



ORACLE[®]

MySQL Proxy meets: Memcache

Jan Kneschke

MySQL Enterprise Tools

Hands up

- MySQL 5.1 ?
- MySQL 5.5 ?

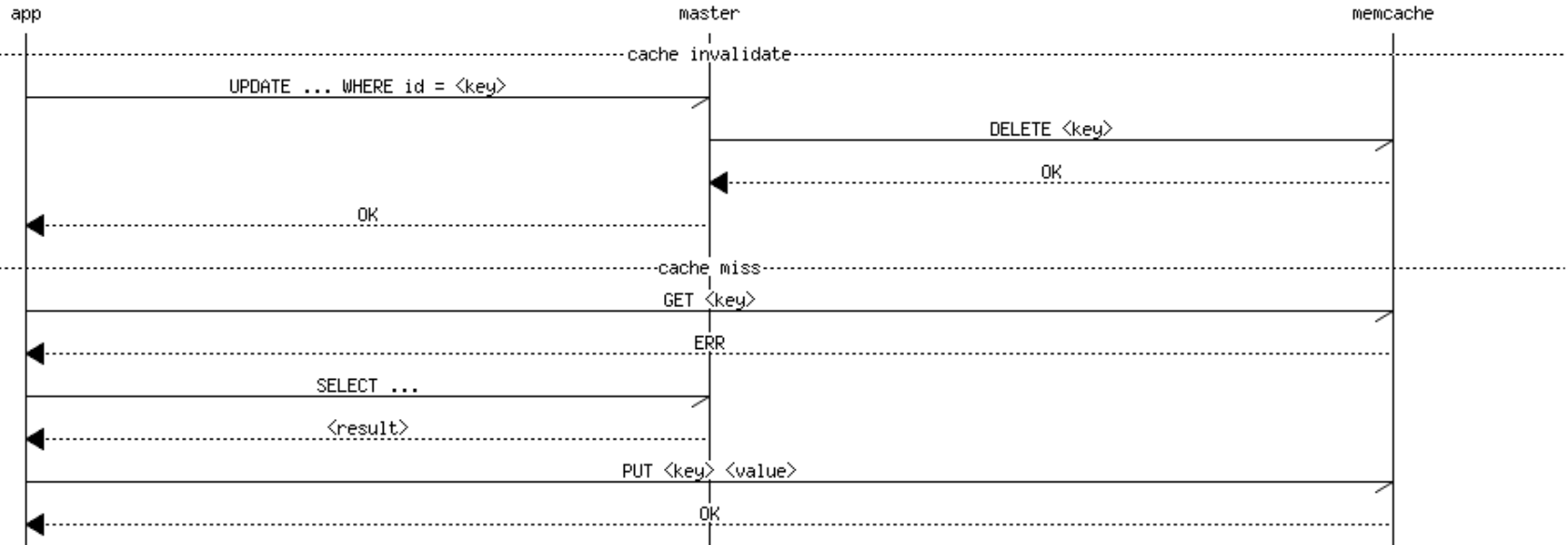
Memcache

- Distributed, memory based key-value storage
- Low latency
 - UDP, binary protocol
- Cache
 - TTL
 - Purge if full

Store in MySQL, cache in Memcache

- Normalized data in MySQL
 - Persistent
 - Consistent
- Cache
 - results of complex ops
 - Good cache hit ratio

Keep them in sync



Let MySQL do it

- Triggers on UPDATE, DELETE and INSERT
 - Memcache UDF <https://launchpad.net/memcached-udfs>
- Synchronous to the query execution
 - Move TRIGGERS to the slave for async

Let MySQL not do it

- Key mapping may be complex
 - Especially in stored procs
 - Memcache key == Primary key ?
- Replication is our friend

MySQL Replication

- Stream of change events
- Statement based
 - Embed memcache key in comment
 - `INSERT INTO tbl VALUES (1) /*!99999 key = 42 */`
- Row Based Replication since MySQL 5.1
 - Pre/Post images of the changed rows
 - Stripped down table definitions
 - No SQL parsing

Semi-Sync Replication

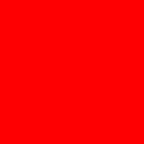
- MySQL 5.5 + semi-sync plugin
- Master waits for ACK on COMMITs
- Extended binlog network protocol

MySQL Proxy

- Framework to build low-level MySQL tools
- Chassis
- Non-Blocking IO, event-driven, threaded
- MySQL Protocol libraries
- Scripting layer

MySQL Proxy meets: Memcache

- Binlog in, Memcache out
- Based on the replicant plugin
 - Event iterator
 - SBR and RBR (5.1+)
 - Semi Sync
- Libmemcached for memcache communication



The preceding 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 decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Q&A

- Or just mail me at jan@mysql.com or @weigon
- See <http://jan.kneschke.de/>