

Galera Replication

codership

Seppo Jaakola, CEO

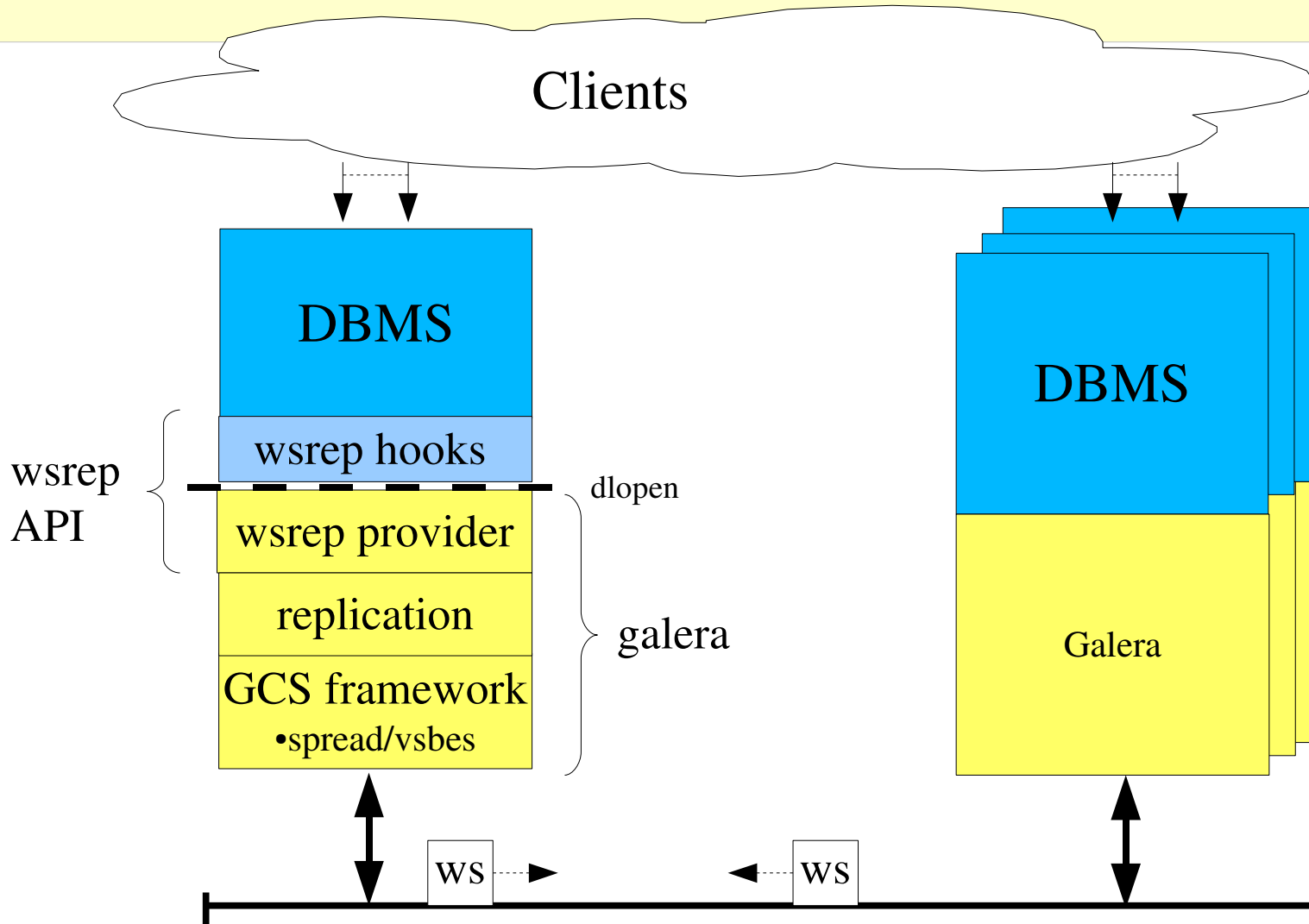
seppo.jaakola@codership.com

<http://www.codership.com>

Galera Replication

- Multi-master synchronous replication system
- Certification based replication model (based on academic research by F. Pedone et al)
- Avoids using middle-ware, connections go directly to DBMS -> transparency
- Row level locking -> write scalability
- Generic replication system to make a cluster from any transactional DBMS
- First implementation MySQL/Innodb cluster

