



# MySQL 開発最新動向

## MySQL 8.0、MySQL InnoDB Clusterなどのご紹介

updated : 2017/06/16

Yoshiaki Yamasaki / 山崎 由章  
MySQL Senior Sales Consultant, Asia Pacific and Japan

## Safe Harbor Statement

以下の事項は、弊社の一般的な製品の方向性に関する概要を説明するものです。また、情報提供を唯一の目的とするものであり、いかなる契約にも組み込むことはできません。以下の事項は、マテリアルやコード、機能を提供することをコミットメントするものではない為、購買決定を行う際の判断材料になさらないで下さい。

オラクル製品に関して記載されている機能の開発、リリースおよび時期については、弊社の裁量により決定されます。

# アジェンダ

- 1 ➤ Oracle MySQL Cloud Service
- 2 ➤ MySQL 8.0 DMR 新機能
- 3 ➤ MySQL Group Replication、MySQL InnoDB Cluster
- 4 ➤ MySQL Enterprise Edition

# アジェンダ

- 1 ▶ Oracle MySQL Cloud Service
- 2 ▶ MySQL 8.0 DMR 新機能
- 3 ▶ MySQL Group Replication、MySQL InnoDB Cluster
- 4 ▶ MySQL Enterprise Edition

# Oracle MySQLクラウドサービスによる TCOの最適化



# Oracle MySQLクラウドサービス

**The World's Most Popular Open Source Database in the Oracle Cloud**



Oracle MySQL Cloud Service は、  
迅速,安全且つコスト効率良く  
MySQLをデプロイする事が可能。

# MySQL Cloud Service: 価値提案



- シンプル
  - わずか数回のクリックで、素早くMySQLデータベース・インスタンスが利用可能。
- 自動化
  - データベース管理を自動化するツールで簡単にMySQLを管理する事が可能。
- 統合
  - 迅速な開発と展開の為に、Oracleクラウドサービスとの統合
- エンタープライズ対応
  - パフォーマンス、セキュリティ&アップタイム用のOracleの実証済みのMySQLエンタープライズ・エディションを標準提供。



Oracle MySQL Cloud Service

Services

Activity

SSH Access

Welcome!

Summary

2

インスタンス

2

OCPU

15<sup>GB</sup>

メモリー

170<sup>GB</sup>

記憶域

2

パブリックIP

インスタンス

インスタンス名別検索



2016/08/12 1時13分33秒 UTC現在

インスタンスの作成



JAPAC-PreSales

Status: Creating service ...

Subscription: Hourly

バージョン: 5.7.13

エディション: Enterprise Edition

送信日: 2016/08/12 1時13分23秒 UTC

OCPU: 1

メモリー: 7.5 GB

ストレージ: 85 GB

STEP1)

“インスタンスの作成”をクリック



carsten-db1

Subscription: Hourly

バージョン: 5.7.13

エディション: Enterprise Edition

作成日: 2016/08/11 15時01分20秒 UTC

OCPU: 1

メモリー: 7.5 GB

ストレージ: 85 GB

▶ インスタンス作成および削除履歴





## Service Configuration

\* Service Name JAPACSC01 ?

Service Description New Release Evaluation ?

\* SSH Public Key ssh-rsa AAAAB3NzaC1yc2EAA 編集 ?

\* Compute Shape OC3 - 1.0 OCPU, 7.5GB RAM ?



## Configuration

\* Usable Database Storage (GB) 25 ?

\* Administration User root ?

\* Administration Password ..... ?

\* Confirm Administration Password ..... ?

\* Database Schema Name JAPAC ?

Configure Enterprise Monitor Yes ?

\* Manager User mem\_admin ?

\* Manager Password ..... ?

\* Confirm Manager Password ..... ?

\* Agent User mem\_agent ?

\* Agent Password ..... ?

\* Confirm Agent Password ..... ?



## Backup and Recovery Configuration

Backup Destination Both Cloud and Disk Storage ?

\* Cloud Storage Container Storage-mysqlsc/JAPACSC01 ?

\* Cloud Storage User Name shinya.sugiyama@oracle.com ?

\* Cloud Storage Password ..... ?

Create Cloud Storage Container ☒ ?

## STEP2)

ホスト名を入力しカタログからサーバータイプを選択し作成。必要に応じてObject Storage, MySQL Enterprise Monitorの設定を入力し完了。

作成時間: 約10分

## Restart Service Completed

Service Name: JAPAC-PreSales  
Operation: Restart Service  
Status: Succeeded

Start Time: 2016/08/12 7時46分39秒 UTC

End Time: 2016/08/12 7時52分37秒 UTC

## Create Service Completed

Service Name: JAPAC-PreSales  
Operation: Create Service  
Status: Succeeded

Start Time: 2016/08/12 1時13分23秒 UTC

End Time: 2016/08/12 1時21分01秒 UTC

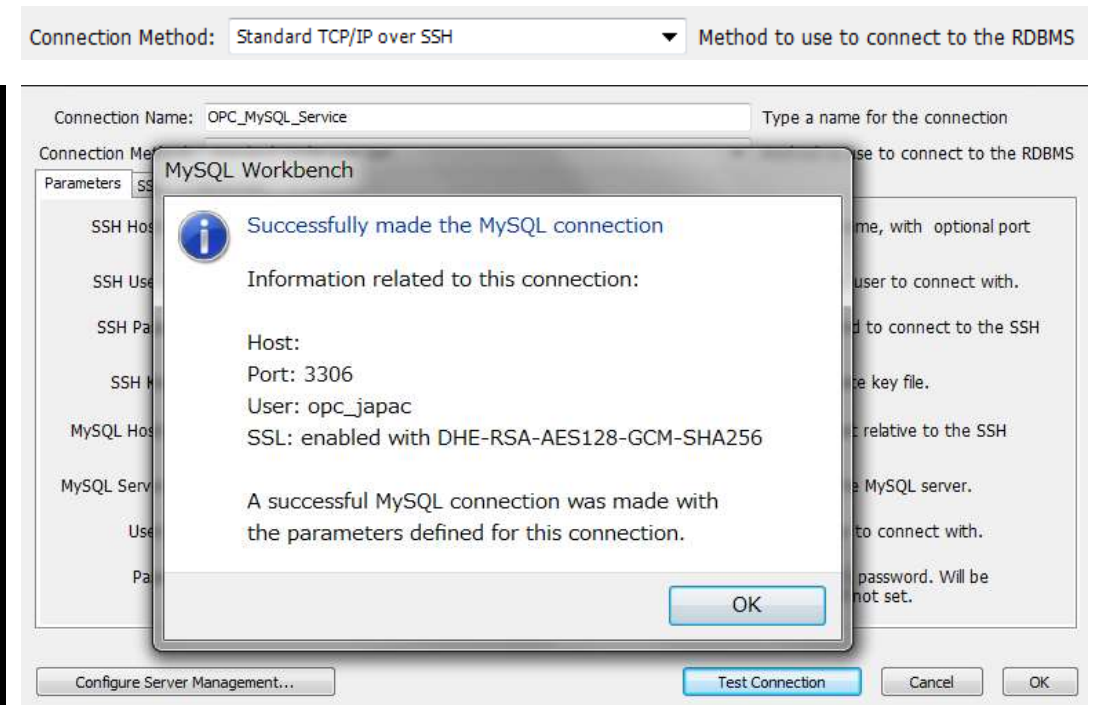
### STEP3)

Public IPが設定されているので、アサインされたIPに対して鍵認証でログインする事が可能。

## SSHを利用した接続

```
*****
*                               Welcome to                               *
*                               MySQL Cloud Service                       *
*                               by                                         *
*                               Oracle                                     *
*                               If you are an unauthorised user please disconnect IMMEDIATELY *
***** MySQL Information *****
* Status:  RUNNING *
* Version:  5.7.13  *
*****
***** Storage Volume Information *****
* Volume      Used      Use%      Available  Size  Mounted on *
* MySQLlog    6.3G    -----  34%         13G   20G   /u01/translog *
* bin         2.6G    -----  28%         6.7G   9.8G   /u01/bin *
* data        151M    --  1%         24G   25G   /u01/data *
*****
[opc@japac-presales-mysql-1 ~]$
```

## Workbench経由でのSSH接続



```
mysql> select PLUGIN_NAME,PLUGIN_STATUS,PLUGIN_TYPE,LOAD_OPTION from PLUGINS
-> where PLUGIN_TYPE <> 'INFORMATION SCHEMA';
```

PLUGIN_NAME	PLUGIN_STATUS	PLUGIN_TYPE	LOAD_OPTION
binlog	ACTIVE	STORAGE ENGINE	FORCE
mysql_native_password	ACTIVE	AUTHENTICATION	FORCE
sha256_password	ACTIVE	AUTHENTICATION	FORCE
InnoDB	ACTIVE	STORAGE ENGINE	FORCE
PERFORMANCE_SCHEMA	ACTIVE	STORAGE ENGINE	FORCE
MRG_MYISAM	ACTIVE	STORAGE ENGINE	FORCE
MyISAM	ACTIVE	STORAGE ENGINE	FORCE
MEMORY	ACTIVE	STORAGE ENGINE	FORCE
CSV	ACTIVE	STORAGE ENGINE	FORCE
BLACKHOLE	DISABLED	STORAGE ENGINE	OFF
partition	ACTIVE	STORAGE ENGINE	ON
FEDERATED	DISABLED	STORAGE ENGINE	OFF
ARCHIVE	DISABLED	STORAGE ENGINE	OFF
ngram	ACTIVE	FTPARSER	ON
audit_log	ACTIVE	AUDIT	FORCE_PLUS_PERMANENT
thread_pool	ACTIVE	DAEMON	ON
authentication_pam	ACTIVE	AUTHENTICATION	ON
auth_socket	ACTIVE	AUTHENTICATION	ON
validate_password	ACTIVE	VALIDATE PASSWORD	ON

MySQL Enterprise版のバイナリーがインストール済みの為、Enterprise版の機能が利用する事が可能。

2016/08/30 6時57分34秒 UTC現在

Summary	0 MB	0 MB	0 %
	Storage Cloud Volume Used	Backup Volume Used	Backup Volume Percent Used
Daily at 16時15分00秒 UTC	Every Tuesday at 16時15分00秒 UTC	Not Available	
Incremental Backups	Full Backups	Last Successful Backup	

MySQL Enterprise Backupも実装されていて、Dashboardからバックアップジョブ設定、Point In Timeリカバリー含めて管理可能。

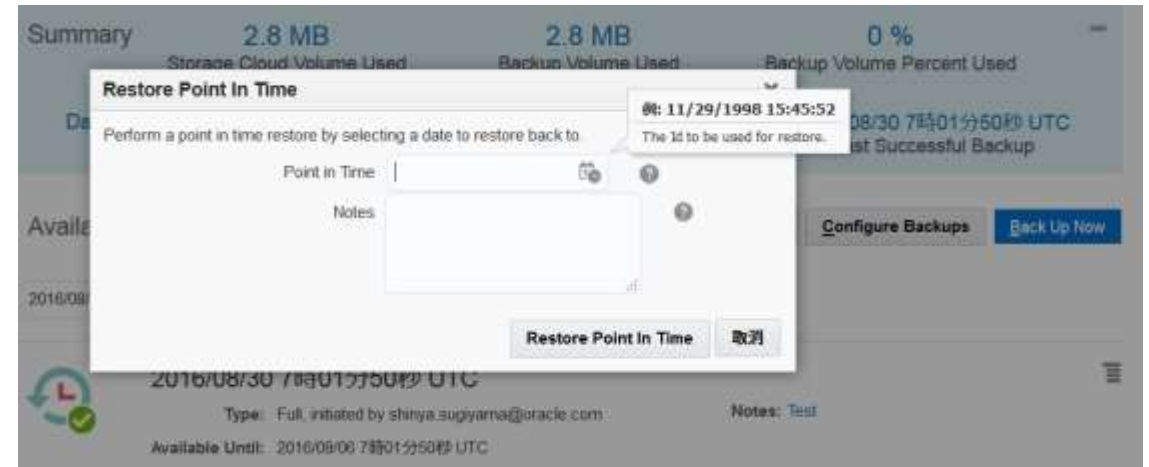
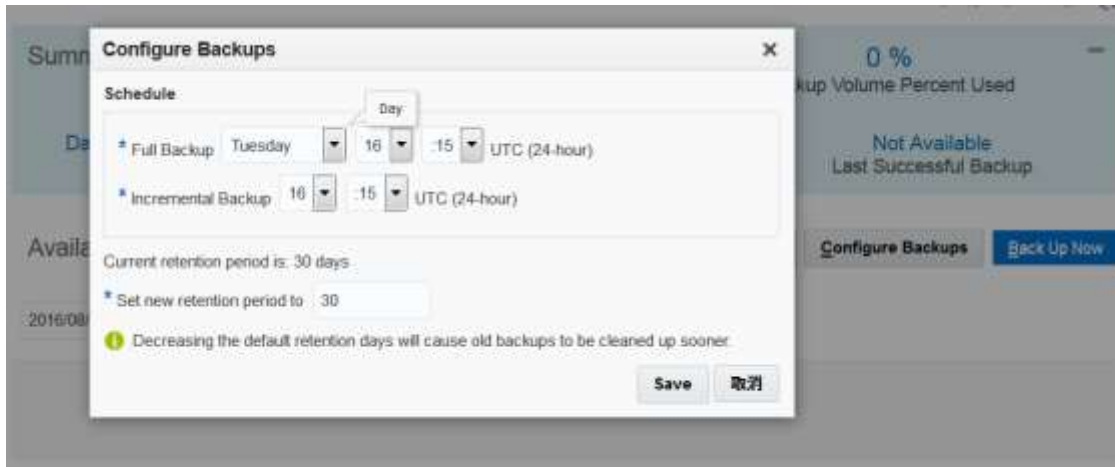
## Available Backups

[Restore Point In Time](#)[Configure Backups](#)[Back Up Now](#)

2016/08/23 to Enter end date

*No backups available.*

## ► Restore History



クエリを参照

Graph for last 30 分 (JST) Edit

データベースの活動状況 - データベースの活動状況 - すべてのMySQLインスタンス (Aggr...)

