

ORACLE®




ORACLE®

PHP and MySQL – The Current State

Johannes Schlüter

MySQL Engineering: Connectors & Client Connectivity



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.

Oracle's Strategy: Complete. Open. Integrated.



- Built together
- Tested together
- Managed together
- Serviced together
- Based on open standards
- Lower cost
- Lower risk
- More reliable

Oracle's Investment in Open Source

- Supported popular open source projects for many years
- Part of Oracle's Complete, Open, Integrated strategy
- Speed up time-to-innovation
- Expand the developer community

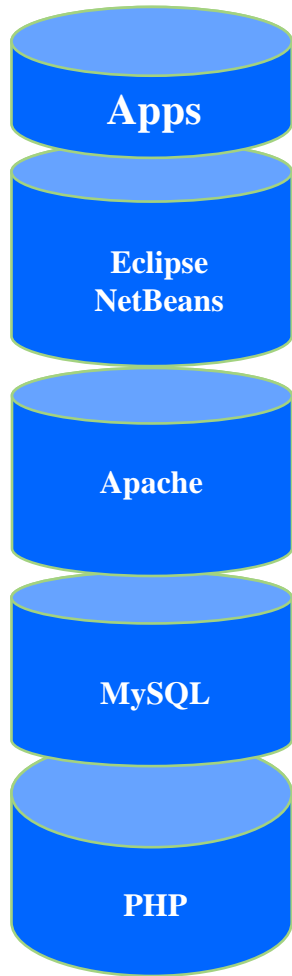




- Oracle never settles for being second best at any level of the stack
- “Complete” means we meet most customer requirements at every level

**That's why MySQL matters
to Oracle and Oracle
customers**

Industry's Most Complete LAMP Stack



- Oracle Enterprise Linux
- Oracle VM (Xen-based)
- Apache
- MySQL
- PHP, Perl, Python

Investment in MySQL

- **Make MySQL a Better MySQL**
 - #1 Open Source Database for Web Applications
- **Develop, Promote and Support MySQL**
 - Improve engineering, consulting and support
 - Leverage 24x7, World-Class Oracle Support
- **MySQL Community Edition**
 - Source and binary releases
 - GPL license



Investment in MySQL

- **MySQL Focus Areas**
 - Web, Embedded & Telecom
 - LAMP
 - Windows
- **Oracle + MySQL Customers**
 - Oracle Enterprise Manager
 - Oracle Secure Backup
 - Oracle Audit Vault



MySQL 5.5

RC

InnoDB will become default

- ACID Transactions, FKs, Crash Recovery

Improved Availability

- Semi-synchronous Replication
- Replication Heartbeat

Improved Usability

- SIGNAL/RESIGNAL
- More Partitioning Options
- PERFORMANCE_SCHEMA

Better Instrumentation/Diagnostics

- InnoDB stats in 5.5 PERFORMANCE_SCHEMA



MySQL 5.5 is Faster!

RC

InnoDB Performance improvements

- Multiple Buffer Pool Instances
- Multiple Rollback Segments
- Extended Change Buffering (with delete buffering, purge buffering)
- Improved Purge Scheduling
- Improved Log Sys mutex
- Separate Flush List mutex

MySQL Performance Improvements

- Better Metadata Locking within Transactions
- Split LOCK_open mutex
- Eliminated LOCK_alarm mutex as bottleneck
- Eliminated LOCK_thread_count as bottleneck
- Improved Performance/Scale on Win32, 64



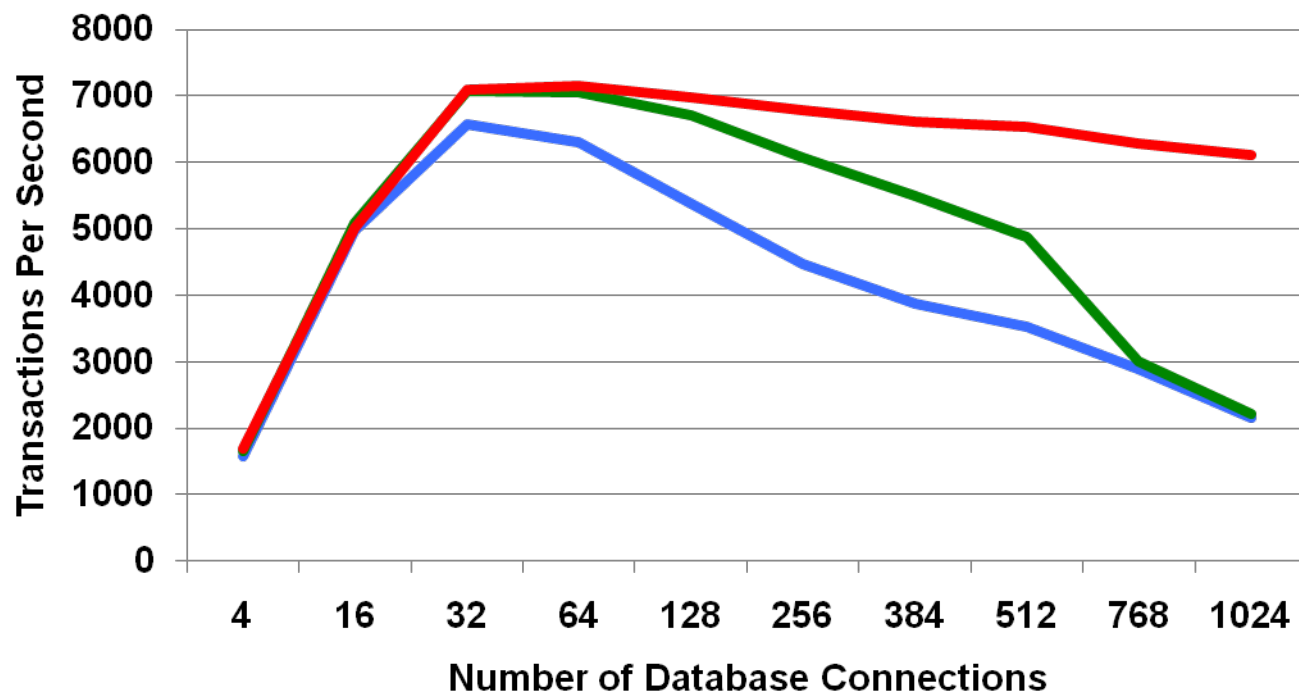
More than 10x improvement in recovery times

