MySQL Cluster
Web Scalability, 99.999% Availability

Andrew Morgan
@andrewmorgan
www.clusterdb.com
Safe Harbour Statement

The following 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.
### MySQL Cluster: Overview

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>HIGH SCALE, READS + WRITES</strong></td>
<td>- Auto-Sharding, Multi-Master</td>
</tr>
<tr>
<td></td>
<td>- ACID Compliant, OLTP + Real-Time Analytics</td>
</tr>
<tr>
<td><strong>99.999% AVAILABILITY</strong></td>
<td>- Shared nothing, no Single Point of Failure</td>
</tr>
<tr>
<td></td>
<td>- Self Healing + On-Line Operations</td>
</tr>
<tr>
<td><strong>REAL-TIME</strong></td>
<td>- In-Memory Optimization + Disk-Data</td>
</tr>
<tr>
<td></td>
<td>- Predictable Low-Latency, Bounded Access Time</td>
</tr>
<tr>
<td><strong>SQL + NoSQL</strong></td>
<td>- Key/Value + Complex, Relational Queries</td>
</tr>
<tr>
<td></td>
<td>- SQL + Memcached + JavaScript + Java + JPA + HTTP/REST &amp; C++</td>
</tr>
<tr>
<td><strong>LOW TCO</strong></td>
<td>- Open Source + Commercial Editions</td>
</tr>
<tr>
<td></td>
<td>- Commodity hardware + Management, Monitoring Tools</td>
</tr>
</tbody>
</table>
Who’s Using MySQL Cluster?
MySQL Cluster Architecture

Data Layer

MySQL Cluster Data Nodes

Management

Application Layer

Java

MySQL

node.js

Clients
MySQL Cluster Architecture
No Single Point of Failure

Clients

Application Layer

MySQL Cluster Data Nodes

Management
Automatic Data Partitioning

Table T1

<table>
<thead>
<tr>
<th>P1</th>
<th>P2</th>
<th>P3</th>
<th>P4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Px Partition</td>
<td>Px Partition</td>
<td>Px Partition</td>
<td>Px Partition</td>
</tr>
</tbody>
</table>

Data Node 1
Data Node 2
Data Node 3
Data Node 4
Automatic Data Partitioning

Table T1

Px Partition

P1
P2
P3
P4

Fx Primary Fragment
Fx Secondary Fragment

Data Node 1
Data Node 2
Data Node 3
Data Node 4
Automatic Data Partitioning

Table T1

Px Partition

P1
P2
P3
P4

Fx Primary Fragment
Fx Secondary Fragment

Data Node 1
Data Node 2
Data Node 3
Data Node 4
Automatic Data Partitioning

Table T1

Px Partition
- P1
- P2
- P3
- P4

Fx Primary Fragment
Fx Secondary Fragment

Data Node 1
- F1

Data Node 2
- F3

Data Node 3

Data Node 4
Automatic Data Partitioning

Table T1

<table>
<thead>
<tr>
<th>P1</th>
<th>Partition</th>
</tr>
</thead>
<tbody>
<tr>
<td>P2</td>
<td></td>
</tr>
<tr>
<td>P3</td>
<td></td>
</tr>
<tr>
<td>P4</td>
<td></td>
</tr>
</tbody>
</table>

Fx Primary Fragment
Fx Secondary Fragment

Data Node 1
F1 F3

Data Node 2
F3 F1

Data Node 3

Data Node 4
Automatic Data Partitioning

Table T1

Px  Partition

P1
P2
P3
P4

Fx  Primary Fragment
Fx  Secondary Fragment

Data Node 1
F1  F3

Data Node 2
F3  F1

Data Node 3
F2  F4

Data Node 4
F4  F2
Automatic Data Partitioning

Table T1

Px Partition

P1
P2
P3
P4

Fx Primary Fragment
Fx Secondary Fragment

Data Node 1

Data Node 2

Data Node 3

Data Node 4

Node Group 1

Node Group 2
Automatic Data Partitioning

Table T1

<table>
<thead>
<tr>
<th>Px</th>
<th>Partition</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td></td>
</tr>
<tr>
<td>P2</td>
<td></td>
</tr>
<tr>
<td>P3</td>
<td></td>
</tr>
<tr>
<td>P4</td>
<td></td>
</tr>
</tbody>
</table>

Fx: Primary Fragment
Fx: Secondary Fragment

Data Node 1
- F1
- F3

Node Group 1
- Data Node 2
- F3
- F1

Data Node 3
- F2
- F4

Node Group 2
- Data Node 4
- F4
- F2
On-line Scheduled Maintenance

Scale

MySQL

Cluster

Evolve

Backup

Upgrade
Multi-Data Center Availability
Geographic Replication

- DR and Data Locality
- Replicate complete clusters across data centers
  - Fully active/active
  - No passive resources
- Split individual clusters across data centers
  - Synchronous replication & auto-failover between sites
1.2 Billion UPDATEs per Minute

- NoSQL C++ API, flexaSynch benchmark
- 30 x Intel E5-2600 Intel Servers, 2 socket, 64GB
- ACID Transactions, with Synchronous Replication
Real-World Test Case

Web-Based Content Management System
- JOINs 11-tables, 33.5k rows
- Returns 2k rows, 19 columns per row

Query Execution Time Seconds

MySQL Cluster 7.1: 87.23 seconds
MySQL Cluster 7.2: 1.26 seconds

70x Faster
MySQL Cluster 7.3 EA: Node.js NoSQL API

- Native JavaScript access to MySQL Cluster
  - End-to-End JavaScript: browser to the app and database
  - Storing and retrieving JavaScript objects directly in MySQL Cluster
  - Eliminate SQL transformation

- Implemented as a module for node.js
  - Integrates full Cluster API library within the web app

- Couple high performance, distributed apps, with high performance distributed database
MySQL Cluster 7.3 DMR1: Foreign Keys

- Brings MySQL Cluster to a broader range of workloads
  - Packaged apps, custom projects
- Adds powerful functionality while reducing complexity
  - App logic & data model
- Enabled by default
- Enforced for SQL & NoSQL APIs
- On-line add and drop
MySQL Cluster 7.2 and 7.3: Auto-Installer
Early Access Feature

- Fast configuration
- Workload optimized

Best practices

New!
When to Consider MySQL Cluster

- Scalability demands
  - Sharding for write performance?

- Latency demands
  - Cost of each second?

- Uptime requirements
  - Cost per minute of downtime?
  - Failure versus maintenance?

- Application agility
  - Developer languages and frameworks?
  - SQL or NoSQL?
Next Steps

Learn More
- www.mysql.com/cluster
- Authentic MySQL Curriculum: http://oracle.com/education/mysql

Try it Out
- dev.mysql.com/cluster
- labs.mysql.com
- github.com/mysql/mysql-js

Let us know what you think
- clusterdb.com
- @clusterdb
- forums.mysql.com/list.php?25