

# **MySQL<sup>®</sup> RoadMap**

*What we have now &  
Where we are heading*

**Arjen Lentz**  
**Technical Writer / Trainer**

**LCA 2004 Adelaide**  
**January 16, 2004**

**MySQL AB**

# The World's Most Popular Open Source Database



- Founded in 1995
- Operations in 16 countries
- Global “virtual” company
- Over 4,000,000 installations
- 35,000 downloads / day





## Our Brisbane “Office”



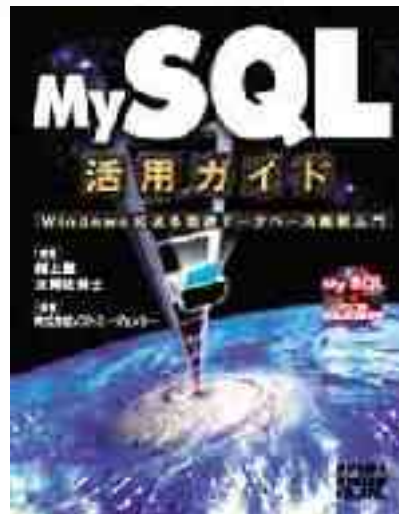
- **MySQL AB is a profitable company**
  - Develops the software in-house; community helps test it
  - Owns source code, copyrights and trademarks
  - Targets the “commoditised” market for databases
- **“Quid Pro Quo” dual licensing**
  - Cost-effective commercial licenses for commercial use
  - Open source GPL license for open source projects

MySQL AB investors  
ABN-AMRO, Benchmark Capital, etc  
are committed to GPL

## MySQL supports its users

- Worldwide 24 x 7 support
- Training & Certification
- Consulting
- Partners

- [bugs.mysql.com](http://bugs.mysql.com)
- [lists.mysql.com](http://lists.mysql.com)
- [www.mysql.com/doc/](http://www.mysql.com/doc/)
- [#mysql](http://irc.freenode.net)
- Lots of books



# MaxDB™ by MySQL

TECHNOLOGY



GLOBAL PARTNER

---

## Release Philosophy - Active Versions

Alpha – Beta – Gamma – Production



We work on 4 different MySQL versions:

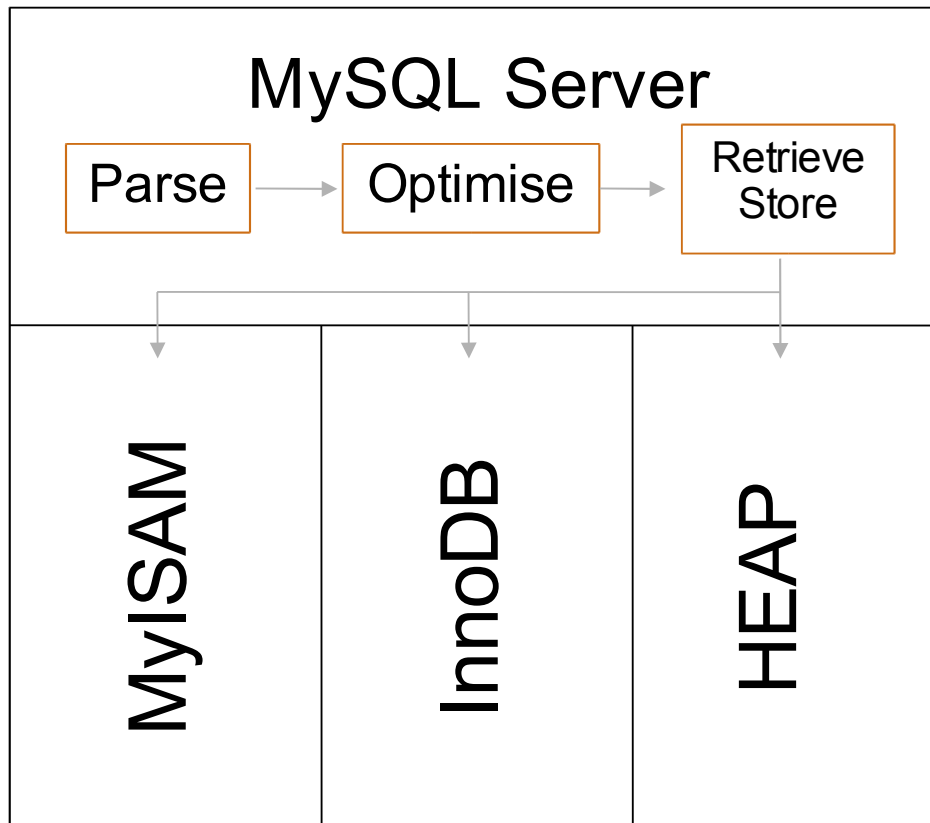
- 3.23 Old – only critical fixes
- 4.0 Production/stable – current release
- 4.1 Alpha – binaries available (beta soon)
- 5.0 Alpha “Preview”