Zoom: 1h 2h 4h 6h 12h 1d 2d

データベースの活動状況 - すべてのMySQLインスタンス

Statements / Seco...

21:30 21:45

✓ Select (SUM) ✓ Insert (SUM) ✓ Update (SUM) ✓ Replace (SUM) ✓ Delete (SUM) ✓ Call (SUM)

MySQL Enterprise Monitorも利用可能  
MySQLの設定、パフォーマンス、クエリー等  
を一元管理する事が可能です。

Show 10 entries データのエクスポートオプション...

Showing 1 to 10 of 716 entries First Previous 1 2 3 4 5 Next Last

クエリ	データベース		カウント			QRTi	待ち時間	
			実行	エラー	警告		合計	最高
COMMIT (1)	mem		21,853	0	0	1.00	8:55.113	0.94
INSERT INTO `mem_quan`...`timestamp`, VALUES ( (1)	mem		9,982	0	0	1.00	2:12.311	1.04
INSERT INTO `mem_quan`...TYPE = VALUES ( TYPE ) (1)	mem		819	108	0	0.84	2:05.992	51.40
INSERT INTO `mem_quan`...en` ) ), `lastSeen` ) (1)	mem		10,695	713	0	1.00	1:53.685	0.70
ROLLBACK (1)	mem		2,086	0	0	1.00	14.800	0.21



Overview

1 Node

Administration

Patching To 5.7.13.002

Aug 9, 2016 5:49:09 PM UTC

Last Successful Backup

Backup Patching

Available Patches

As of Aug 9, 2016 5:51:06 PM UTC

Quarterly Update 5.7.13.002

Release Date: Apr 29, 2016 1:05:00 PM UTC

Requires Restart: No

Readme

Precheck summary

Quarterly Update 5.7.13.001

Release Date: Apr 29, 2016 1:05:00 PM UTC

Requires Restart: Yes

Readme

Patch and Rollback History

Overview

1 Node

Administration

0 Patches available

Aug 9, 2016 5:49:09 PM UTC

Last Successful Backup

Backup Patching

Available Patches

As of Aug 9, 2016 5:52:05 PM UTC

No patches available.

Patch and Rollback History

5.7.13.002

Patched By: weblogic on Aug 9, 2016 5:49:08 PM UTC

Notes: Apply patch 5.7.13.002

Readme

Roll Back

Dashboardから、MySQLのパッチ適用、適用前の事前検証と適用後のロールバックを実施する事が可能  
※適用前にMEBでバックアップが自動取得されます。



最高レベルのセキュリティ



スケーラビリティと可用性



MySQLエキスパート  
テクニカルサポート



Oracleクラウド環境へ統合



ハイブリッドにデプロイ可能  
クラウド& オンプレミス



TCOの削減

# MySQL Cloud Service: ビジネス上のメリット



- **ビジネスの俊敏性を向上:**  
イノベーションにリソースを集中し、迅速に最新のアプリケーションを提供。
- **確実なセキュリティ, パフォーマンス, 稼働時間:**  
ソースレベルから、最も包括的なMySQL Cloud プラットホームを利用する事が可能。
- **TCO (総所有コスト) を削減:**  
稼働時間を向上させながら、インフラストラクチャ及びデータベース管理操作コストを節約可能。

REST APIを利用して、自動化する事も可能です。

<http://docs.oracle.com/cloud/latest/stcompute/cs/STCSA/QuickStart.html>

※ MySQL ServiceのRESTマニュアルは現在、準備中です。



# MySQL Cloud Service Pricing

課金方法は2つオプション

- Metered (実際に利用した分のみのコスト負担)
- No-Metered (月単位での固定課金)

Japanese Yen

Buy Now

Product	Price	Metric
MySQL Cloud Service - Metered	¥ 28	OCPU / Hour
MySQL Cloud Service - Non-metered	¥ 15,600	OCPU / Month

1 OCPU = 2 vCPU

```
[root@japacsc01-mysql-1 opc]# cat /proc/cpuinfo | grep processor
processor      : 0
processor      : 1
[root@japacsc01-mysql-1 opc]#
```

詳細: [https://cloud.oracle.com/en\\_US/mysql/pricing](https://cloud.oracle.com/en_US/mysql/pricing)

# 詳細情報 @ cloud.oracle.com/mysql



The screenshot shows the Oracle Cloud MySQL landing page. At the top, there's a navigation bar with 'ORACLE Cloud' on the left, and links for 'Sign In', 'English', and a 'Free Trial' button on the right. Below the navigation bar, there's a green banner with the MySQL logo and a 'Try It' button. Underneath the banner, there's a navigation menu with 'Overview', 'Features', 'Pricing', and 'Learn More'. The main content area has a light blue background with the text 'MySQL in the Oracle Cloud for Your Enterprise Needs.' and a description: 'The world's most popular open source database powered by the Oracle Cloud, delivering a secure, cost-effective and enterprise-grade MySQL database service for your modern applications.' To the right of the text is a large MySQL logo with a dolphin. Below the text is a 'Watch Video' button. At the bottom, there are four columns of text: 'Simple.' (Quickly provision MySQL database instances with just a few clicks.), 'Automated.' (Database management made easy with tools that automate administrative tasks.), 'Integrated.' (Integrated with Oracle Cloud Services for quick development and deployment.), and 'Enterprise Ready.' (Oracle's proven MySQL Enterprise Edition delivers performance, security and uptime to address your enterprise needs.).

Sign up today for a free trial @  
<https://cloud.oracle.com/mysql>

# アジェンダ

- 1 ▶ Oracle MySQL Cloud Service
- 2 ▶ MySQL 8.0 DMR 新機能
- 3 ▶ MySQL Group Replication、MySQL InnoDB Cluster
- 4 ▶ MySQL Enterprise Edition

# MySQL 8.0

# MySQL 5.7 – 全面的な改良

全部で200以上の改善！！

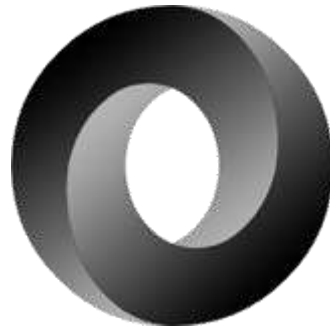
- Replication
- InnoDB
- Optimizer
- Security
- Performance Schema
- GIS

- Triggers
- Partitioning
- **New!** SYS Schema
- **New!** JSON
- Performance

# The MySQL 5.7 Story (*cont.*)



Mobile



Schemaless  
Database



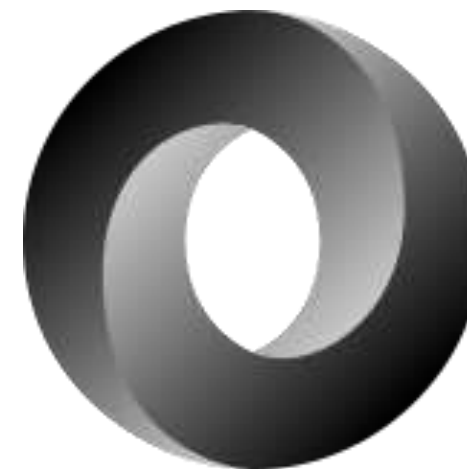
Improved  
MySQL

# MySQL 8.0 DMR (開発途上版)

- 2017年5月時点の最新版はMySQL 8.0.1
- フィードバック募集中
- バグ報告や機能追加要望はこちらから
  - MySQL Bugs  
<https://bugs.mysql.com/>

# JSON対応、MySQL Shellの拡張

- MySQL ドキュメントストア
  - ドキュメントストアとリレーショナル・テーブルを自由に組み合わせ可能
- MySQL Shell を使ってMySQLを管理
  - MySQLのためのワンストップDevOpsツール
  - ユーザーの好きな言語を選択可能 : SQL, JavaScript, Python, ...
  - リレーショナル or ドキュメント
  - MySQL InnoDB Clusterを素早く構築可能






# GIS機能の拡張



- Geography サポート
  - st\_distance() 関数
- Spatial Reference Systems (SRS) サポート
- SQL/MM インフォメーションスキーマビュー
- 標準に準拠したaxis order(緯度、経度の指定順)を使用したimport/export関数
- 演算、データ変換に役立つ各種のSpatial関数
  - st\_x(geom, x)
  - st\_y(geom, y)
  - st\_srid(geom, srid)

# UTF-8

- 最新のUnicode 9.0をサポート
- デフォルトキャラクタセットをutf8mb4に変更
  - デフォルト照合順序はutf8mb4\_0900\_ai\_ci
- 国ごとの照合順序を実装
  - utf8mb4\_ja\_0900\_as\_csを実装済み 
  - ハハ、パパ問題を解消可能
  - as: アクセントセンシティブ(アクセント、濁音、破裂音の区別)
  - cs: ケースセンシティブ(大文字、小文字の区別)
  - utf8mb4\_ja\_0900\_as\_cs\_ksなどの追加を検討中
    - Ks: カナセンシティブ('あ'、'ア'、'ァ'を区別)



## 補足: フィードバックのお願い

- バグ報告、機能追加リクエストがあると、「実際にユーザーさんがどんな機能を使っているのか?」、「どんな使い方をしているのか?」といったことが開発者に伝わるので、とても有益です
- MySQL 8.0での絵文字を区別する実装も、以下のバグ報告をきっかけにして実装されました
  - Bug#76553: Sushi-Beer issue of MySQL with utf8mb4  
<https://bugs.mysql.com/bug.php?id=76553>
- 日本語に適した照合順序など、日本語に依存した機能の開発方向性を決めるうえで、日本のユーザーの皆さんのフィードバックはとても重要です

# UUID とビット処理の改善

- UUID とバイナリデータを変換する関数を追加
  - UUID\_TO\_BIN()
  - BIN\_TO\_UUID()
  - 上記に加え IS\_UUID()
- バイナリデータ型に対してもビット処理が可能に
  - 5.7まではBIGINTに対してのみビット処理が可能
  - 8.0では、VARBINARYやBLOBに対してもビット処理可能
- UUIDとIPv6関連の処理を意識
  - UUID、IPv6用のデータ型はないが、VARBINARY(16)を使用可能
  - INET6\_ATON(address) & INET6\_ATON(network)

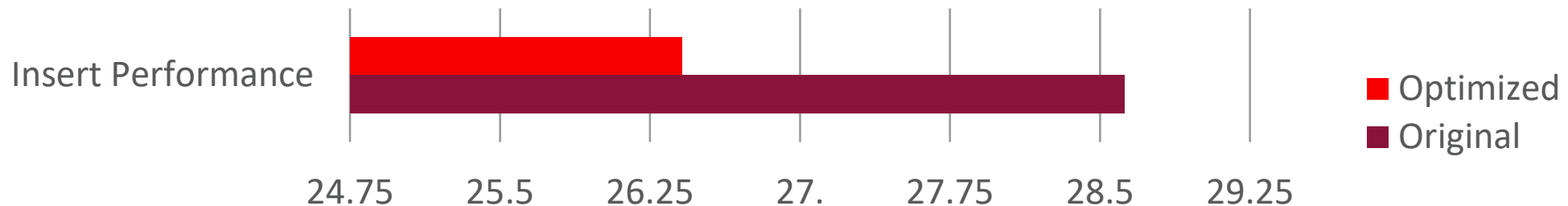


# UUID\_TO\_BIN の最適化

- フォーマットを変更し、INSERT効率向上
- サイズも縮小

From VARCHAR(36) 53303f87-78fe-11e6-a477-8c89a52c4f3b

To VARBINARY(16) 11e678fe53303f87a4778c89a52c4f3b



# SELECT... FOR UPDATE の拡張



```
SELECT seat_no  
FROM seats  
JOIN seat_rows USING ( row_no )  
WHERE seat_no IN (3,4)  
AND seat_rows.row_no IN (12)  
AND booked = 'NO'  
FOR UPDATE OF seats SKIP LOCKED  
FOR SHARE OF seat_rows NOWAIT;
```

行が既にロックされてい  
れば、その行に対する  
ロック取得はあきらめる

行が既にロックされ  
ていれば、直ぐに  
エラーを返す

# 新しいヒントの追加

- JOIN順序を制御するヒントを追加
- MySQL 5.7から追加されたオプティマイザヒントの構文を使用
  - 構文: `SELECT /*+ ヒント名 */ FROM`
  - コメントとしてヒントを埋め込める
- STRAIGHT\_JOINヒントの置き換えだけでなく、拡張されたヒントも追加
  - `JOIN_FIXED_ORDER`     `/* replacement for STRAIGHT_JOIN*/`
  - `JOIN_ORDER`             `/* use specified order */`
  - `JOIN_PREFIX`            `/* use specified order for first tables */`
  - `JOIN_SUFFIX`            `/* use specified order for last tables */`