Galera Cluster



wsrep API

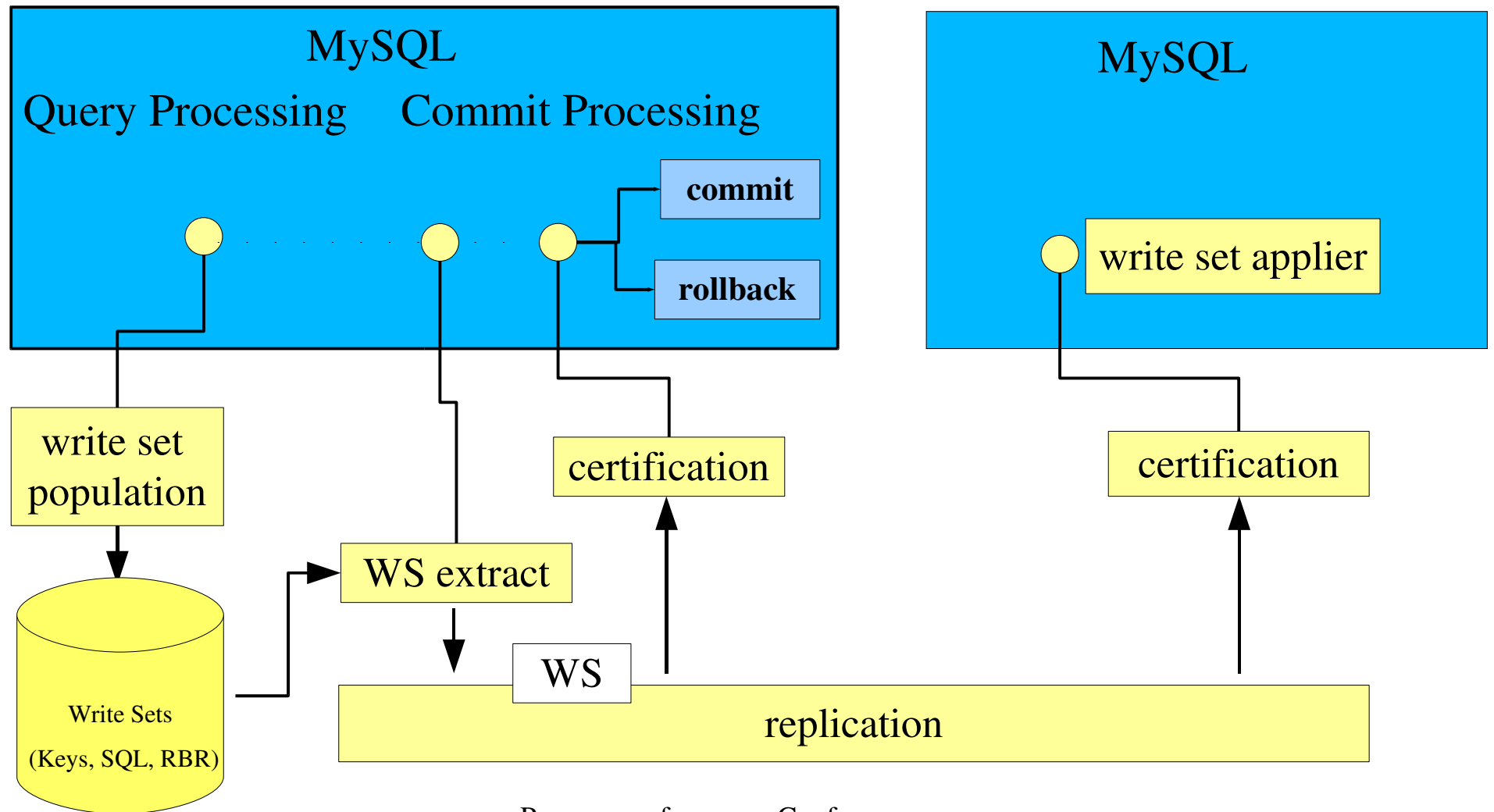


- Defines a generic interface for DBMS and replication system
- Write set replication API for transactions
- DDL replication using TO isolation
- Launchpad project: <https://launchpad.net/wsrep>

wsrep integration in MySQL

- Launchpad project:
<https://launchpad.net/codership-mysql>
- Calls to wsrep provider:
 - Ws populating, replication...
- Handlers for various wsrep callbacks:
 - ws applying, DDL applying ...
- Changes in innodb code to provide **prioritized transactions**

Certification Based Replication



Write Set

- Write set can contain data changes specified in different replication levels:
 - 1.SQL statement**
 - 2.Lex structures (AST) from parser
 - 3.RBR event**
 - 4.Row (as binary image)
- All row changes are identified with keys