All releases, even alphas, must pass the standard test suite

## Basic Features - Buzzword Bingo

- Extended subset of ANSI SQL standards
- Most platforms including native Windows, 64-bit CPUs
- Extensive API support including ODBC, JDBC, .NET
- Per-table choice of storage engine
- Versioned ACID transactions
- Foreign key constraints, cascading **UPDATE/DELETE**
- Full-Text indexing
- Master-Slave Replication (asynchronous)
- Point-in-time recovery
- User-Defined Functions / Aggregates

# The Storage Engine Concept



**MySQL Database Management Level**

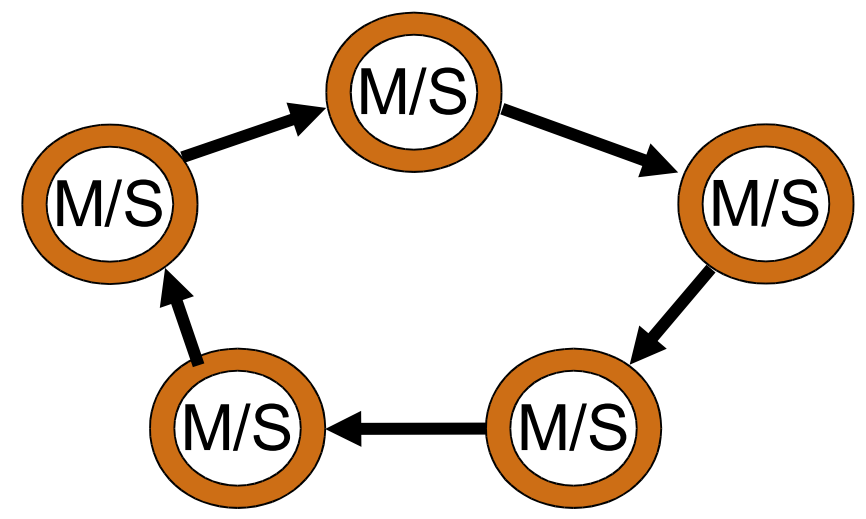
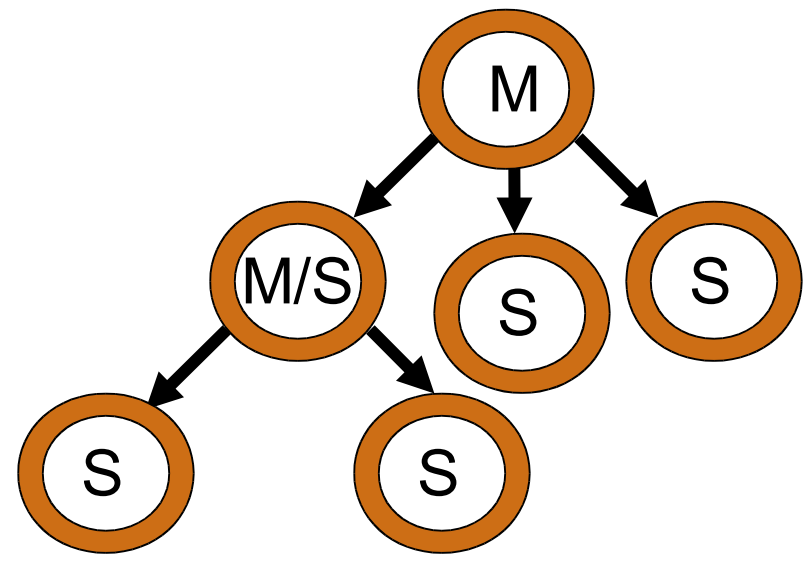
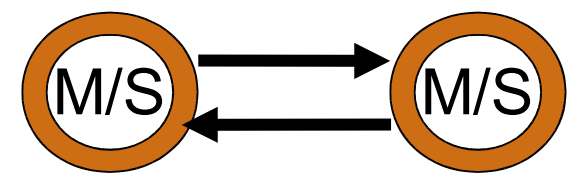
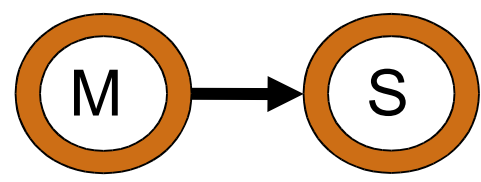
---