# 降順索引 (Descending Indexes)

## For B+tree indexes

```
CREATE TABLE t1 (  
  a INT,  
  b INT,  
  INDEX a_b (a DESC, b ASC)  
);
```

- In 5.7: 索引は昇順で作成される。降順でSELECTした時は索引を後ろ向きにスキャンする
- In 8.0: 索引を降順でも作成可能。降順でSELECTした時は索引を前向きにスキャンする

### 利点:

- 高速 (索引は後ろ向きよりも前向きにスキャンしたほうが高速)
- ファイルソートを避けるために索引を使用できるケースの拡大



# 不可視索引 (Invisible Indexes)

- オプティマイザーから見えない索引
  - 索引の無効化とは異なる
  - データ更新時に不可視索引は更新される
- 2つのユースケース:
  - 仮削除 (ゴミ箱)
  - 索引採用のテスト



# ユースケース1: 索引の仮削除(ゴミ箱)

## 使用例

- 索引の仮削除

```
ALTER TABLE Country ALTER INDEX c INVISIBLE;
```

- 索引の復旧

```
ALTER TABLE Country ALTER INDEX c VISIBLE;
```

- 索引の削除

```
ALTER TABLE Country DROP INDEX c;
```

## ユースケース2: 索引採用のテスト

- 新しい索引作成は、既存の実行計画を変化させる可能性があるためリスクを伴う
- 不可視索引は索引採用のテスト容易にする
  - 準備段階で不可視索引を作成し、必要に応じて一時的に有効化しながら索引の有効性を確認する
  - 上手く動くことが確認出来てから索引を永続的に有効化する
  - 現時点では、クエリー単位で不可視索引を有効化する機能は無いが、ユーザーからの要望が多く集まれば機能追加を検討している(※次ページ参照)

```
ALTER TABLE Country ADD INDEX c (Continent) INVISIBLE;  
# after some time  
ALTER TABLE Country ALTER INDEX c VISIBLE;
```

# クエリー単位で不可視索引を有効化する機能について

- 機能追加要望をBug#83066で登録済みです
- Bug #83066
  - Feature Request : Use Invisible Indexes Specific Query
  - <http://bugs.mysql.com/bug.php?id=83066>
- 機能追加を希望する方は、“Affects me”をクリックして下さい！

# 不可視索引の確認方法

```
SELECT * FROM information_schema.statistics WHERE is_visible='NO' ;  
***** 1. row *****
```

```
TABLE CATALOG: def  
TABLE SCHEMA: world  
TABLE NAME: Country  
NON UNIQUE: 1  
INDEX SCHEMA: world  
INDEX NAME: c  
SEQ IN INDEX: 1  
COLUMN NAME: Continent  
COLLATION: A  
CARDINALITY: 7  
SUB PART: NULL  
PACKED: NULL  
NULLABLE:  
INDEX TYPE: BTREE  
COMMENT: disabled  
INDEX COMMENT:  
IS_VISIBLE: NO
```

# 仮削除の候補選定に役立つsys.schema\_unused\_indexes

- sys.schema\_unused\_indexesから、使用していないインデックスを確認可能
  - DB起動後にアクセスがあったテーブルに付けられているインデックスが対象
  - 再起動後の稼働期間が十分でない場合は、この情報を確認後、不可視索引を使って安全にインデックスを削除可能

# 仮削除の候補選定に役立つsys.schema\_unused\_indexes

```
mysql> SELECT * FROM sys.schema_unused_indexes;  
Empty set (0.00 sec)
```

```
mysql> use world;  
Reading table information for completion of table and column names  
You can turn off this feature to get a quicker startup with -A
```

Database changed

```
mysql> SELECT * FROM sys.schema_unused_indexes;  
+-----+-----+-----+  
| object_schema | object_name      | index_name  |  
+-----+-----+-----+  
| world         | City             | CountryCode |  
| world         | CountryLanguage | CountryCode |  
+-----+-----+-----+  
2 rows in set (0.00 sec)
```

# 仮削除の候補選定に役立つsys.schema\_unused\_indexes

```
mysql> SELECT count(*) FROM world.City WHERE CountryCode='JPN';
```

```
+-----+  
| count(*) |  
+-----+  
|      248 |  
+-----+
```

```
1 row in set (0.00 sec)
```

```
mysql> SELECT * FROM sys.schema_unused_indexes;
```

```
+-----+-----+-----+  
| object_schema | object_name      | index_name      |  
+-----+-----+-----+  
| world         | CountryLanguage | CountryCode     |  
+-----+-----+-----+
```

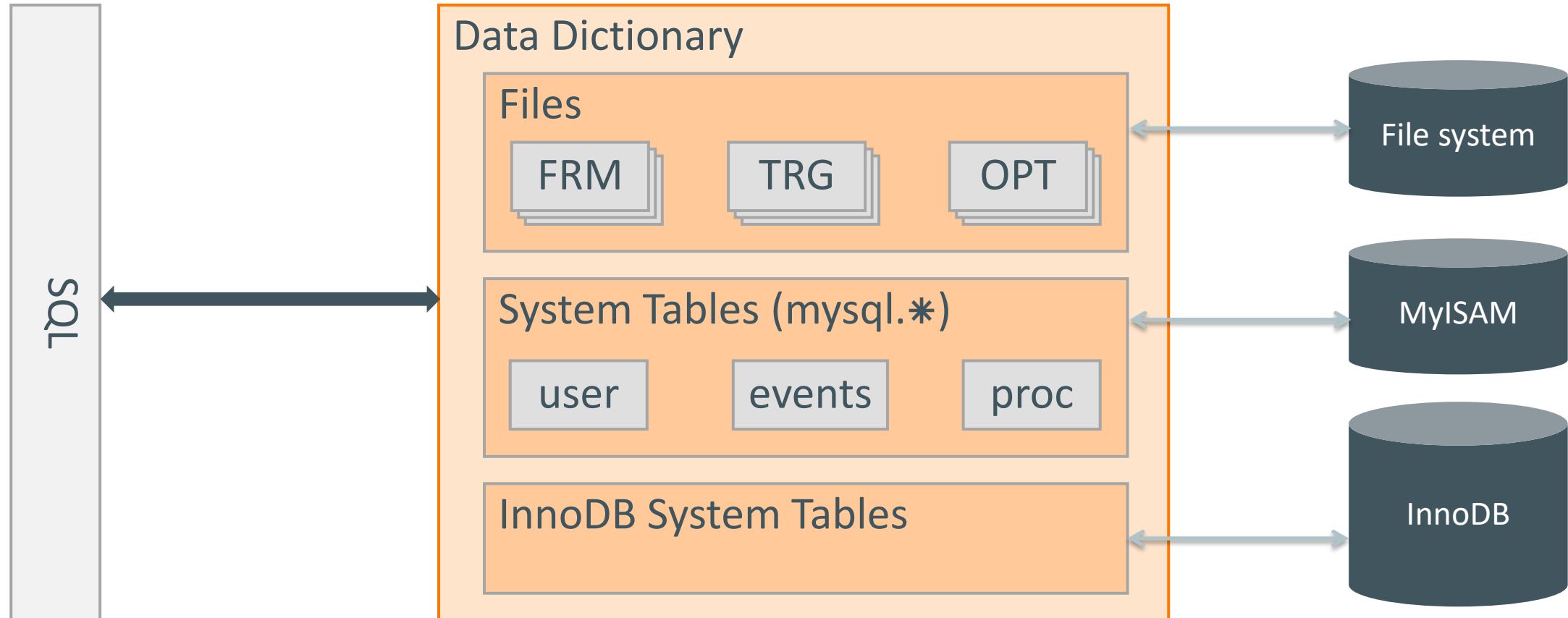
```
1 row in set (0.00 sec)
```



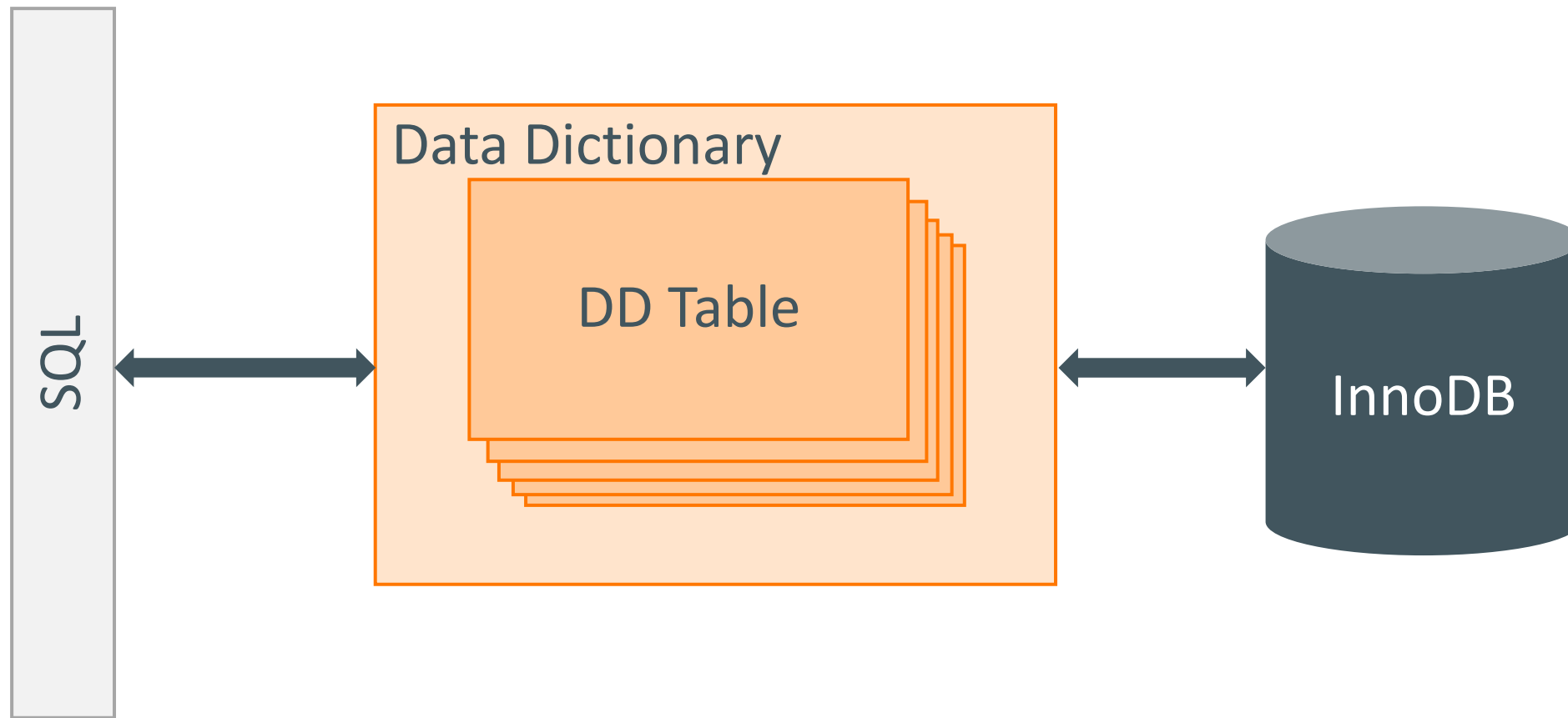
# トランザクショナルなデータディレクトリ

- 信頼性の向上
- InnoDBを使ってデータディレクトリを実現
  - No FRM files
  - No DB.OPT files
  - No TRG files
  - No TRN files
  - No PAR files
- MySQL 8.0のデフォルトインストールでは、MyISAMテーブルを含まない

# MySQL 8.0より前のMySQLデータディレクトリ



# MySQL 8.0でのデータディレクトリ



# アトミックなユーザー作成、権限付与

- MySQLの長年の課題を解決！
  - レプリケーション、バックアップ取得、などに関連
- ユーザー情報、権限情報はInnoDBテーブルで管理される
- 一度に複数処理を行ってもアトミックに処理される
  - CREATE USER u1, u2
  - GRANT SELECT ON \*.\* TO u1, u2
- ユーザー管理テーブル、権限管理テーブルに対するロックを取得することで、一連の操作をアトミックに処理できる



# InnoDB Redo/Undo 暗号化

- AES 256 暗号化
- Redo/Undoログがディスクに書き出される時に暗号化される
- Redo/Undoログがディスクから読み込まれる時に複合される
- メモリ上ではRedo/Undoログデータは暗号化されていない
- InnoDB表領域暗号化と同様の2層暗号化鍵管理
  - 鍵のローテーションが高速、高パフォーマンス
- 簡単に使用可能
  - システム変数 [innodb redo log encrypt](#)、[innodb undo log encrypt](#) で制御



