What's New in MySQL Cluster 7.2?

Percona Live NYC May 2011

Yves Trudeau

Percona Inc
MySQLPerformanceBlog.com

DIAMOND SPONSORSHIPS

- THANK YOU
- TO OUR
- DIAMOND SPONSORS

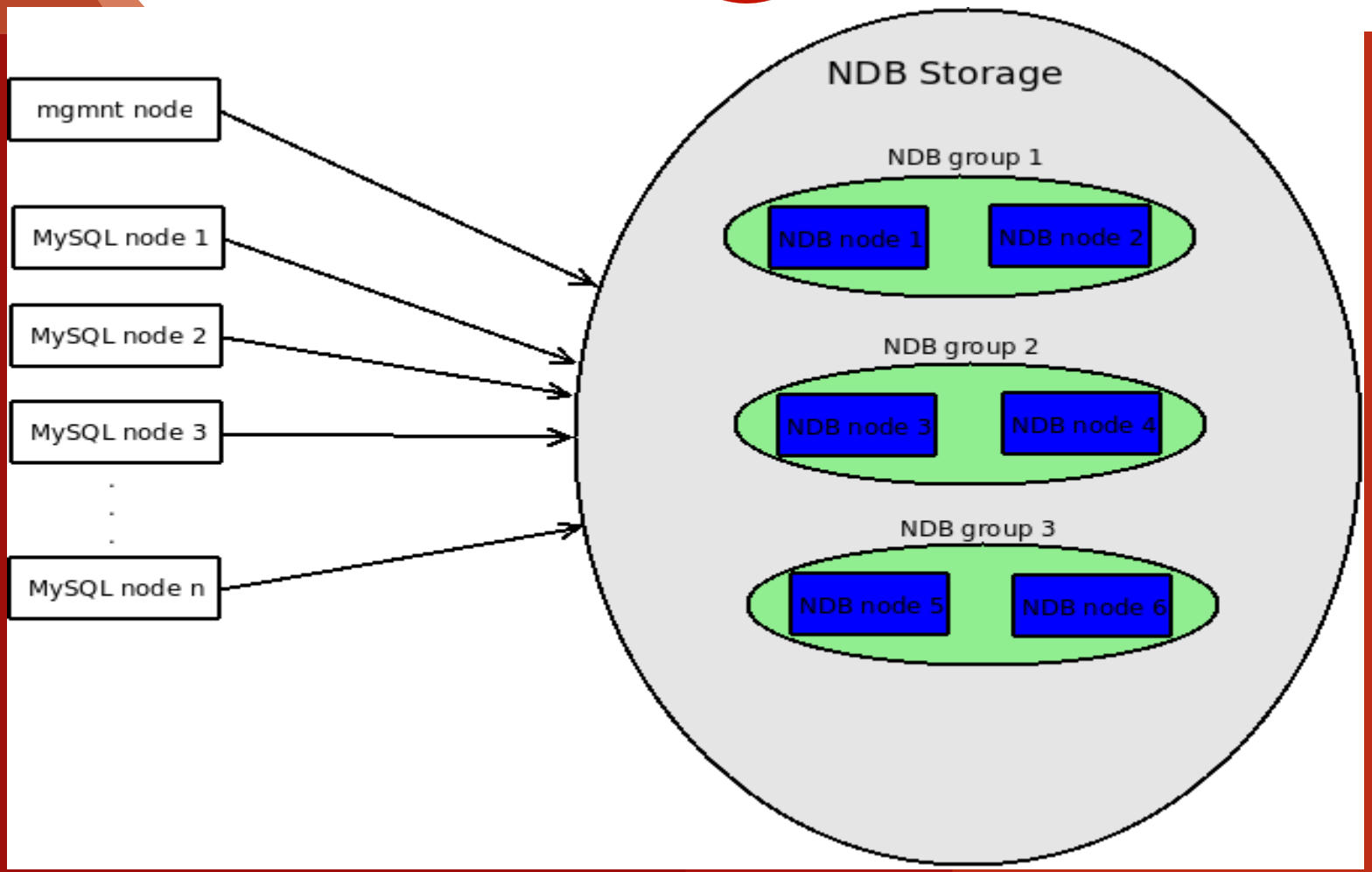
About Me

- http://www.percona.com
- http://www.mysqlperformanceblog.com/
- http://www.bigdbahead.com

Yves Trudeau, Ph.D.

Principal Consultant

NDB Architecture



So, you have tried 5.0.x?

PERCONA

5.0.x

5.1.x

NDB evolution

Since 5.1

- Disk based tables
- Variable size columns in memory
- Custom partitioning
- Online add and drop index
- Replication
- Schema auto discovery

NDB evolution

PERCONA

Since 6.1

- Larger clusters (more nodes)

Since 6.2

- NDB connection pools for MySQL
- Online add column

NDB evolution

PERCONA

Since 6.3

- Compress backup and LCP
- Distribution awareness
- Realtime extensions

NDB evolution

PERCONA

Since 7.0.x

- Compress backup and LCP
- Distribution awareness
- Realtime extensions

NDB evolution

PERCONA

Since 7.1.x

- Ndbinfo schema
- Windows supports

Since 7.2.x:

- Joins pushdown
- MySQL privileges in NDB

NDB evolution

Coming:

- More distributed operations (join types and aggregations)
- Tables replica across nodes
- NDB optimizer stats
- BKA support

Why would you use NDB?

- HA
- Performance
- Sharding
- NoSQL

Sharding, what NDB gives you?

- Distribution awareness
- Scalable with online repartitioning
- Aggregation across partitions
- Cross nodes joins and transactions
- Consistent backup and restore
- Consistent replication

NDB for NoSQL

- As fast as memcached
- Memcached to NDB plugin
- Data can also be accessed by SQL
- Transactions
- Persistence, backups, replication, HA (all for granted!)

NDB gotchas

- Ordered index updates limit performance
- Tuning required for schema, queries, application
- records_in_range call returns 10 (hint required)
- Large unbounded joins are slow (nested loops)
- Read-commited **ONLY**

Join pushdown

How MySQL executes joins (without pushdown)?

```
SELECT e.name, d.deptName, e2.managerName,
t.titleName FROM employee e INNER JOIN employe e2
ON e.managerId = e2.Id INNER JOIN dept d ON
e.deptId = d.Id INNER JOIN title t ON e.titleId =
t.Id
```

```
1. get row e from employee
2. get row from e2 where
e2.id = e.managerId
3. get row from d where d.Id
= e.deptId
4. get row from t where t.Id
= e.titleId
5. output results
6. goto 1.
```

Join pushdow

How MySQL executes join (with pushdows)

- Well... the joins are handles by the data nodes, in parallel
- Join performance scale with number of data nodes
- Less network hops -> schema tuning important
- A leap forward in term of join performance (up to 20x)

Join pushdow

Limitations:

- Only eq_ref and const
- No data type conversion
- No blob columns
- No locks

Questions

Questions?

Yves Trudeau: yves@percona.com

http://www.percona.com/

http://www.mysqlperformanceblog.com/