**Storage Engine Level**

## InnoDB Storage Engine

- Fully ACID compliant
- Nested transactions with savepoints
- “Oracle style” tablespace, multiple table spaces
- Multi-versioning
- Row-level locking
- All SQL-99 transaction isolation levels
- Next-key locking (no phantoms!)
- Clustered indexes
- Foreign keys with cascading DELETE/UPDATE

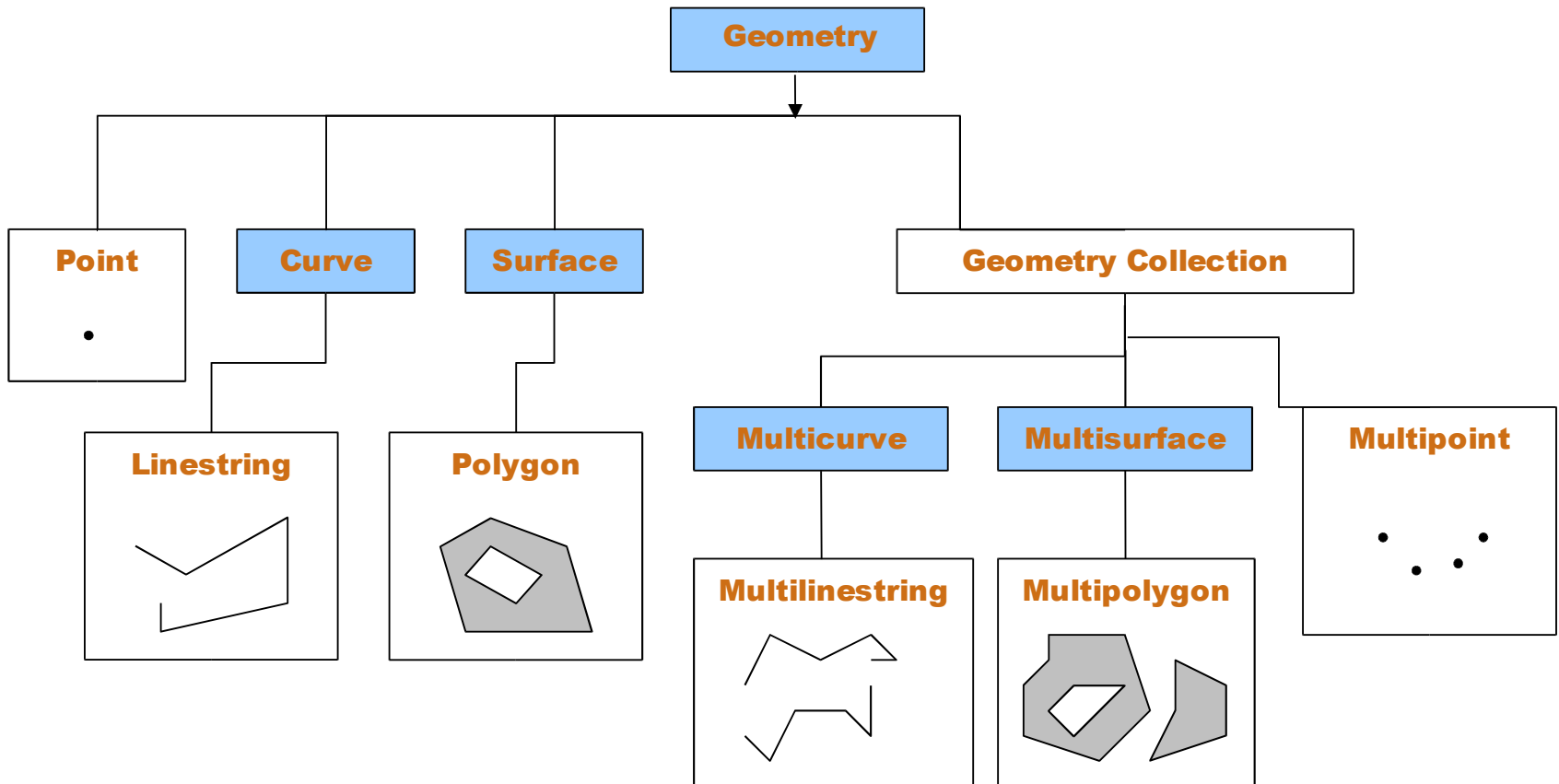
# Replication



## Subqueries (4.1)

```
DELETE FROM t1
WHERE s11 > ANY
  (SELECT COUNT(*) /* no hint */ FROM t2
  WHERE NOT EXISTS
    (SELECT * FROM t3
     WHERE ROW(5*t2.s1, 77) =
      (SELECT 50, 11*s1 FROM t4
       UNION SELECT 50, 77
       FROM (SELECT * FROM t5) AS t5))) ;
```

