MySQL and OpenSolaris

Martin 'MC' Brown
Technical Writer
Database Group

About me

- Technical Writer at MySQL
- Long time Solaris user
- OpenSolaris Contributor
- Past Life

Introduction to MySQL

- MySQL is a full SQL RDBMS
  - SQL-compliant
  - Multi-threaded
  - Wide range of datatypes
  - Capable of handling petabytes+ of data
- Very Fast
- Very Configurable
- But also usable straight out the box
- Compatible with many languages and environments

MySQL Functionality (5.0+)

- Transactions/ACID (with suitable Storage Engine)
- Stored Routines
- Triggers
- Strict modes and error handling
- Views
- XA (Distributed) Transactions
- Information Schema (metadata)

MySQL Storage Engines

- Wide selection
  - MyISAM
  - Maria (MyISAM++)
  - InnoDB
  - Memory
  - Archive
  - Federated
  - NDB (Cluster)
- Mixed tables/cross joins

Comparing MySQL 5.0 and 5.1

- MySQL 5.1 in RC state - coming soon
- Main features:
  - Partitioning
  - Row-based Replication
  - Plugin API
  - Event Scheduler
  - XML Functions

MySQL 6.0

- Falcon Storage Engine
  - Designed for High CPU/High RAM/Web-style loads
  - MVCC
  - ACID
- Maria (now in 5.1)
  - MyISAM with Crash Protection
- Backup/Restore Database

MySQL in the Web stack

- The M in LAMP, SAMP, WAMP
  - (Linux|Solaris|Windows) Apache MySQL (Perl|Python|PHP)
- OSAMP (OpenSolaris...)
- Part of the Solaris Webstack and Coolstack
- Integration with other Web tools
  - Perl/PHP/Python
  - Ruby, Java, etc...
- Works with Glassfish and NetBeans

Scalability

- Replication for read scalability
  - Add multiple ‘slaves’
  - Replicate the data to the slaves
  - Use slaves for read
  - Write to ‘master’
- Sharding/Partitioning
  - Splits users or objects into groups
  - Reads for an object go to a database handling objects for the group (shard)
- More exciting solutions at the end of the presentation!

MySQL Connectors

- Connector/JDBC (build 92)
- Connector/ODBC (coming)
- PHP (mysqli/mysqlnd)
- Connector/NET
- Connector/MXJ

MySQL Customers

- Google (Adsense)
- Yahoo
- Facebook
- YouTube
- Flickr
- Wikipedia
- Fotolog (Solaris)
- Go to coalface.mcslp.com for videos at MySQL Conference and Expo with the scalability panel

MySQL Elsewhere

- Ericsson
- Nortel
- Lucent
- Airbus/EADS
- CERN
- NASA
- For more: http://www.mysql.com/customers/

MySQL in OpenSolaris

- MySQL 5.0.45 (32-bit) in build 79
  - SXDE 01/08
- MySQL 5.0.45 (64-bit) in build 87
  - SXCE
- See coalface.mcslp.com for speed comparison
- MySQL 5.1 coming soon (32/64 bit)
- MySQL 5.0 and 5.1 will coexist
- MySQL 4.x being retired

Installing MySQL into OpenSolaris

- Earlier builds and SXDE/SXCE installed by default
- In OpenSolaris 2008.05:
  - # pkg install SUNWmysql5
  - # pkg install SUNWmysql5jdbc
- Other tools:
  - Standard PHP comes with MySQL support
  - Perl/Python/Ruby/others use native tools (CPAN, etc.)
  - Use WebStack or Coolstack

OpenSolaris MySQL Layout

- Standard layout:
  - Everything in /usr/local/mysql
    - Substructure of data, bin, etc, lib etc.
  - You can change this during build with --prefix
- Designed to support all versions
- Installation is /usr/mysql/5.0
- Data is /var/mysql/5.0/data
  - Standard is /var/lib/mysql
- Configuration is /etc/mysql/5.0
- Symbolic links /usr/mysql/bin -> /usr/mysql/5.0/bin

Starting MySQL

- Integrated into SMF
  - # svcadm enable mysql
- Change data directory:
  - Use svccfg
  - select mysql:version_50
  - setprop mysql/data=/my/new/directory
  - svcadm refresh mysql:version_50
- Enabling 64-bit
  - setprop mysql/enable_64bit=true

Using MySQL in Virtual Environments

- MySQL fully compatible with Zones/Containers
- Usual VM rules apply
  - Take care with resources
  - Disk biggest bottleneck
  - For max performance spread disk usage
- Some environments show increase

DTrace in MySQL

- Not in OpenSolaris MySQL
- Currently only in 6.0
- Working on 5.1
- Working on 5.1/6.0 Improvements