- Last_seen_seqno & seqno tracking trx processing state
- Write sets can be saved for future needs

Replication Features

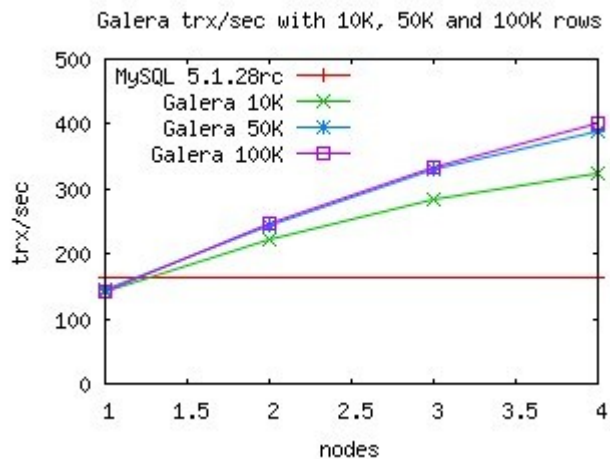
- Replication is optimistic in nature
 - Hot spots cause replication aborts
- Flow control
 - GCS feature to adjust nodes' progress
- Autoincrement management
 - Cluster adjusts increments and offsets on the fly
- Asymmetric lock granularity issue
 - Solved by replaying as slave trx
- Retrying of aborted autocommit trxs

Benchmarking

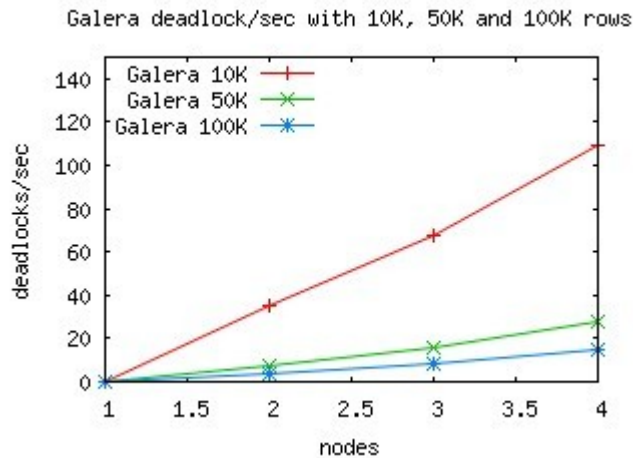
Benchmarking

- Tested with several benchmarks
 - Sysbench, dbt2, DOTS, osdb, jmeter, sqlgen...
- Benchmarks testing with 'physical hardware' and with Amazon EC2 small and large instances
- Currently tests only up to 5 cluster nodes
- In general, shows good scalability even with write intensive work loads