# エラーを確認できるテーブル追加 (パフォーマンス・スキーマ)

Aggregation	Table Name
By Account	events_errors_summary_by_account_by_error
By Host	events_errors_summary_by_host_by_error
By Thread	events_errors_summary_by_thread_by_error
By User	events_errors_summary_by_user_by_error
Global	events_errors_summary_global_by_error

```
SELECT * FROM test.no_table;
```

```
ERROR 1146 (42S02): Table 'test.no_table' doesn't exist
```

```
SELECT * FROM performance_schema.events_errors_summary_global_by_error  
WHERE sum_error_handled > 0 OR SUM_ERROR_RAISED > 0
```

```
***** 1. row *****
```

```
ERROR_NUMBER: 1146
```

```
ERROR_NAME: ER_NO_SUCH_TABLE
```

```
SQL_STATE: 42S02
```

```
SUM_ERROR_RAISED: 1
```

```
SUM_ERROR_HANDLED: 0
```

```
FIRST_SEEN: 2016-09-11 20:52:42
```

```
LAST_SEEN: 2016-09-11 20:52:42
```

```
1 row in set (0.00 sec)
```

# 設定変更の永続化



Cloud Friendly

- 以下の構文でシステム変数の変更を永続化可能
  - `SET PERSIST`  
`sql_mode='STRICT_TRANS_TABLES,NO_ENGINE_SUBSTITUTION';`
- システム変数変更のためにファイルシステムへのアクセス不要
- `read_only=ON⇒OFF`への移行時など、再起動がしばらくできない場合などに便利
  - `my.cnf`の修正を忘れるリスクの回避
  - `my.cnf`の書き間違いにより再起動に失敗するリスクの回避
- システム変数がいつ、どこで、誰に設定されたかを確認出来る情報も追加 (`performance_schema.variables_info`)



# Common Table Expressions (WITH句)

- WITH句
- 再帰的なクエリの記述
- 複雑なSQLの簡素化

```
WITH t1 AS (SELECT * FROM tblA WHERE a='b' )  
SELECT * FROM t1;
```



# Window関数

Coming Soon!

```
SELECT name, department_id, salary,  
SUM(salary)  
  OVER (PARTITION BY department_id)  
  AS department_total  
FROM employee  
ORDER BY department_id, name;
```



# 新しいオプティマイザー・コストモデル

- データがメモリ上にある/ない、を考慮した改善

```
SELECT * FROM Country  
WHERE population > 20000000;
```



## テーブルスキャンの場合

# pages in table \*  
(IO\_BLOCK\_READ\_COST |  
MEMORY\_BLOCK\_READ\_COST)

# records \* ROW\_EVALUATE\_COST

= 25.4 100% in memory

= 29.9 100% on disk

## レンジスキャンの場合

# records\_in\_range \*  
(IO\_BLOCK\_READ\_COST |  
MEMORY\_BLOCK\_READ\_COST)

# records\_in\_range \*  
ROW\_EVALUATE\_COST + #  
records\_in\_range \*  
ROW\_EVALUATE\_COST

= 22.5 100% in memory

= 60 100% on disk

この例では、データがメモリ上にある  
場合にはレンジスキャンの方が適切  
※IOブロックリードコストのデフォルト値

- ディスク上 : 1
- メモリ上 : 0.25

データがメモリ上にある/ないに  
よって、レンジスキャンの  
パフォーマンスの差が大きい

# コストモデルの改善: 動機

- より正確なコスト見積もり
  - オプティマイザがより正確な実行計画を選択でき、パフォーマンスが向上する
- 新しいH/W環境への適応
  - SSD、大容量メモリ、キャッシュ
- よりメンテナンス可能なコストモデルの実装
  - ハードコートされた「コスト定数」を避ける
  - 既存のコストモデルのリファクタリング
- チューニング可能なコストモデル



より高速な  
クエリー

# ヒストグラム

Coming Soon!

- データが偏っている場合のクエリーの精度向上
- インデックスより保守コストが低い



# 他にも多くの点を改良

- Source code now documented with Doxygen
- Plugin Infrastructure!
- Expanded GIS Support
- Expanded Query Hints Support
- Improved Scan Query Performance
- Improved BLOB Storage
- Improved Memcached Interface
- Cost Model Improvements
- Scalability Improvements
- Atomicity in Privileges
- Parser Refactoring
- Improvements to Temporary Tables
- C++11 and Toolchain Improvements
- Replication Applier Thread Progress Reports
- GTID\_PURGED always settable
- **New!** Improved Parallel Replication
- **New!** SQL Grouping Function
- **New!** Optimizer Trace detailed sort statistics
- **New!** Smaller Package Downloads
- **New!** JSON Aggregate, Pretty print functions
- **New!** JSON performance improvements
- **New!** Expanded Query Hints
- **New!** Improved usability of cost constant configuration

# アジェンダ

- 1 ▶ Oracle MySQL Cloud Service
- 2 ▶ MySQL 8.0 DMR 新機能
- 3 ▶ MySQL Group Replication、MySQL InnoDB Cluster
- 4 ▶ MySQL Enterprise Edition

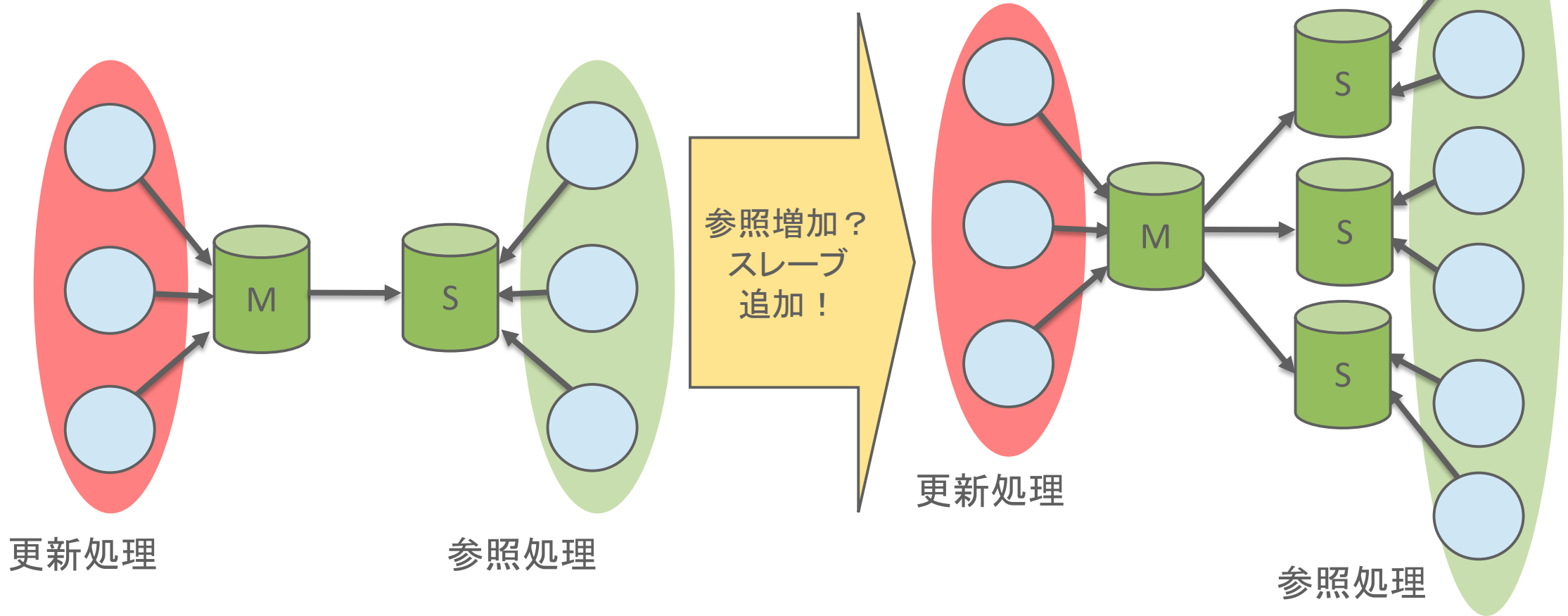
# グループレプリケーションとは？

- MySQL5.7以降で利用可能な仮想同期レプリケーション
- MySQLがサポートする全てのプラットフォームに対応
  - Linux, Windows, Solaris, OSX, FreeBSD
- 手軽に高可用性構成を実現可能
  - 複数台でグループを組み、全台が同じデータを持つ
  - 3台以上の奇数でグループを構成することを推奨
- シングルプライマリーモード(デフォルト)とマルチマスターモードが使用可能



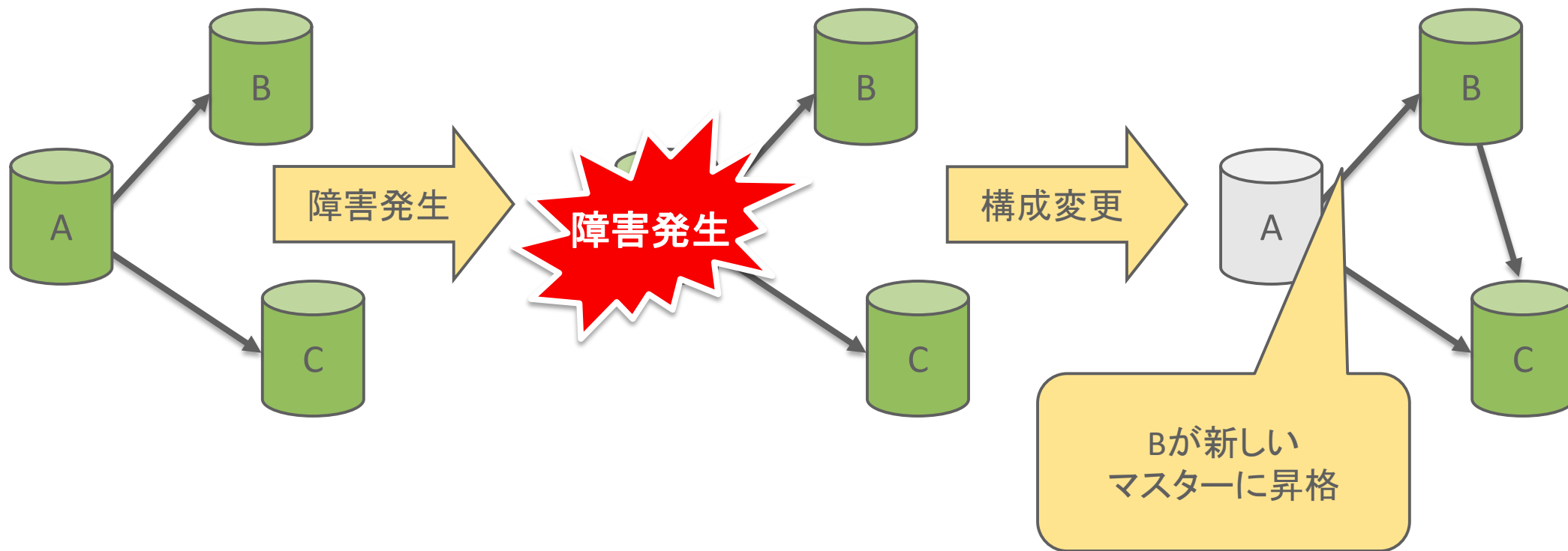
# 補足: レプリケーションの用途

## 参照性能のスケールアウト



# 補足: レプリケーションの用途

冗長性: マスターがクラッシュした場合, スレーブをマスターに昇格



# シングルプライマリーモード

- シングルメンバーが書き込み可能なマスター(PRIMARY)として動作して、他のグループメンバーはホットスタンバイ(SECONDARY)として動作します。
  - リーダー選択メカニズムにて、どのメンバーがPRIMARYとして動作すべきか、判断して自動的にグループ内でコーディネートします。
- この新しいコンフィギュレーションモード導入の背後にある主な動機は、グループレプリケーションユーザーに対して、ユーザーが使い慣れている非同期レプリケーションに近い、シングルマスターレプリケーションの環境を選択出来るようにする為

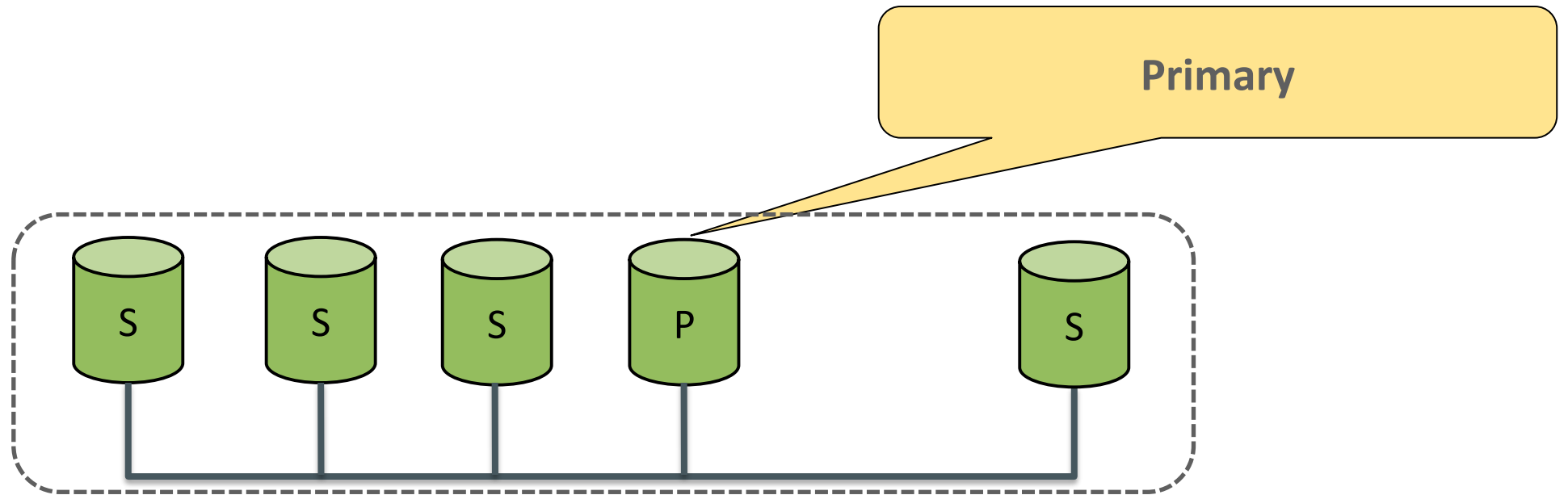
variable_name	value
group_replication_single_primary_mode	ON

variable_name	value
group_replication_enforce_update_everywhere_checks	OFF

[1]: <http://mysqlhighavailability.com/mysql-group-replication-for-mysql-5-7-15/>

