm: MySQL table definition file Version 10

localhost:test root# cat test_csv.CSV
"Sheeri Cabral","this talk"
"Neil Patrick Harris","actor/magician"
"Gerry Narvaja","oursql podcast co-host"
```

# Benefits of Using CSV Table

Hybrid logical/physical backups

- Can copy the .CSV file
- Low impact to server
- Multiple restore options

# Gotchas with Using CSV Table

CSV does not support everything

- e.g. no auto increment primary key

One table at a time

- No joins

Logical import can be tedious

Physical import requires schema

# How to Restore CSV table

```
mysql>  CREATE TABLE test2_csv
(name VARCHAR(50) NOT NULL,
claim_to_fame VARCHAR(50) NOT NULL) ENGINE=CSV;
Query OK, 0 rows affected (0.03 sec)

$ cd /usr/local/mysql/data
$ sudo cp test_csv.CSV test2_csv.CSV
```

# mysqldump has lots of flags

--fields-terminated-by

--fields-enclosed-by

--fields-optionally-enclosed-by

--fields-escaped-by

--lines-terminated-by

Looks kind of like SELECT...INTO

--xml

# mysqldump defaults

--lock-tables

--quick (no buffering, direct to stdout)

--extended-insert

# mysqldump

Choosing what to backup

- mysqldump db tbl
- --all-databases
- --databases
- --tables
- --events
- --routines
- --triggers

# mysqldump output

--result-file (or just redirect stdout with >)

--master-data

--dump-slave

# mysqldump tweaks

--no-create-info

--no-data

--insert ignore

--replace

--single-transaction

# mysqldump restore

mysql < backup.sql

Or import text files as from SELECT...INTO

# mysqldump gotchas

Locking

Lukewarm backup

Logical imports take time to import

# Physical Backups

File copy

Xtrabackup

MySQL Enterprise Backup

# File copy

Cold

Consistent

Easy

# File copy

Shut down MySQL

Copy/archive files

Start MySQL

# File copy

Requires shutdown

Make sure to grab all the files you need

You don't need all the files in $datadir

# xtrabackup

Free and open source

From Percona

Backs up InnoDB, XtraDB

# xtrabackup

innobackupex wrapper script

Based on older InnoDB hot backup

“warm” backup

# How xtrabackup works

# xtrabackup Features

Physical backup

Stream or copy to remote host directly

Full backup

Incremental backup

# xtrabackup Features

Backup/restore individual partitions

Backup/restore databases/tables

- by regular expression

# xtrabackup Features

Compressed backup

Easy “recipes” to follow

Can specify multiple threads

- will increase I/O load

Can throttle IO/s

# xtrabackup example: New Slave

On the slave, install MySQL, set up my.cnf

Start listening with netcat, and extract with tar:

slave$ nc -l 9999 | tar xfi -

# xtrabackup example: New Slave

On the master, stream the backup to the slave

```
master$ innobackupex --stream=tar | nc
   slave.company.com 9999
```

# xtrabackup example: New Slave

On the slave, apply the logs

slave$ cd /var/lib/mysql

slave$ innobackupex –apply-logs .

slave$ chown -R mysql:mysql /var/lib/mysql

slave$ /etc/init.d/mysql start

Use the binary log file/position in xtrabackup_binlog_info

# Zmanda Recovery Manager

http://www.zmanda.com

Full and incremental backups

Schedule backups

Logical or Physical backups

# Zmanda Recovery Manager

Compression

Encryption

Notification via e-mail or RSS feed

Monitor and browse backups

# Zmanda Recovery Manager

Define retention policies

Full or point-in-time recovery

Binary log management and parsing

# MySQL Enterprise Backup

Full/incremental backup

Backup tables/databases

Supports hot and cold backups

# MySQL Enterprise Backup

Full restore

Point-in-time restore

Can be multi-threaded

Compressed backups

# MySQL Enterprise Backup

Uses checksums to detect corruption

Faster, smaller backups

- does not copy unused blocks

Consistent

# Other Backup Methods

Disk snapshots:

- ZFS
- LVM
- Amazon images

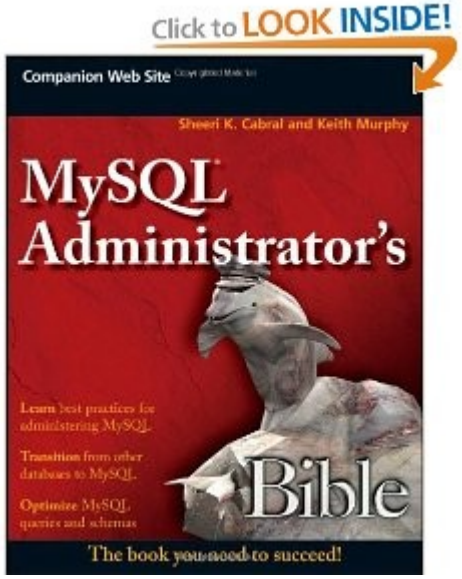
# Questions? Comments?

OurSQL Podcast

www.oursql.com

MySQL Administrator's Bible

- tinyurl.com/mysqlbible




mysqlmarinate.com

planet.mysql.com