# OpenGIS Geometric Data (4.1)

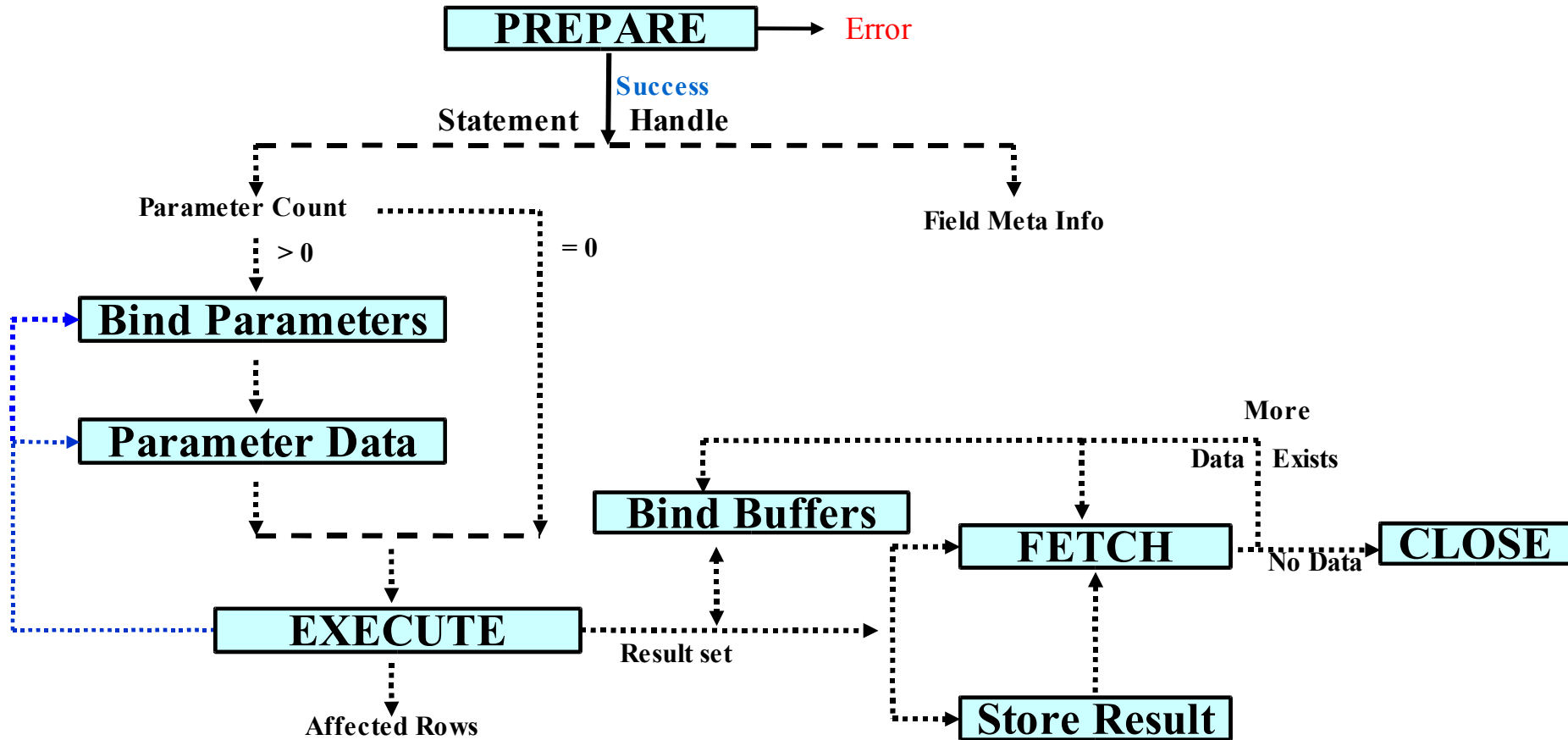


## Unicode Character Sets (4.1)

Official Name	MySQL name	Description
UCS-2	ucs2	16-bit variation
UTF-8	utf8	variable-number-of-bytes variation

```
CREATE TABLE t1 (
  c1 CHAR(10) CHARACTER SET latin1
    COLLATE latin1_german1_ci
)
CHARACTER SET latin2 COLLATE latin2_bin;
```