MySQL 5.5 SysBench Benchmarks

Linux

RC

MySQL 5.5 vs. 5.1 - Read Only



MySQL 5.5.6
(New InnoDB)

MySQL 5.1.50
(InnoDB Plug-in)

MySQL 5.1.50
(InnoDB built-in)

200% performance gain

for MySQL 5.5 over 5.1.50; at scale

Intel Xeon X7460 x86_64
4 CPU x 6 Cores/CPU
2.66 GHz, 32GB RAM
Fedora 10

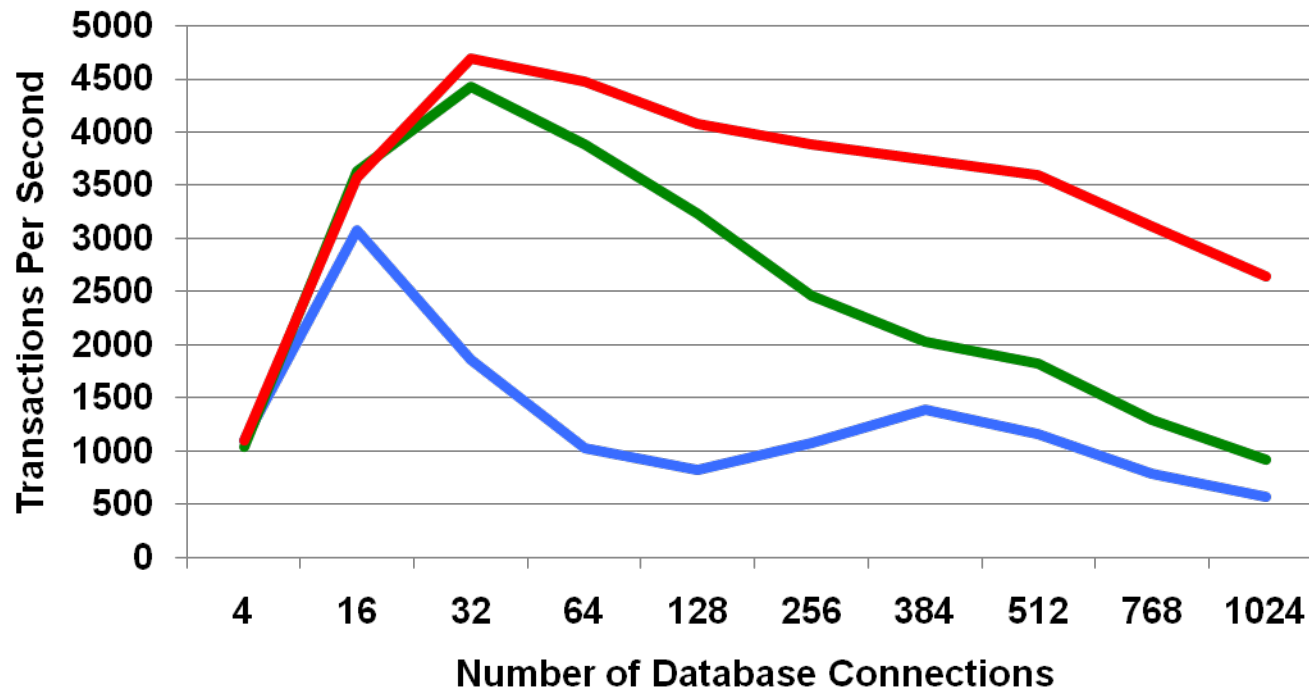
ORACLE

MySQL 5.5 SysBench Benchmarks

Linux

RC

MySQL 5.5 vs. 5.1 - Read Write



MySQL 5.5.6
(New InnoDB)

MySQL 5.1.50
(InnoDB Plug-in)

MySQL 5.1.50
(InnoDB built-in)