Current DTrace Probes

- Basic probe triggers only
  - SELECT start/end
  - INSERT start/end
  - DELETE start/end
  - UPDATE start/end
  - FILESORT start/end
  - External Lock

Forthcoming DTrace Improvements

- Adding arguments:
  - Query
  - User
  - Client hostname
  - Database
- Row counts and matched/affected
- Deeper into row-by-row operations
- Ultimately deeper into individual storage engines
  - Falcon now
  - Jenny Chen has InnoDB probes

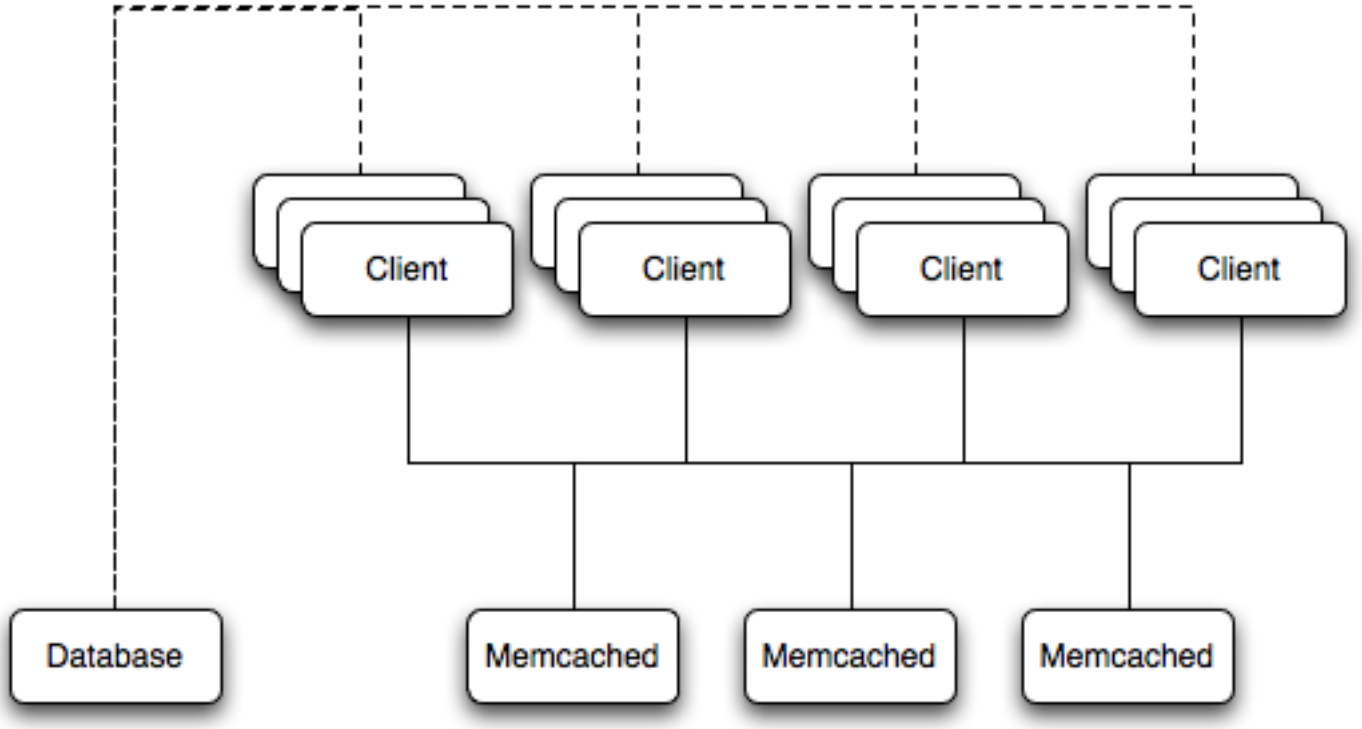
DTrace Example

- Potential new probes

```
Who                    Dur (ms) Query
root@localhost               0  select @@version_comment limit 1
root@localhost               0  SELECT DATABASE()
root@localhost               0  show databases
root@localhost               0  show tables
root@localhost               0  select * from t1 where i=99
root@localhost               9  select * from t1 limit 10
root@localhost             110  select count(*) from t1 where i=
```

MySQL and Memcached

- Distributed Memory Cache



Using Memcached

- Use with MySQL
  - Load data from cache
  - Not available, load from MySQL, insert into cache
- Use direct from MySQL (UDF) or through API
- memcached available in OpenSolaris soon

MySQL Proxy

- Sits between client and MySQL server
- Allows for inspection and manipulation of packets
- Can provide scalability and availability
- Will be particularly useful with replication

MySQL and OpenSolaris

Martin 'MC' Brown
Technical Writer
Database Group