# Prepared Statements (4.1)



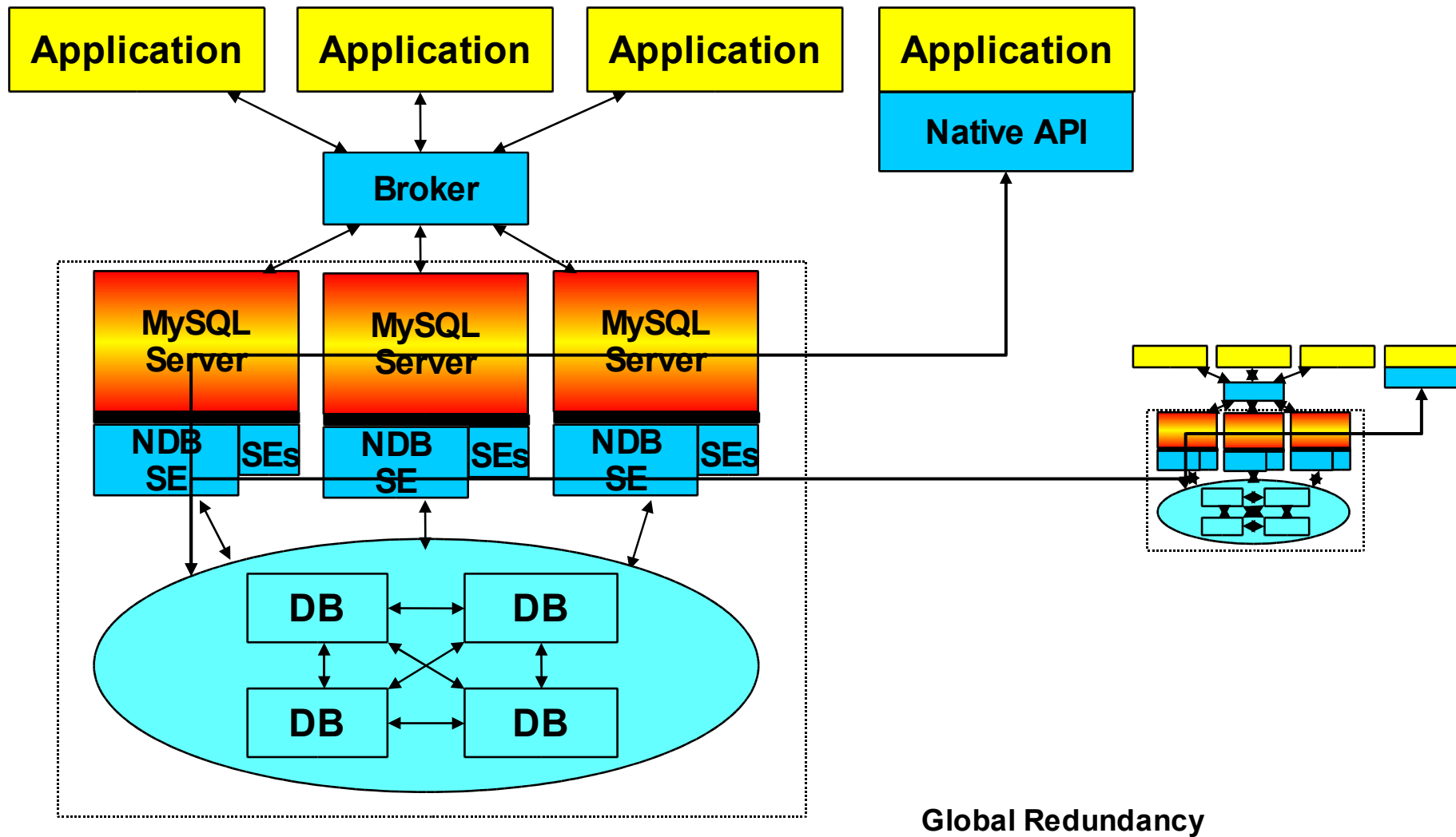
# Stored Procedures & Functions (5.0)

```
CREATE PROCEDURE curdemo ()
BEGIN
    DECLARE done INT DEFAULT 0;
    DECLARE CONTINUE HANDLER FOR SQLSTATE '02000' SET done = 1;
    DECLARE cur1 CURSOR FOR SELECT id,data FROM test.t1;
    DECLARE a CHAR(16);
    DECLARE b INT;
    OPEN cur1;
    REPEAT
        FETCH cur1 INTO a, b;
        IF NOT done THEN
            INSERT INTO test.t3 VALUES (a,b);
        END IF;
    UNTIL done END REPEAT;
    CLOSE cur1;
END
```