369% performance gain

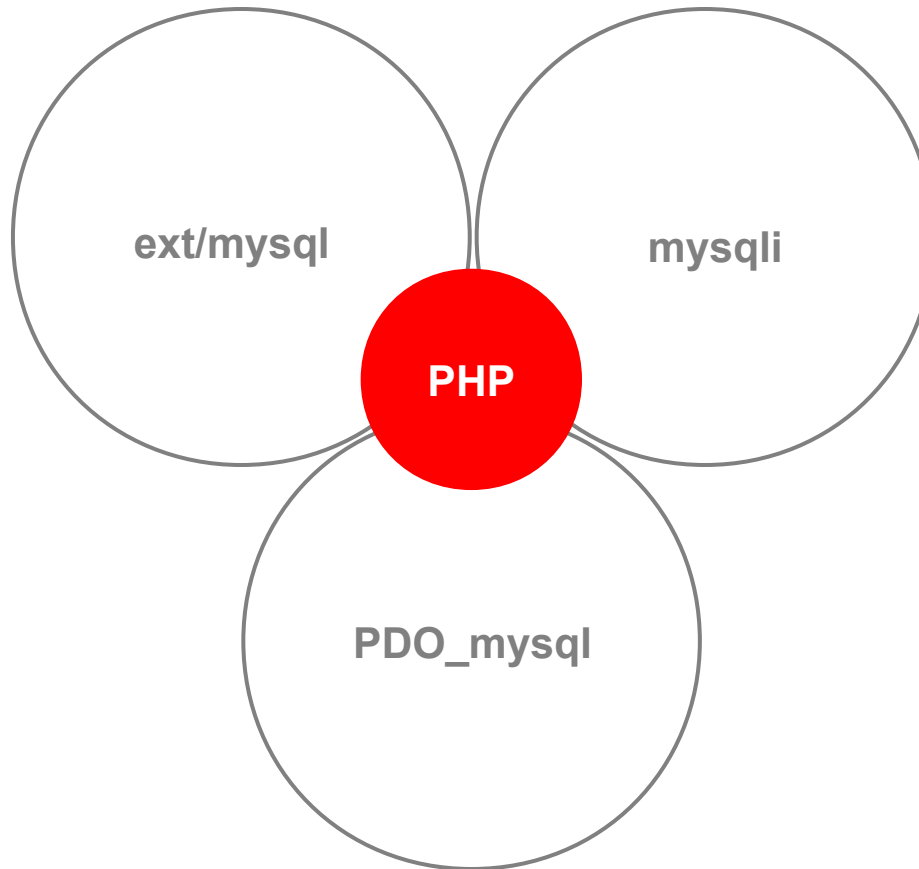
for MySQL 5.5 over 5.1.50; at scale

Intel Xeon X7460 x86_64
4 CPU x 6 Cores/CPU
2.66 GHz, 32GB RAM
Fedora 10

ORACLE



PHP Extensions for MySQL



ext/mysql

- One of the first PHP extensions
- Actively maintained with PHP 4
 - No new features in PHP 5
 - Exception: Added mysqlnd support with PHP 5.3
 - Bug fixing only
- Best documented database extension
 - Tons of books, tutorials, ...
- Missing support for many MySQL features
 - Prepared statements, Queries with multiple result sets (stored procedures), compression, encryption, full charset support, ...

PDO_mysql

- “The PHP Data Objects (PDO) extension defines a lightweight, consistent interface for accessing databases in PHP.” <http://php.net/intro.pdo>
- Lowest common denominator
- PHPish API
- Broken by Design™

PDO – Broken by Design

```
$ php --rf PDO::sqliteCreateFunction
```

Exception: Method PDO::sqliteCreateFunction() does not exist

Intermezzo: Prepared Statements

Client

Server

```
SELECT foo  
FROM bar  
WHERE id = 42
```

query()



- Create Execution plan
- Query database

ResultSet(s)



Intermezzo: Prepared Statements

Client

Server

SELECT foo
FROM bar
WHERE id = ?

prepare()

• Create Execution plan

Handle

Handle
Param 1: 42

execute()

• Query database

ResultSet(s)

PDO – Broken by Design

```
<?php
$pdo = new PDO("mysql:host=localhost;dbname=test",
    "user", "password");

$query = $pdo->prepare(
    "SELECT id FROM table LIMIT ?, ?");
$query->bindValue(1, $_GET["offset"]);
$query->bindValue(2, $_GET["limit"]);

$query->execute();
```

1064 You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near "1', '2"

What The PHP Manual Is Saying

“The *mysqli* extension, or as it is sometimes known, the MySQL **improved** extension, was developed to take advantage of new features found in MySQL systems versions 4.1.3 and newer. [...] If you are using MySQL versions 4.1.3 or later it is **strongly** recommended that you use this extension.”