# シングルプライマリーモード

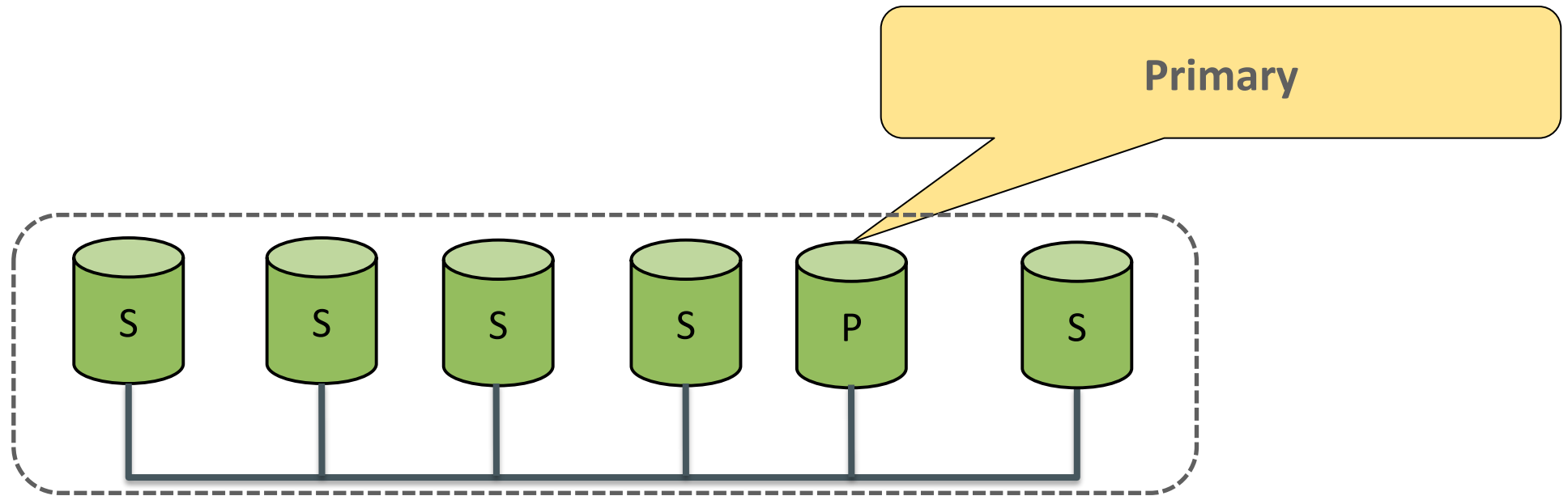
- 自動的なリーダー選択メカニズム



[1]: <http://mysqlhighavailability.com/mysql-group-replication-for-mysql-5-7-15/>

# シングルプライマリーモード

- 自動的なリーダー選択メカニズム
  - Secondaryノードは参照のみ可能 (書き込んだ場合: ERROR 1290)



ERROR 1290 (HY000): The MySQL server is running with the --super-read-only option so it cannot execute this statement

# シングルプライマリーモード

- 現在のプライマリサーバーのメンバーIDは、次のSQL文を実行することによって確認する事が出来ます。

```
mysql> SELECT * FROM performance_schema.replication_group_members;
```

CHANNEL_NAME	MEMBER_ID	MEMBER_HOST	MEMBER_PORT	MEMBER_STATE
group_replication_applier	919a7d77-9ff3-11e6-acb8-080027d65c57	replications	3310	ONLINE
group_replication_applier	9c8e6f77-9ff3-11e6-b1d1-080027d65c57	replications	3320	ONLINE
group_replication_applier	ac0ec52b-9ff3-11e6-b801-080027d65c57	replications	3330	ONLINE

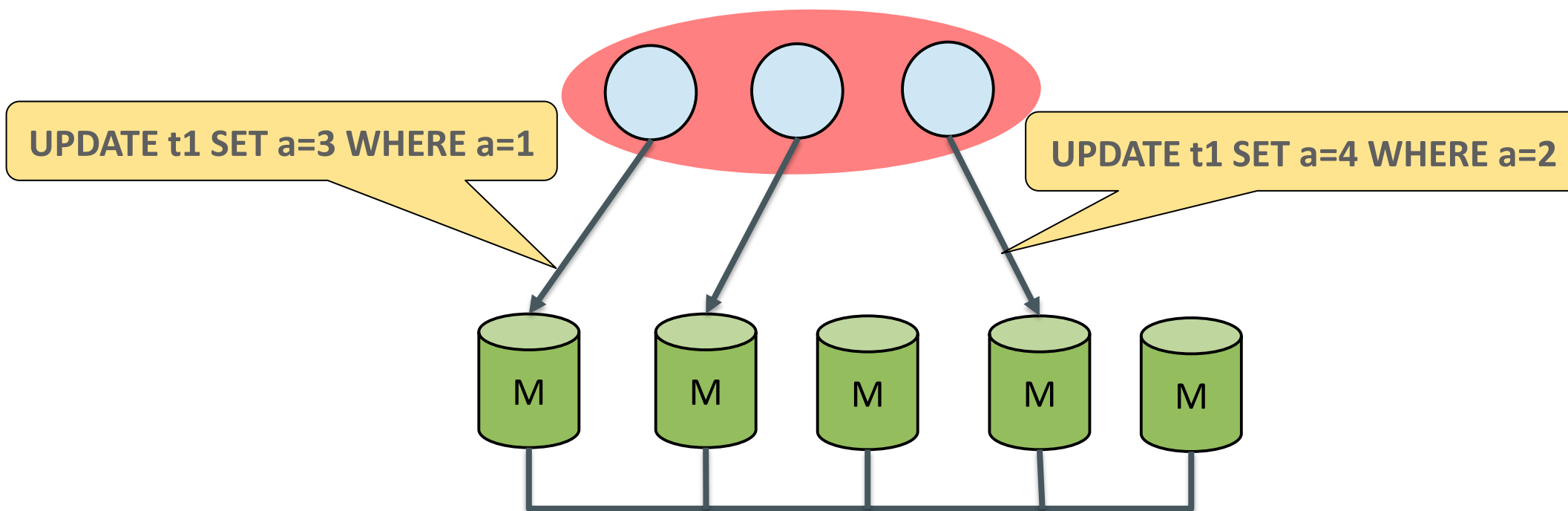
```
mysql> SELECT * FROM performance_schema.global_status WHERE  
VARIABLE_NAME='group_replication_primary_member';
```

VARIABLE_NAME	VARIABLE_VALUE
group_replication_primary_member	919a7d77-9ff3-11e6-acb8-080027d65c57

[1]: <http://mysqlhighavailability.com/mysql-group-replication-for-mysql-5-7-15/>

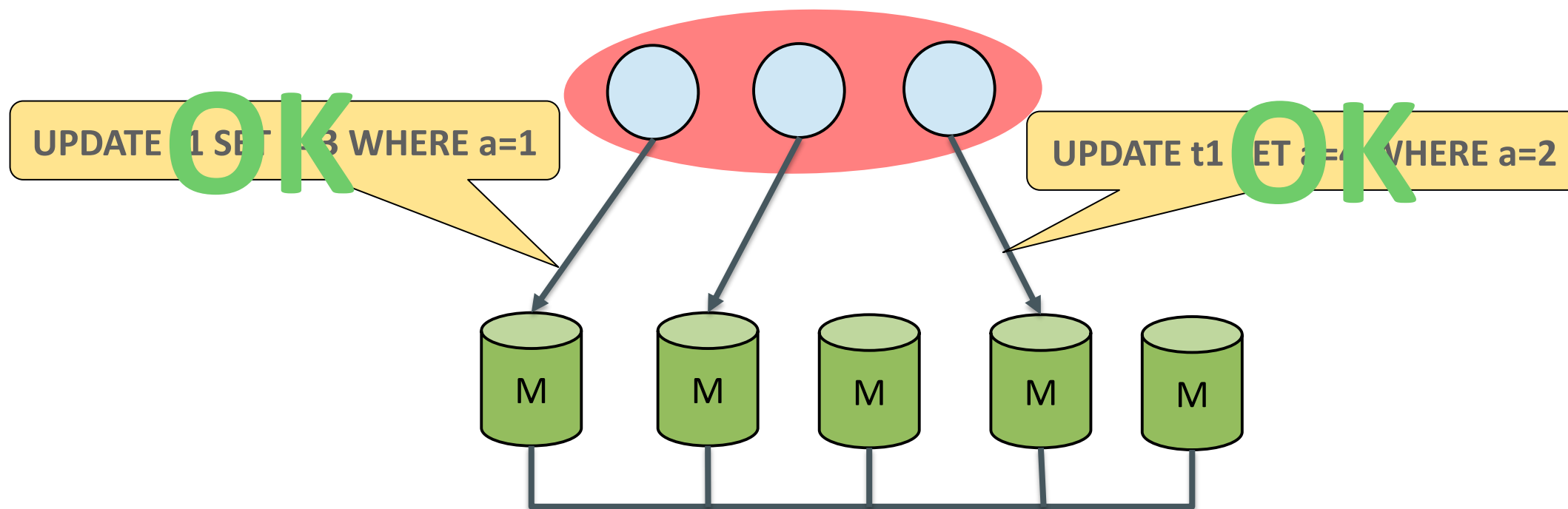
# マルチマスターモード:どこでも更新可能

- 異なるサーバー上での2つのトランザクションは、同じデータを更新可能
- 競合が検出された場合、自動的に対処される
  - 先にコミットしたトランザクションが優先される



# マルチマスターモード:どこでも更新可能

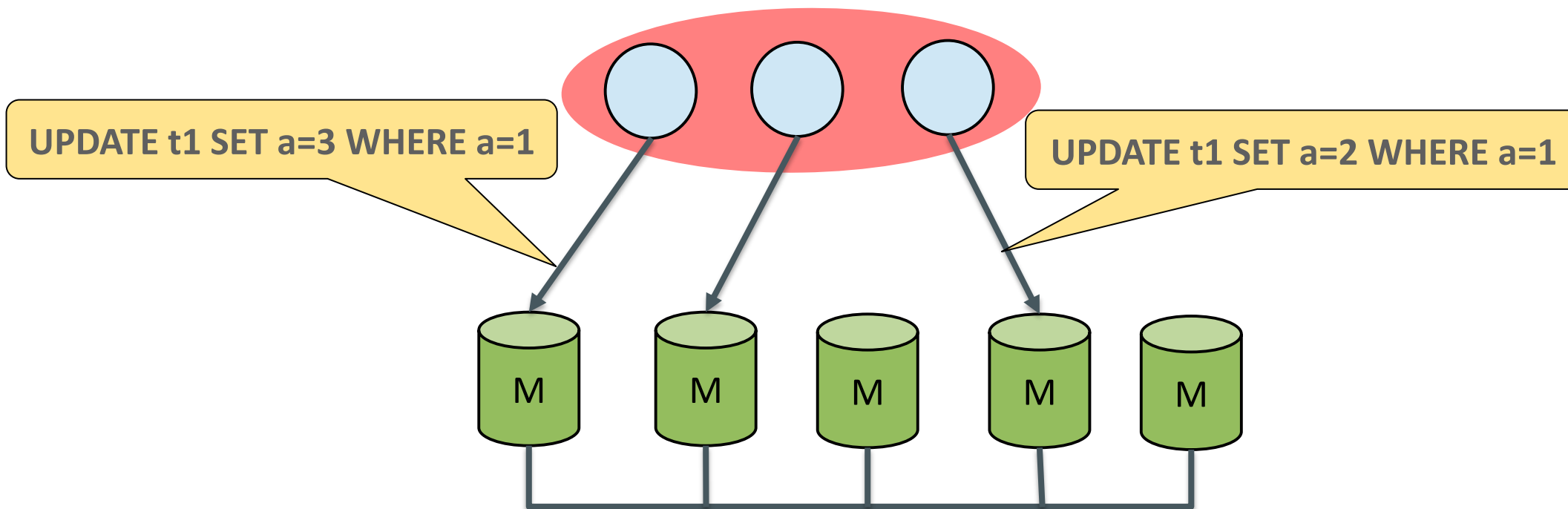
- 異なるサーバー上での2つのトランザクションは、同じデータを更新可能
- 競合が検出された場合、自動的に対処される
  - 先にコミットしたトランザクションが優先される





