Lessons from an Interactive Environment

Bruce Dembecki
Vice President, Solutions Development
LiveWorld, Inc
Lessons from an Interactive Environment

- Why LiveWorld?
- Interactive vs Published
- Our Database Server Setup
- Things that make a difference
Why LiveWorld?

• Leading provider of collaborative/community solutions since 1996
• High profile high volume client base
• High Volume High Redundancy configuration
• Unique database deployment
LiveWorld: Working with Market leaders
Just the facts, Ma’am
(Feb ‘05)

- 18 Discussion Board Clients
- 78 Discussion Boards
- 30 Countries
- 18 Languages
- 99.6M page views
Just the facts, Ma’am
(Feb ‘05)

• 4 and a bit MySQL Servers (4.0.23)
• 1478 queries per second average
• 78 databases
• 112 daily server backups
• 1344 server snapshots
Interactive vs Published

• Published Content
  – Pushed from one source
  – Static content or Search
  – One master, many slaves, even many locations
  – CNN, Yahoo, eBay Auctions
Interactive vs Published

• Interactive Content
  – Content comes from multiple sources
  – Many to Many (chat, discussion boards etc)
  – All “application servers” need to change data
  – All Slaves must be Masters
LiveWorld Database Architecture

MySQL Servers

HW Load Balancers Distribute Connections from App Servers

Production Servers talk to Database Pairs

mysqlld_multi to enable better use of server memory (2 - 3 instances of mysqld per server)

Replication keeps data synchronized between both servers in pair

One way “delayed” replication to MySQL Admin

MySQL Admin

• Nightly Backups
• 2 hourly Snapshots
• Stats & Client Reports
• Watches (digests)
Database Hardware Platform

- Apple XServe with Dual 2.3Ghz G5
- 8 Gbyte ECC RAM
- 400 Gbyte System Drive
- 2 x 400 Gbyte drives on hardware RAID card
Production vs Admin

- Production machines = fast
- Admin tasks = slow
- Admin impact on Production servers = :-(
- Separate Production & Admin tasks
mysql-admin

- Many instances of mysqld
- Each instance slaves one production pair
- Reports, Backup, List Watches
- Complete logs here, less in Production

SET SQL_LOG_BIN=0;
#!/bin/sh
#
# LiveWorld's Delete Old Stats Script (Local delete only)
#
# Get the date in Seconds and subtract 7 days, turn it into milliseconds
#
seconds=`date +%s`
seconds=$((seconds-604800))
seconds=$((seconds*1000))

#
# Loop for each database... connect, turn off binlog, delete logs older than 30 days
#
for DB in `mysql --exec="SHOW DATABASES LIKE 'boards%';" --column-names=FALSE`
do

        mysql --exec="SET SQL_LOG_BIN=0; DELETE FROM stats WHERE timestamp < $seconds;" $DB

done
“Delayed” Replication

- Makes use of MySQL’s 2 Replication threads
- Cron controls reads and executes on slave

Timeline

4:00pm
Slave’s SQL Thread Stopped

4:01pm
Slave’s I/O Thread Started

4:05pm
Slave’s I/O Thread Stopped
Flush logs

4:10pm
Slave’s SQL Thread Started

Reading Master changes made between 2:05pm and 4:05pm

Executing Master changes made between 2:05pm and 4:05pm
“Delayed” Replication

- Quick recovery if caught before execution
- `mysqladmin flush-logs` creates “snapshots”
- Snapshots + nightly backup = easy rollback
- Reports & Watches run from “delayed” data
“Delayed” Replication

• “start” scripts use if [ -f /var/mysql/replicate ]

• Emergency Stop Script:
  – rm /var/mysql/replicate
  – execute all “stop” scripts

• Link Emergency Stop Script to TCP Port
Living with your memory

- Use as much memory as you can
- InnoDB uses its own memory pool
- MySQL/MyISAM + InnoDB = 2 memory pools
- 32 Bit vs 64 Bit
Things that make a difference, “Many to Many” or “One to Many”

• Tuning... indexes, caches
• Separate Admin from Production
• Monitor & Notify
• Hardware
• Automate
Lessons from an Interactive Environment

Bruce Dembecki

bruce@liveworld.com