<http://php.net/mysqli.overview>

**Extended(!) support for
MySQL 4.0(!) ended 2008-09-30**

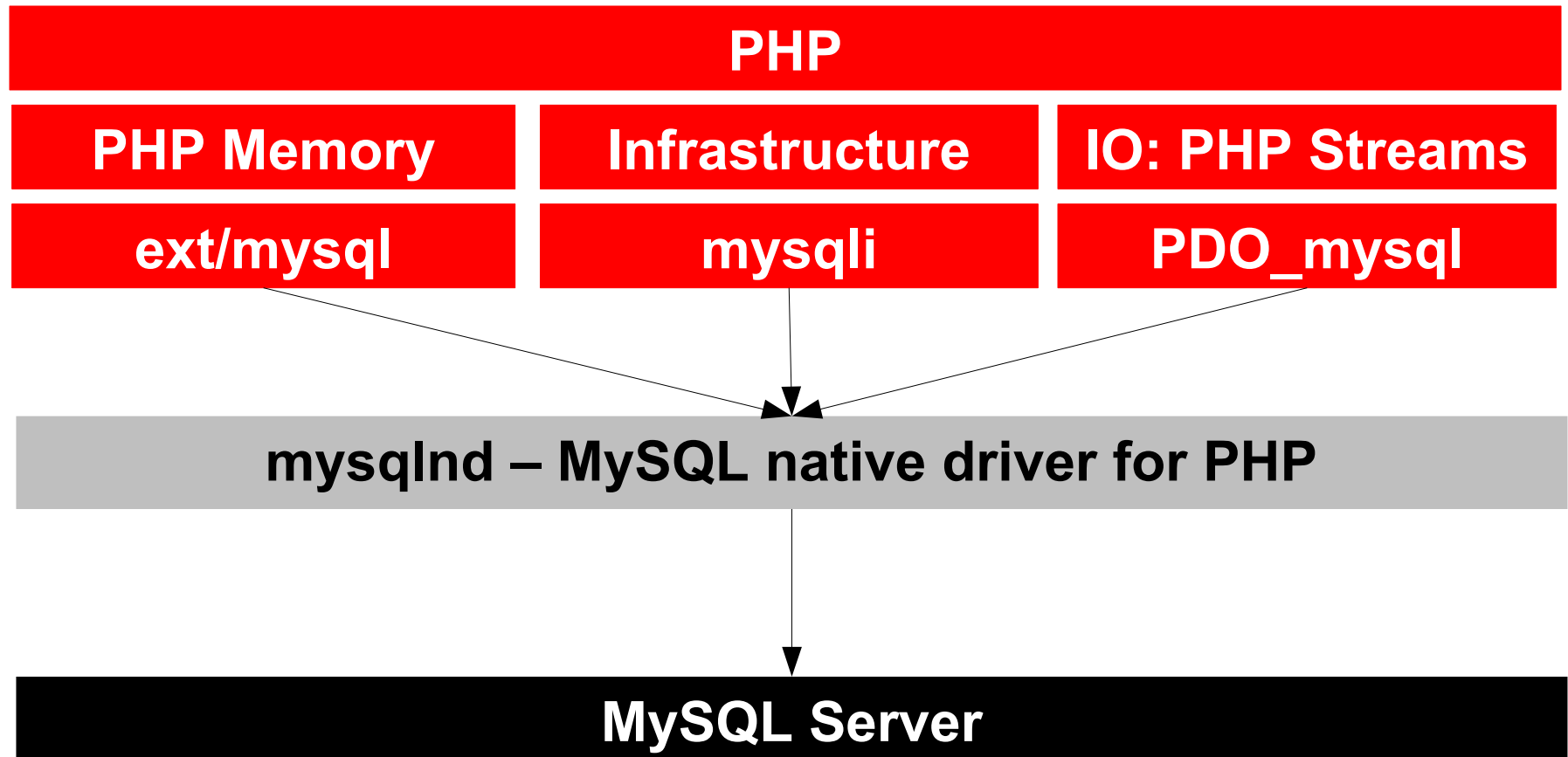


mysqli

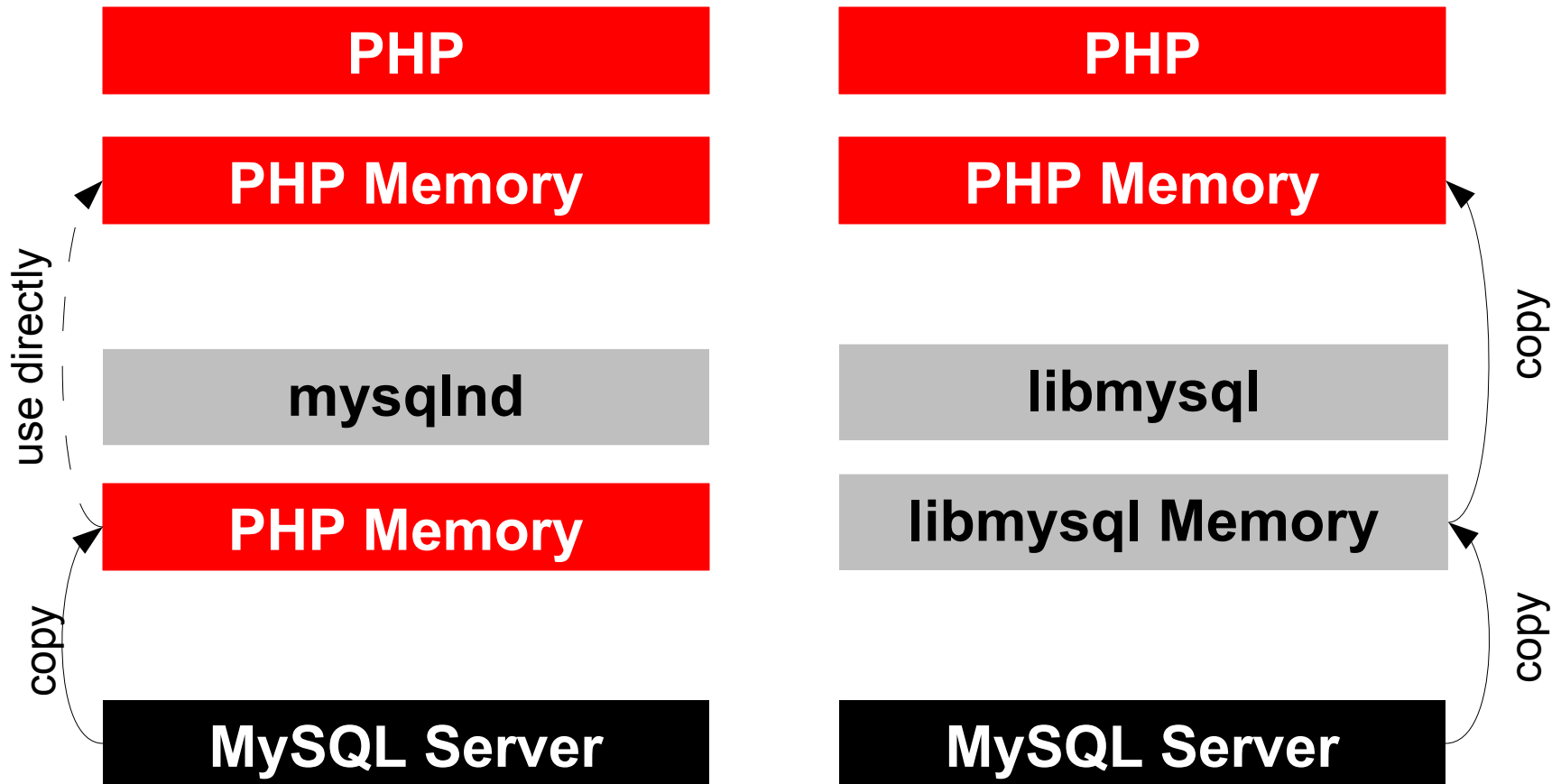
The Improved MySQL Extension

- Full support for all MySQL features
 - Stored Procedures
 - Prepared Statements
 - Encryption (SSL)
 - Compression
 - Charsets
 - ...
- Actively developed, maintained and supported by Oracle

PHP and mysqlnd



libmysql vs. mysqlnd



Building PHP with mysqlnd

- `./configure \`
 `--with-mysql=mysqlnd \`
 `--with-mysqli=mysqlnd \`
 `--with-pdo-mysql=mysqlnd`
- Default on Windows and some distributions

mysqli

Mysqli Support	enabled
Client API library version	mysqlnd 5.0.7-dev - 091210 - \$Revision: 296270 \$
Active Persistent Links	0
Inactive Persistent Links	0
Active Links	9