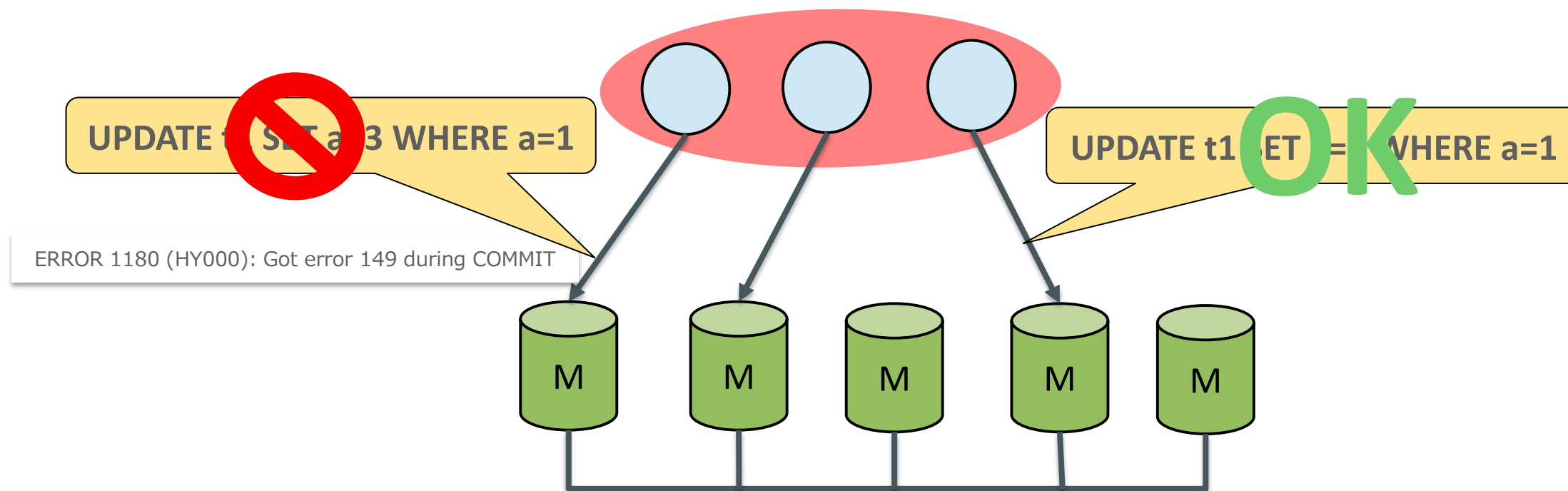
# マルチマスターモード:どこでも更新可能

- 異なるサーバー上での2つのトランザクションは、同じデータを更新可能
- 競合が検出された場合、自動的に対処される
  - 先にコミットしたトランザクションが優先される



# マルチマスターモード: どこでも更新可能

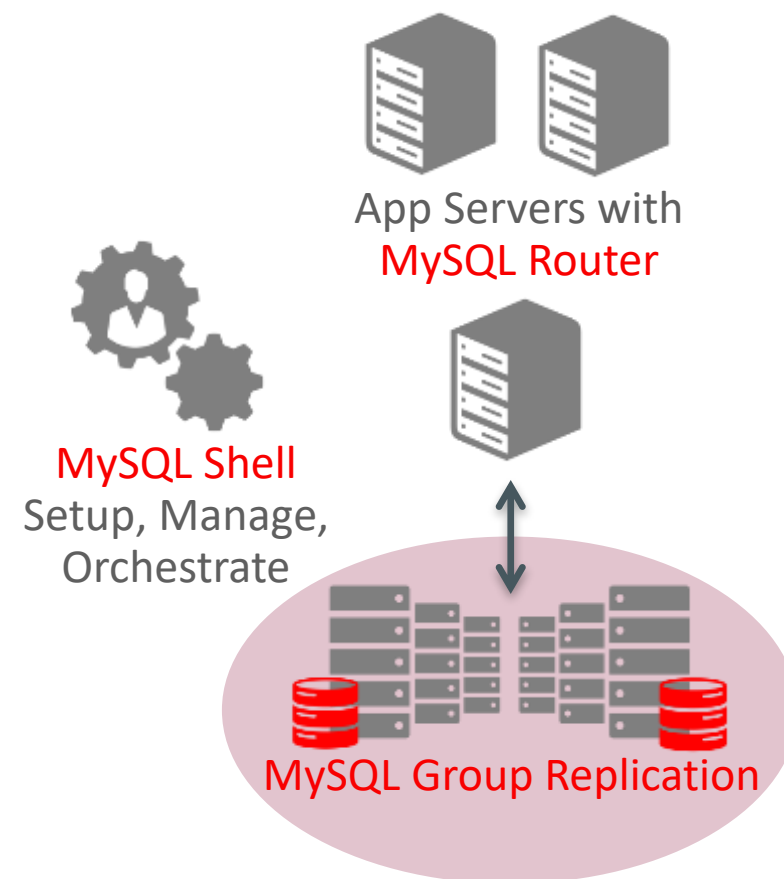
- 異なるサーバー上での2つのトランザクションは、同じデータを更新可能
- 競合が検出された場合、自動的に対処される
  - 先にコミットしたトランザクションが優先される



# MySQL InnoDB Cluster

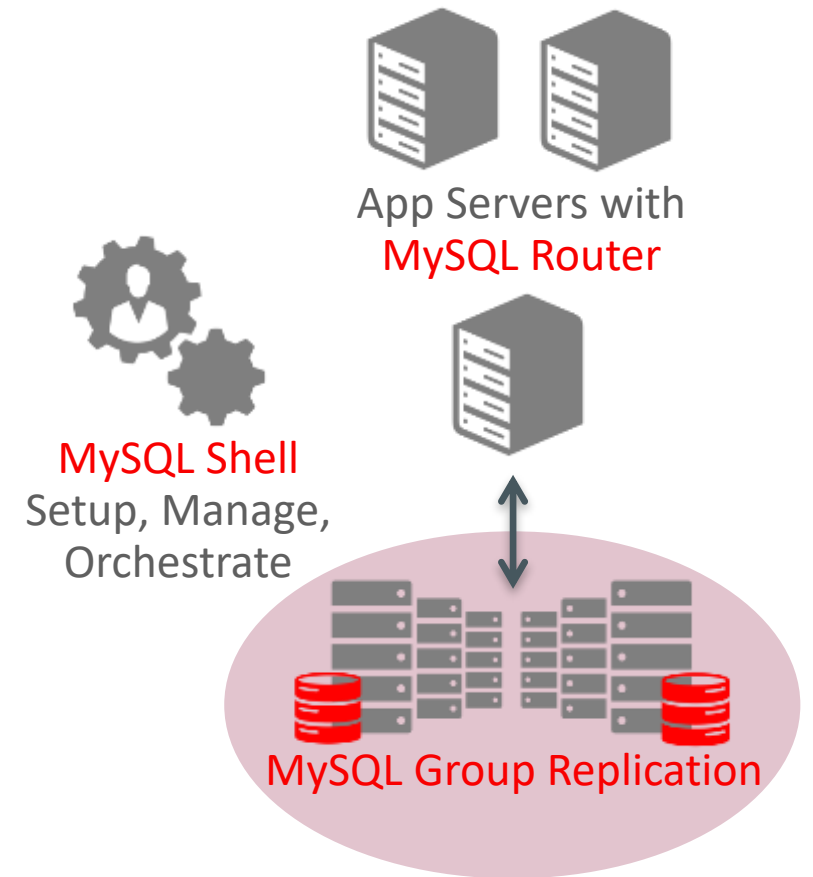
# MySQL InnoDB Clusterとは？

- 以下のコンポーネントの組み合わせから構成されるMySQLの高可用性フレームワーク
  - MySQL Group Replication
    - DBの読み取り拡張性、自動フェイルオーバーを提供
  - MySQL Router
    - アプリ接続先の自動フェイルオーバーを提供
  - MySQL Shell
    - グループ・レプリケーション環境の構築、設定、Routerの設定



# MySQL InnoDB Clusterとは？

- 2017年4月12日 GA
  - 以下の製品を個別にインストールすることで使用可能
    - MySQL 5.7.18 (2017-04-10)
    - MySQL Router 2.1.3 (2017-04-12)
    - MySQL Shell 1.0.9 (2017-04-12)



# MySQL InnoDB Clusterのチュートリアル

- 以下のセミナー資料に、コマンド付きのMySQL InnoDB Clusterのチュートリアルが含まれています
  - MySQLの新しい高可用性構成  
MySQLグループ・レプリケーションとMySQL InnoDB Cluster  
<https://www.mysql.com/jp/why-mysql/presentations/mysql-innodb-cluster-201704-ja/>
- チュートリアル動画もあります
  - <https://youtu.be/RfyxIGS4Zks>

## Ease-of-Use

Built-in HA

MySQL  
**InnoDB**  
cluster

## Out-of-Box Solution

Everything Integrated

## Scale-Out

High Performance



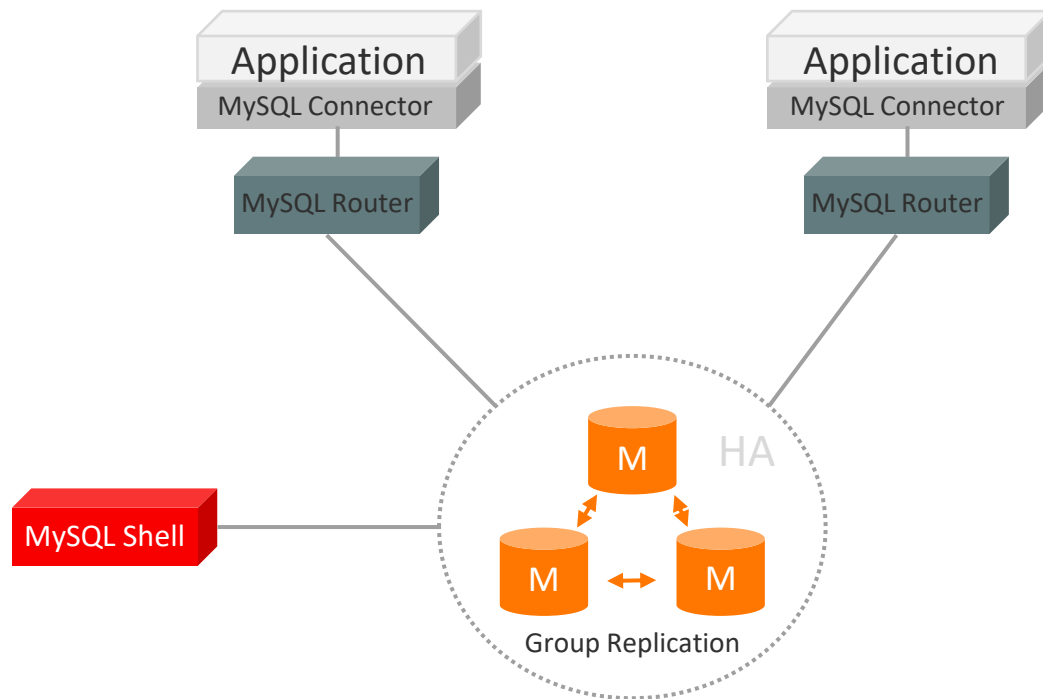
- Ease-of-Use
- 15分でインストール, HA ,スケールアウト設定が可能
  - MySQLユーザーの為のシングルインターフェイス
  - 簡単にセットアップ, スケールアウト, 管理 & モニタリング
  - 優れた品質

- Out-of-Box Solution
- 統合ソリューション vs. 個別のコンポーネント
  - 設計& 開発済み環境を同時に提供
  - 検証済み環境を同時に提供
  - 管理及び監視環境を同時に提供