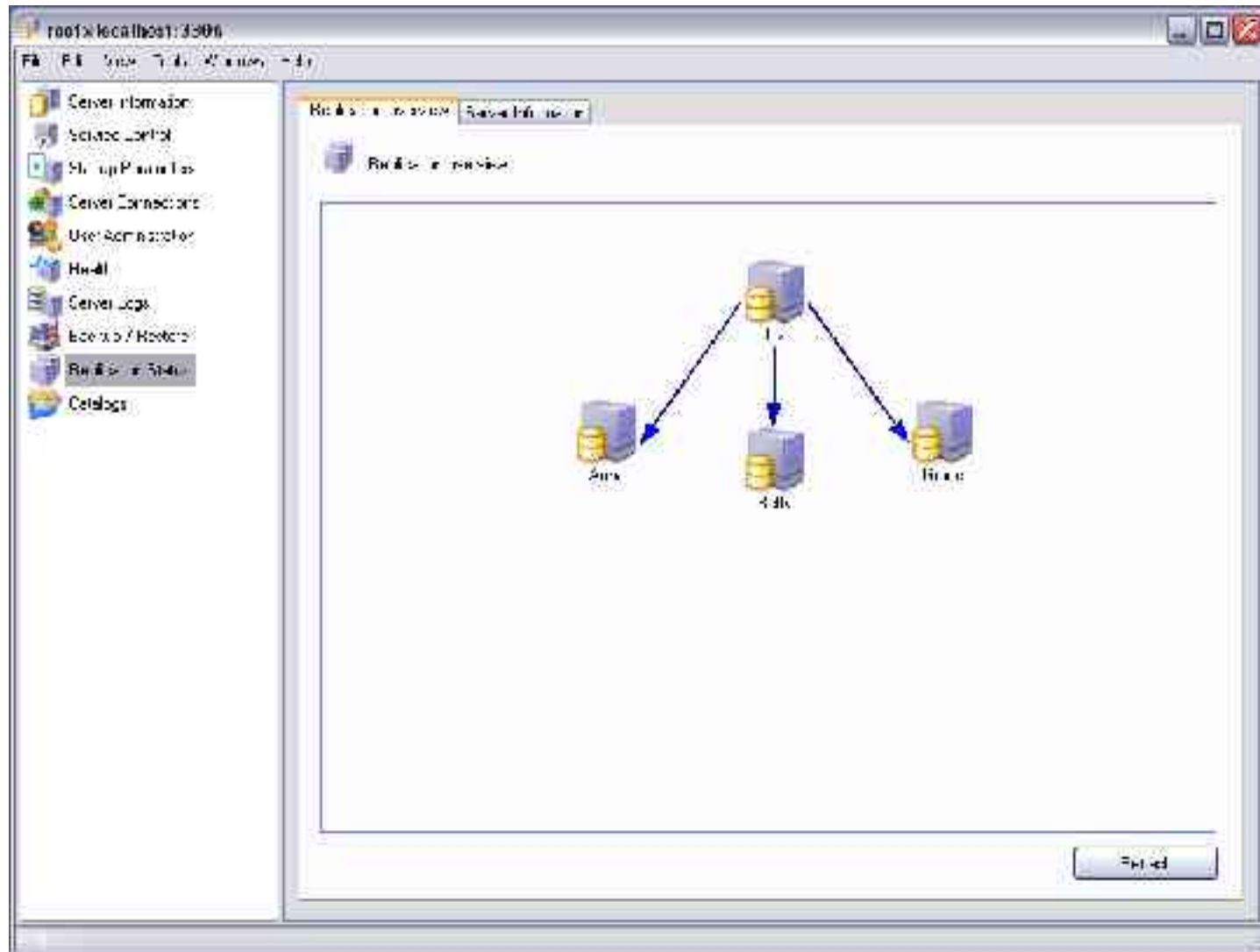
## Next Tuesday?



# NDB Cluster: A New Storage Engine



# Sneak Preview: MySQL Graphical Tools



**Thank you! Questions?**