mysqlnd Statistics

Client statistics	
bytes_sent	5381679
bytes_received	39375881
packets_sent	16117
packets_received	469633
protocol_overhead_in	1878532
protocol_overhead_out	64468
bytes_received_ok_packet	18436
bytes_received_eof_packet	67682
bytes_received_rset_header_packet	68915
bytes_received_rset_field_meta_packet	8416202
bytes_received_rset_row_packet	30690198
bytes_received_prepare_response_packet	0
bytes_received_change_user_packet	0
packets_sent_command	15279
packets_received_ok	1676
packets_received_eof	13522

- Around 150 statistic values collected
- `mysqli_get_client_stats()`,
`mysqli_get_connection_stats()`

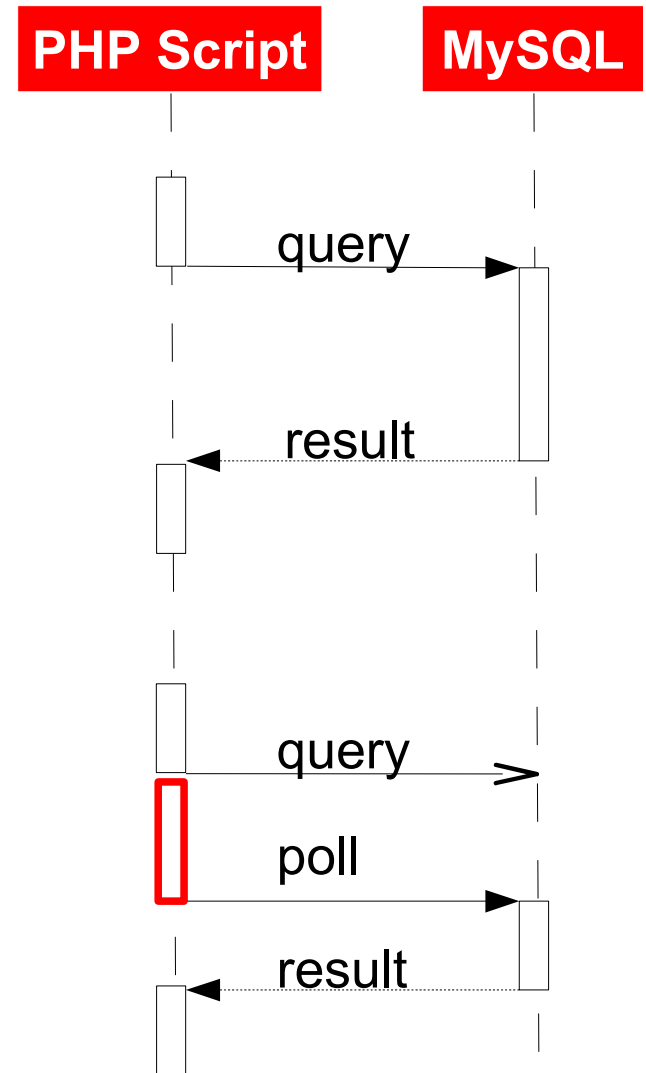
Asynchronous Queries

```
$conn = new MySQLi(...);  
$conn->query(  
    "SELECT * FROM t WHERE ....",  
    MYSQLI_ASYNC);
```