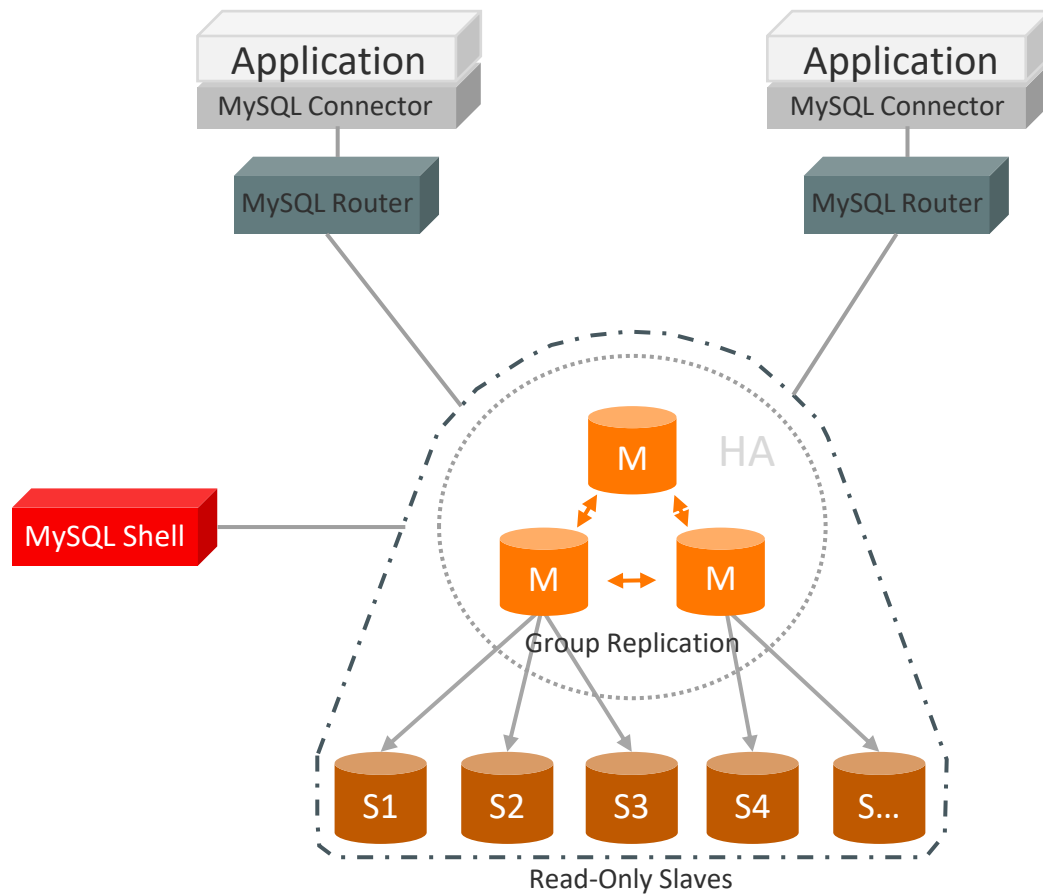
- Scale-Out
- ワールドクラスの性能を維持
  - 自動フェイルオーバー含め信頼性のあるHAをサーバー側で提供
  - 参照処理の拡張:レプリケーション
  - 書き込み処理の拡張:シャーディング



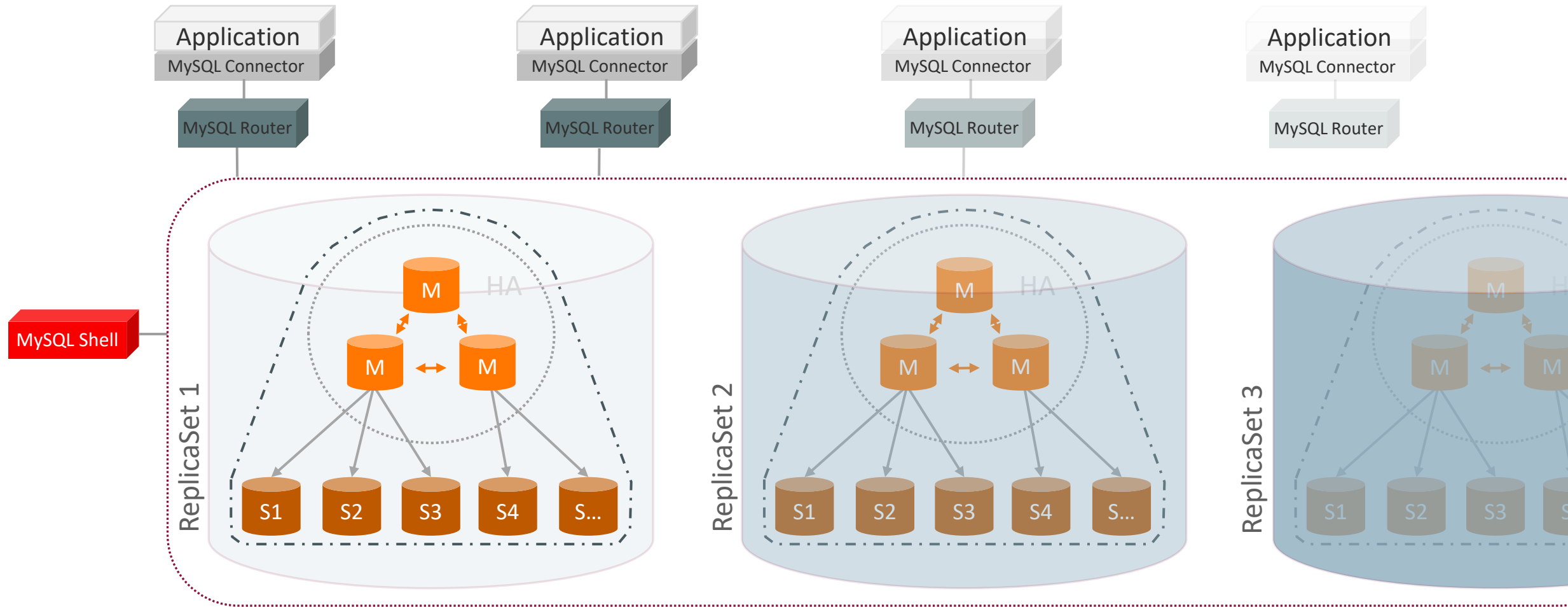
# MySQL InnoDB Cluster: **Architecture – Step 1**



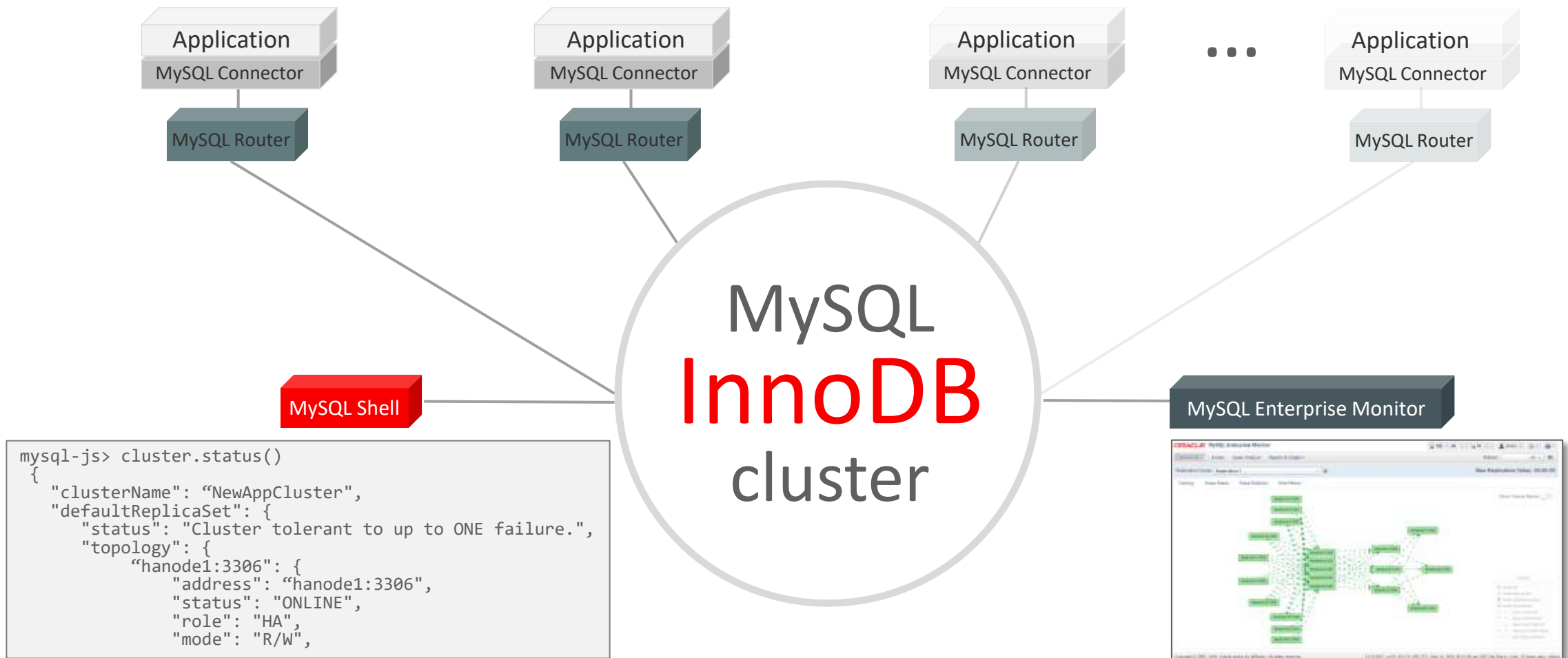
# MySQL InnoDB Cluster: Architecture – Step 2



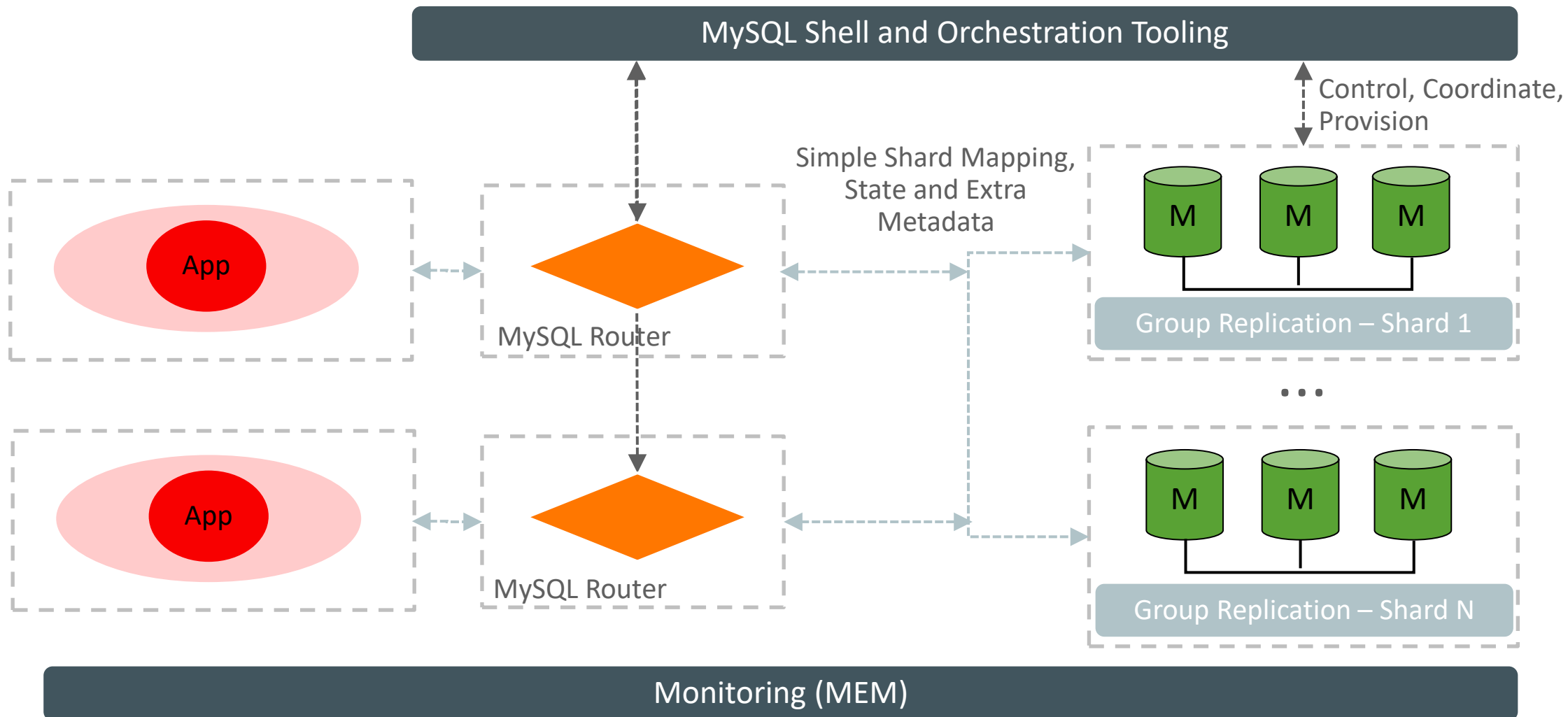
# MySQL InnoDB Cluster: **Architecture – Step 3**