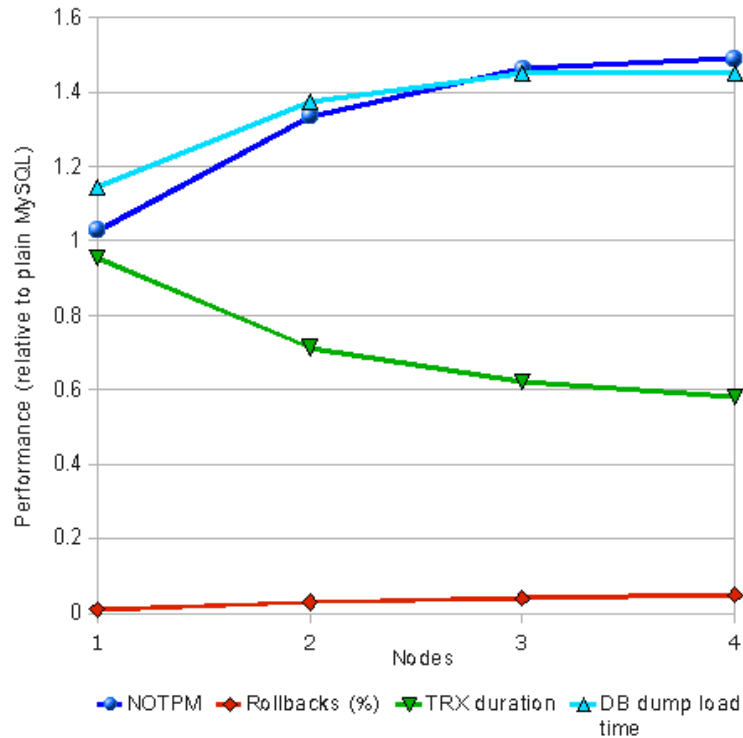
Sysbench Benchmarks



- Sysbench oltp mode test
- 10K – 100K table sizes
- Using 5 HP proliant servers



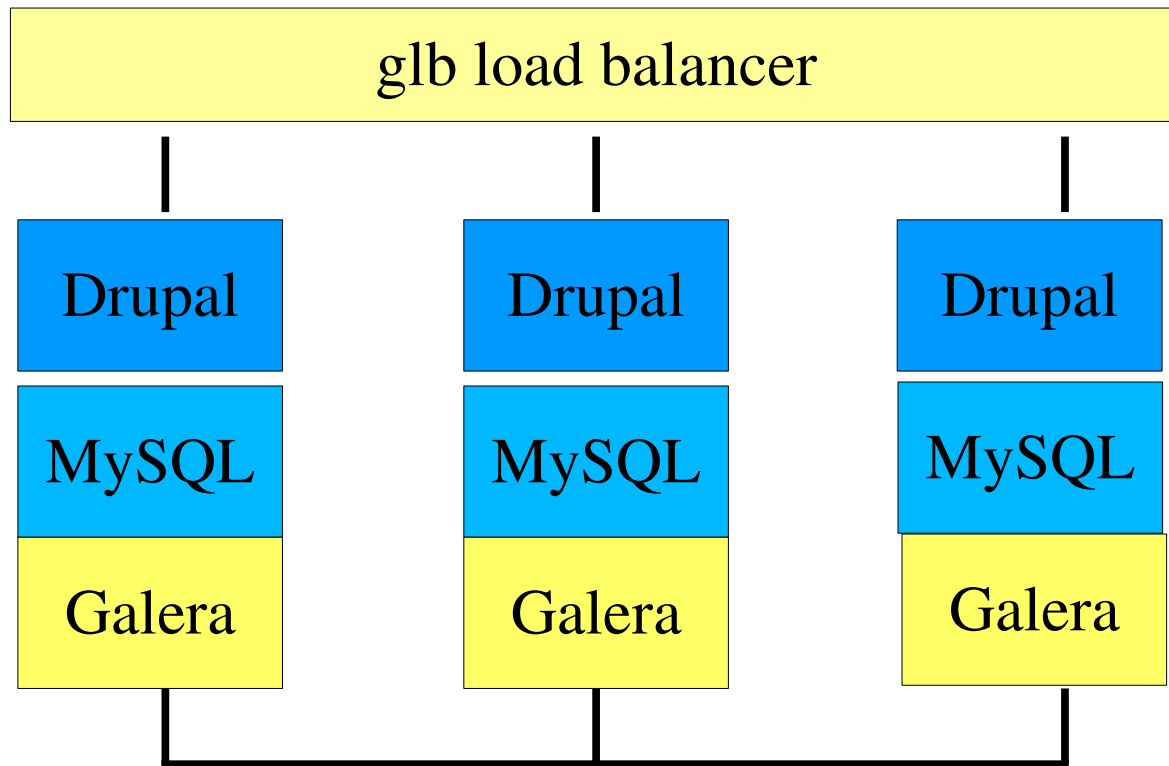
Dbt2 Benchmark



- EC2 large instances
- Dbt2 benchmark
- 60 warehouses

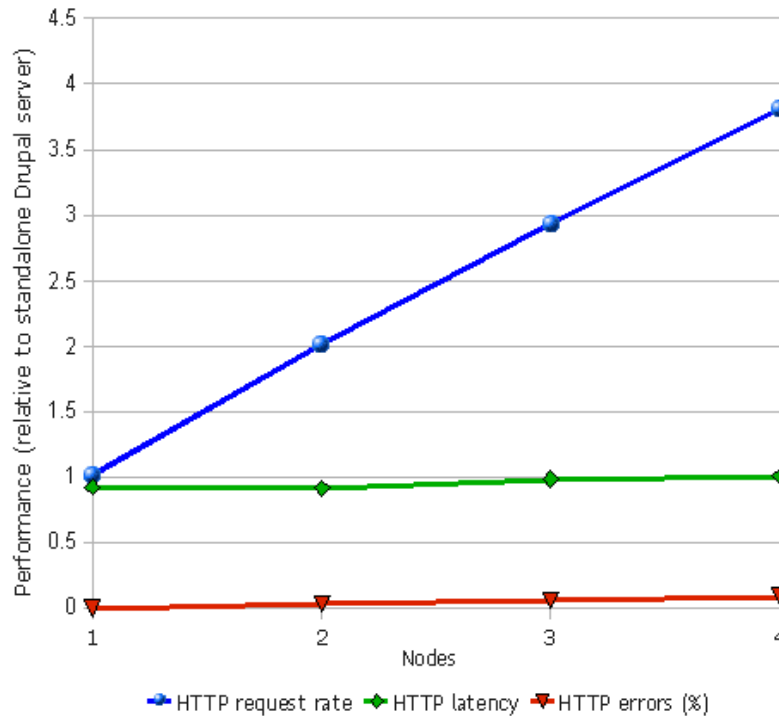
	Conns	NOTPM	Rollbacks(%)	TRX duration(sec)	Dump load(min)
Plain 5.1.30:	20	~7220	1	2.27	26
1 node	: 12	~7420	1	2.17	30
2 nodes	: 24	~9630	3	1.63	36
3 nodes	: 36	~10555	4	1.41	38
4 nodes	: 48	~10753	5	1.32	38

Drupal Scale-Out



- Proof of concept
- Each drupal node has local MySQL
- all nodes identical
- ~10% of CPU for MySQL
- glb load balancer

Drupal Cluster on AWS



- Jmeter test with 3 threadgroups
- Posters, commenters, browsers
- Testing with Amazon EC2 large instances

Nodes	Users	Throughput (req/min)	Latency (ms, median)	Latency (ms, average)	Errors (%)
1	180	724	1203	1827	0.00
2	360	1436	1190	1829	0.03
3	540	2091	1280	2150	0.06
4	720	2717	1214	2330	0.12

Summary

- High Availability
- Transparency
- Good scalability even with high write rates
- Roadmap:
 - feature complete release by Q2/09
 - GA release by Q4/09