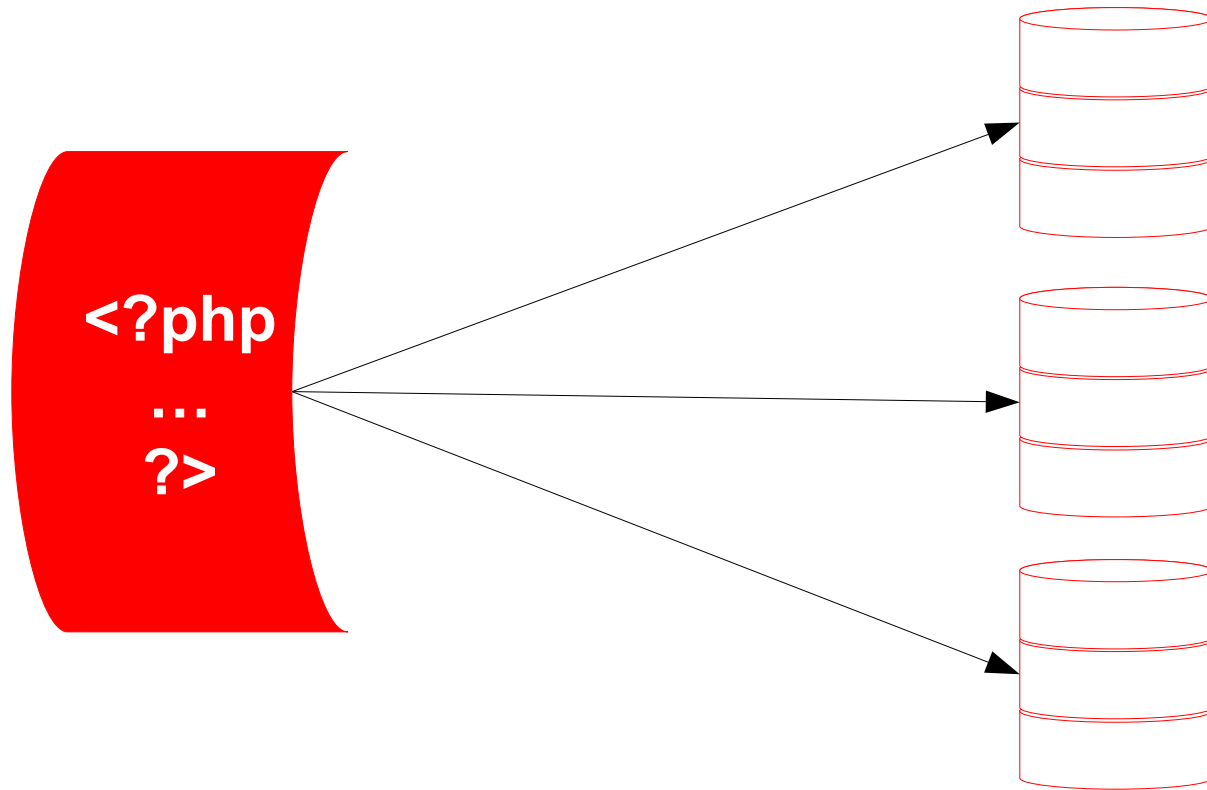
```
/* Do something */
```

```
mysqli_poll($links, $errors, $reject, 1);
```

```
/* Process query results */
```



Sharding



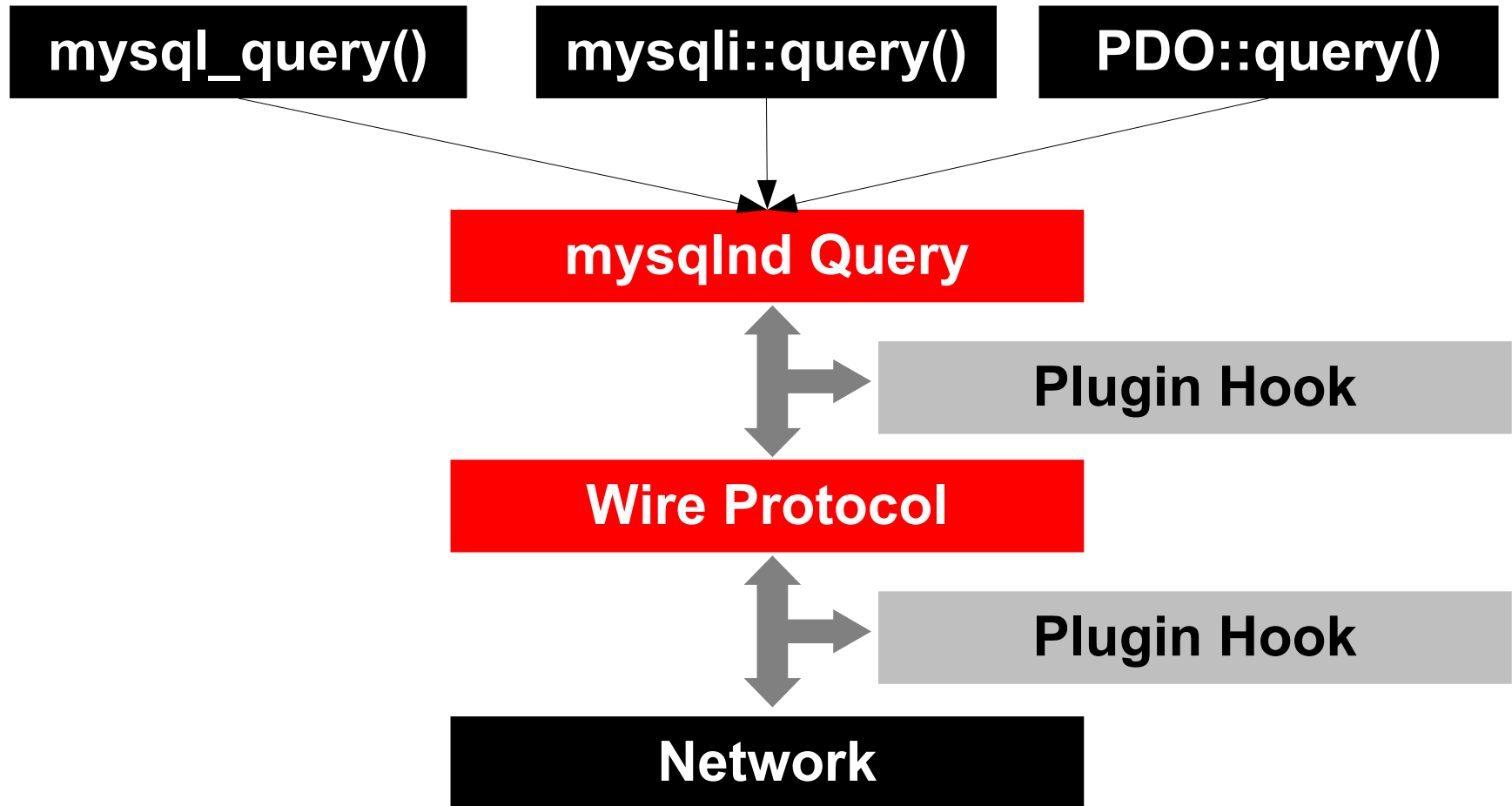
Sharding

```
foreach ($all_links as $link)
    $link->query("SELECT 'test' ", MYSQLI_ASYNC);

$processed = 0;
do {
    $links = $all_links;
    if (!mysqli_poll($links, $errors, $reject, 1)) continue; /* TIMEOUT */

    foreach ($links as $link) {
        if ($result = $link->reap_async_query()) {
            print_r($result->fetch_row());
            mysqli_free_result($result);
            $processed++;
        }
    }
} while ($processed < count($all_links));
```

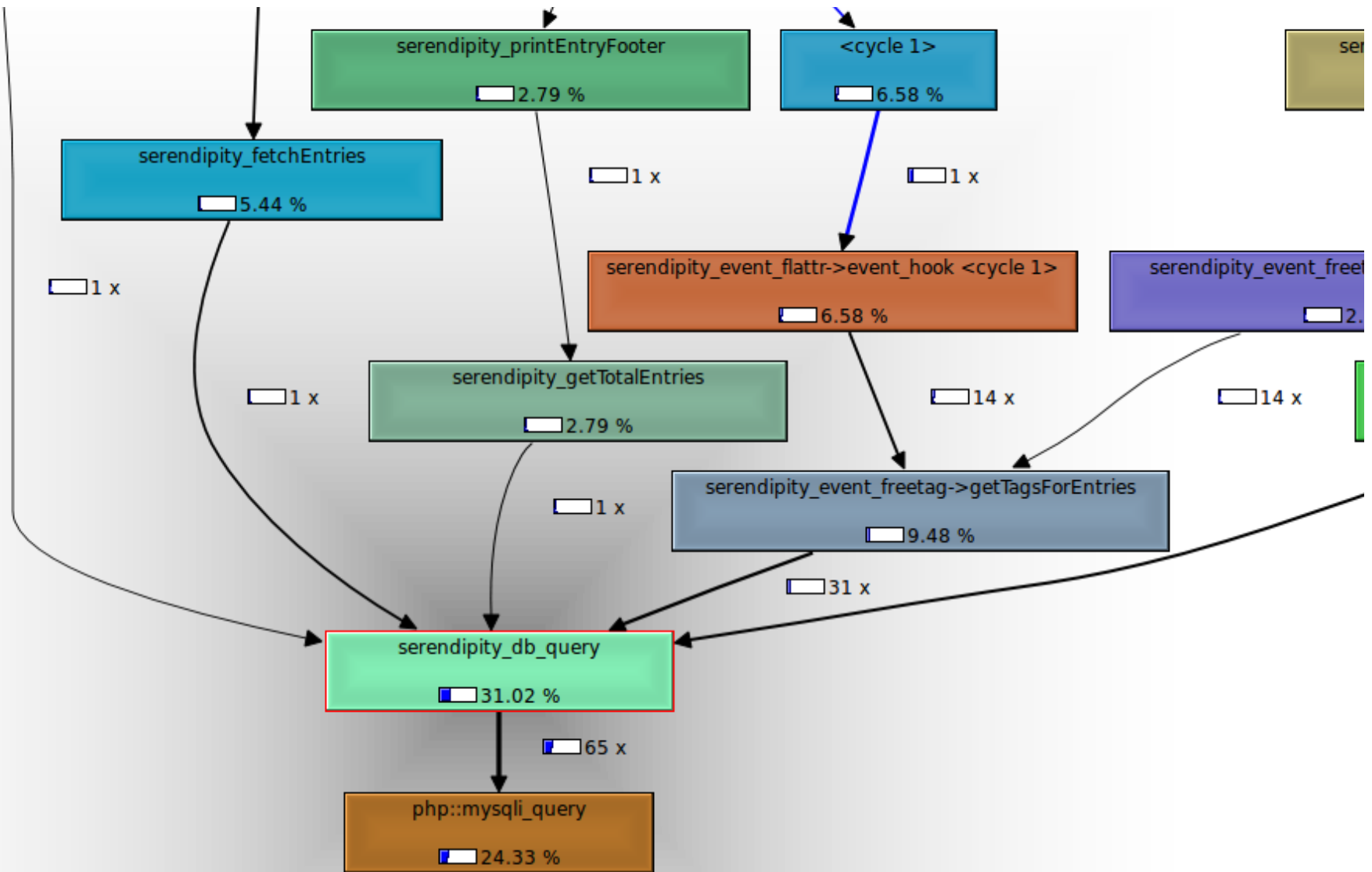
mysqlnd plugins



mysqlnd Plugins

- “mysqlnd client proxy”
 - Load Balancing
 - Read / Write splitting
 - Failover
 - Monitoring
 - Query Logging
 - Query Auditing
 - Performance
 - Caching
 - Sharding

The Database Is The Bottleneck





Caching!

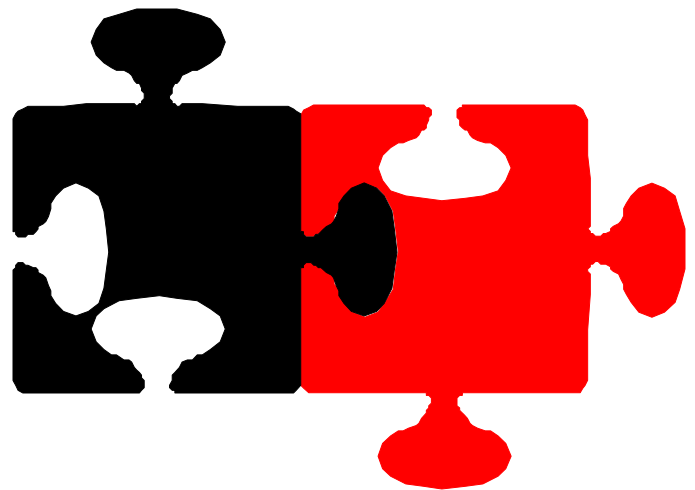


```
# pecl install mysqlnd_qc-beta  
... and you are done!
```

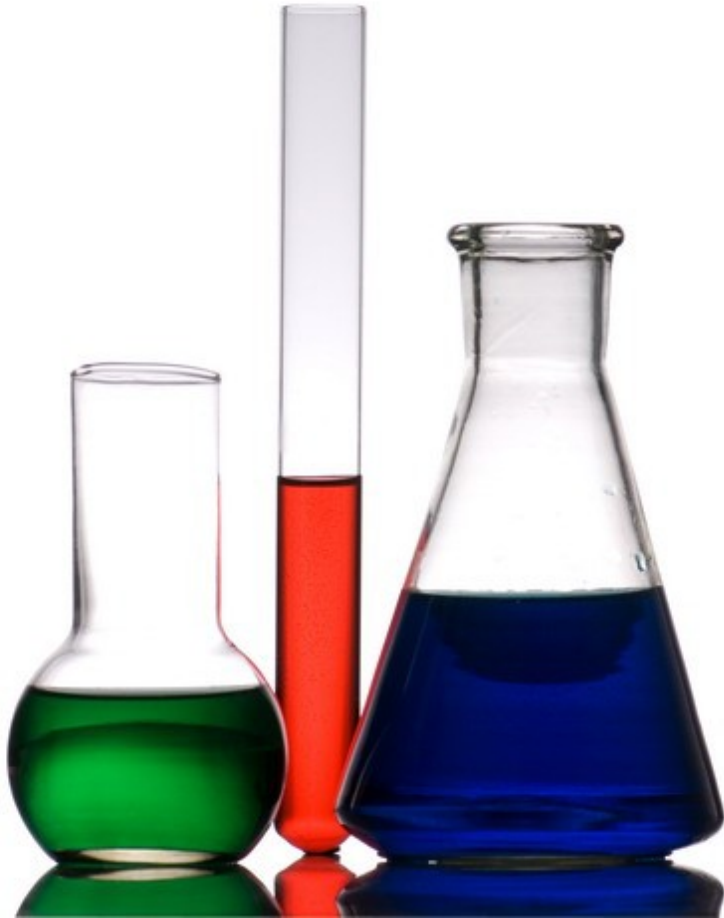
Almost at least ;-)

mysqlnd Query cache

- For more:
- Ulf's presentation → Tomorrow 16:30 (German)



Experimental Extensions



- By Oracle:
 - mysqlnd_sip
 - mysqlnd_mc
 - mysqlnd_ms
 - mysqlnd_pscache
- By Community:
 - mysqlnd_uh
(David Soria Parra / Mayflower GmbH)

mysqlnd_uh

```
class ConnProxy extends MysqlndUHConnection {  
    public function query($conn, $query) {  
        if ($query == "SELECT 1") {  
            $query = "SELECT 2";  
        }  
  
        return parent::query($conn, $query);  
    }  
}
```

```
mysqlnd_uh_set_connection_proxy(new ConnProxy());
```

Key Takeaways

- MySQL is important to Oracle and our customers
 - Part of our Complete, Open, Integrated strategy
- Oracle is making MySQL better **today**
 - Major Feature, Performance, Scalability enhancements
 - 24x7, Global support in 145 countries

Download Now

<http://dev.mysql.com/downloads>



SOFTWARE. HARDWARE. COMPLETE.

ORACLE®