# MySQL InnoDB Cluster: High Level Architecture



# 参考) MySQL InnoDB Cluster: The End Goal



# MySQL InnoDB Cluster: ゴール

## 単一製品: MySQL

- 全てのコンポーネントを同時に開発
- 全てのコンポーネントを同時に検証
- 一つのパッケージとして提供

## 容易な利用

- シングルクライアント: MySQL Shell
- 容易なパッケージング
- 同種のサーバー群

## 近代的な柔軟性

- C++ 11 (ISO標準 ISO/IEC 14882:2011)
- Protocol Buffers
- 開発フレンドリー

## スケールアウト

- シャード・クラスター
- Nレプリカセットのフェデレーテッド構成
- 各レプリカセットによるシャードの管理

デモ: <https://www.youtube.com/watch?v=JWy7ZLXxtZ4>

# MySQL Shell

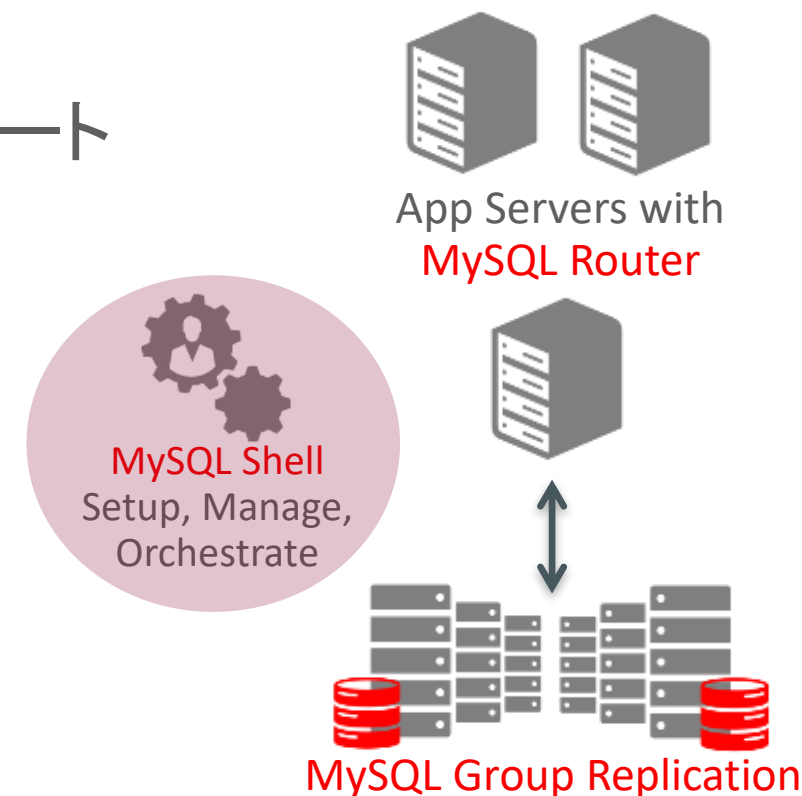
MySQL Shell 1.0.9 GA (2017-04-12)

すべての運用管理タスクのための、統一された単一クライアント

- 多言語対応: JavaScript, Python, and SQL
- ドキュメントとリレーショナルモデルの両方をサポート
- 開発と管理用に完全なAPIを提供

*"MySQL Shell provides the developer and DBA with a single intuitive, flexible, and powerfull interface for all MySQL related tasks!"*

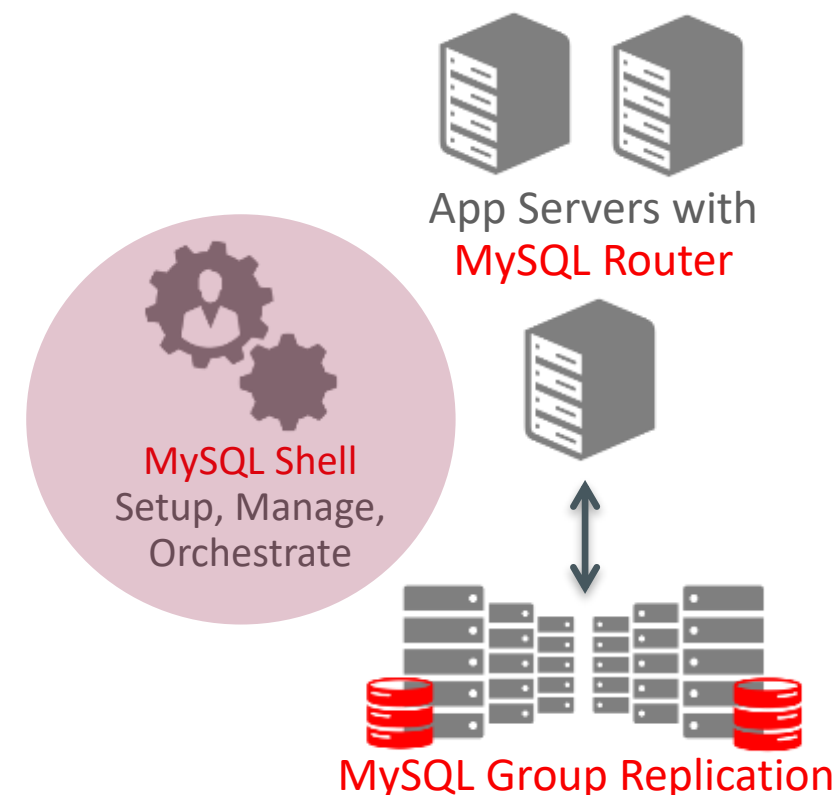
```
[root@misc01 admin]# mysqlsh --help | egrep -i "Start in"
--sql           Start in SQL mode using a node session.
--sqlc          Start in SQL mode using a classic session.
--js            Start in JavaScript mode.
--py            Start in Python mode.
[root@misc01 admin]#
```



# MySQL Shell: 管理用API

## データベース管理者向けインターフェース

- `mysql-js> dba.help()`
- グローバル変数 'dba' がMySQLの管理用APIにアクセスする為に使用可能
- DBA管理オペレーション
  - Manage MySQL InnoDB clusters
    - クラスター作成
    - MySQLインスタンスの構築
    - クラスターの状況を確認可能
    - MySQLインスタンスの開始・停止
    - MySQLインスタンスの検証 ...





mysql-js> **dba.help()**

The global variable 'dba' is used to access the MySQL AdminAPI functionality and perform DBA operations. It is used for managing MySQL InnoDB clusters.

The following properties are currently supported.

- verbose Enables verbose mode on the Db operations.

The following functions are currently supported.

- |                       |  |
|-----------------------|--|
| - createCluster       | Creates a MySQL InnoDB cluster.                          |
| - deleteLocalInstance | Deletes an existing MySQL Server instance on localhost.  |
| - deployLocalInstance | Creates a new MySQL Server instance on localhost.        |
| - dropMetadataSchema  | Drops the Metadata Schema.                               |
| - getCluster          | Retrieves a cluster from the Metadata Store.             |
| - help                | Provides help about this class and its members           |
| - killLocalInstance   | Kills a running MySQL Server instance on localhost.      |
| - resetSession        | Sets the session object to be used on the Db operations. |
| - startLocalInstance  | Starts an existing MySQL Server instance on localhost.   |
| - stopLocalInstance   | Stops a running MySQL Server instance on localhost.      |
| - validateInstance    | Validates an instance for usage in Group Replication.    |

For more help on a specific function use dba.help('<functionName>')

e.g. dba.help('deployLocalInstance')

# MySQL Shellの機能拡張

MySQL Shell  
Setup, Manage,  
Orchestrate



GA

ノードの追加, Group Replication設定, Router連携機能を実装

## MySQL Shell – Deploy MySQL Instances

```
shell> mysqlsh
mysql-js> dba.deployLocalInstance(3306)
mysql-js> dba.deployRemoteInstance('192.168.1.2:3306')
mysql-js> dba.deployRemoteInstance('192.168.1.3:3306')
```

## MySQL Shell – Create InnoDB Cluster

```
shell> mysqlsh --uri root@localhost:3306
mysql-js> cluster = dba.createCluster('NewAppCluster')
mysql-js> cluster.addInstance('root@192.168.1.2:3306')
mysql-js> cluster.addInstance('root@192.168.1.3:3306')
```

## MySQL Shell – Add MySQL Router

```
shell> mysqlrouter --bootstrap localhost:3306
shell> mysqlrouter &
shell> mysqlsh --uri root@localhost:6442
```

## MySQL Shell – Check Status

```
shell> mysqlsh --uri root@localhost:3306
mysql-js> cluster = dba.getCluster()
mysql-js> cluster.status()
```

<https://www.youtube.com/watch?v=JWy7ZLXxtZ4>

# アジェンダ

- 1 ▶ Oracle MySQL Cloud Service
- 2 ▶ MySQL 8.0 DMR 新機能
- 3 ▶ MySQL Group Replication、MySQL InnoDB Cluster
- 4 ▶ MySQL Enterprise Edition

# MySQL Enterprise Editionとは？

# MySQL Enterprise Edition

ビジネス・クリティカルな環境において、最高レベルのMySQLスケーラビリティ、セキュリティ、信頼性、アップタイムを実現し、ビジネス・クリティカルな環境においてリスクとコストを削減を実現



## MySQL導入の最適化



## ROIの最適化をサポート



## ユーザビリティ・顧客満足度の向上



# MySQL Enterprise Edition のサービスカテゴリー



## 拡張機能

- 拡張性
- 高可用性
- 統合認証
- 監査
- 暗号化
- ファイヤーウォール
- 透過的データ暗号化



## 管理ツール

- 監視
- バックアップ
- 開発
- 管理
- マイグレーション



## サポート

- 技術サポート
- コンサルティングサポート
- オラクル製品との動作保証



	MySQL Editions		
	Standard Edition	Enterprise Edition	Cluster CGE
機能概要			
MySQL Database	✓	✓	✓
MySQL Connectors	✓	✓	✓
MySQL Replication	✓	✓	✓
MySQL Fabric, MySQL Utilities, MySQL Router		✓	✓
MySQL Partitioning		✓	✓
Storage Engine: MyISAM, InnoDB	✓	✓	✓
Storage Engine: NDB (ndbcluster)			✓
MySQL Workbench SE/EE*	✓	✓	✓
MySQL Enterprise Monitor*		✓	✓
MySQL Enterprise Backup*		✓	✓
MySQL Enterprise Authentication (外部認証サポート)*		✓	✓
MySQL Enterprise TDE (Transparent Data Encryption)*		✓	✓
MySQL Enterprise Encryption (非対称暗号化)*		✓	✓
MySQL Enterprise Firewall (SQLインジェクション対策)*		✓	✓
MySQL Enterprise Audit (ポリシーベース監査機能)*		✓	✓
MySQL Enterprise Scalability (スレッドプール)*		✓	✓
MySQL Enterprise High Availability (HAサポート)*		✓	✓
Oracle Enterprise Manager for MySQL *		✓	✓
MySQL Cluster Manager (MySQL Cluster管理)*			✓
MySQL Cluster Geo-Replication			✓

	MySQL Editions		
	Standard Edition	Enterprise Edition	Cluster CGE
Oracle Premium Support			
24時間365日サポート	✓	✓	✓
インシデント数無制限	✓	✓	✓
ナレッジベース	✓	✓	✓
バグ修正&パッチ提供	✓	✓	✓
コンサルティングサポート	✓	✓	✓
オラクル製品との動作保証			
Oracle Linux	✓	✓	✓
Oracle VM	✓	✓	✓
Oracle Solaris	✓	✓	✓
Oracle Enterprise Manager		✓	✓
Oracle GoldenGate		✓	✓
Oracle Data Integrator		✓	✓
Oracle Fusion Middleware		✓	✓
Oracle Secure Backup		✓	✓
Oracle Audit Vault and Database Firewall		✓	✓

※最新の対比表は、[MySQL Editions](#)のサイトを参照下さい



# MySQL Enterprise Edition 管理ツールと拡張機能概要

## MySQL Enterprise Edition

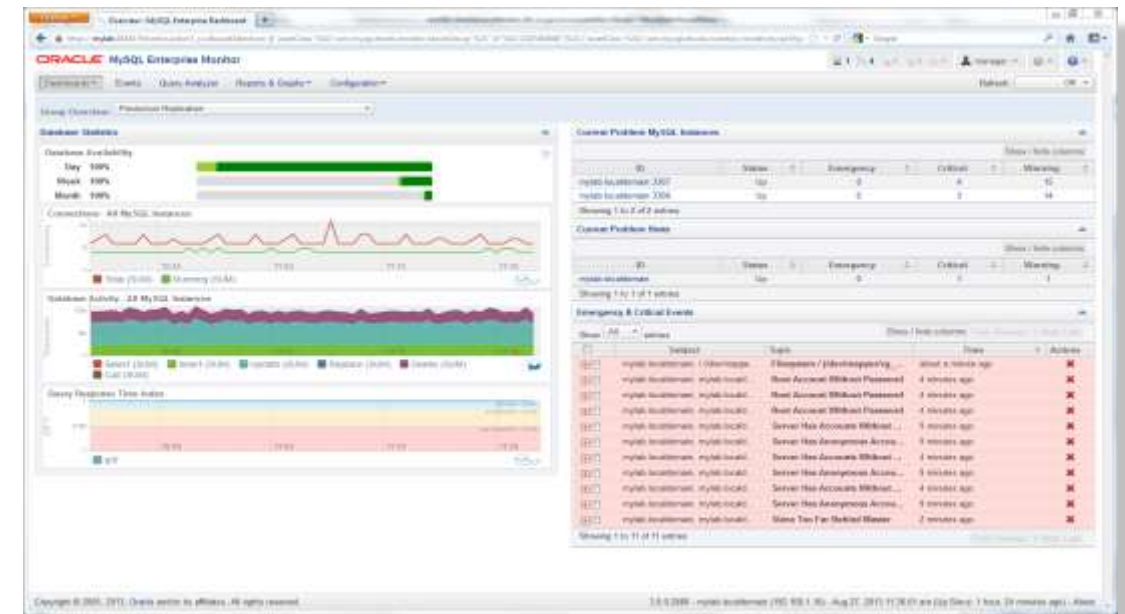
MySQL Enterprise Monitor	複数サーバの一括管理、クエリ性能分析
MySQL Enterprise Backup	高速なオンラインバックアップ、ポイントインタイムリカバリ
MySQL Enterprise Authentication	LDAPやWindows Active Directoryとの外部認証と統合管理
MySQL Enterprise TDE	データベース全体の暗号化(透過的)
MySQL Enterprise Encryption	非対称暗号化( <a href="#">公開鍵暗号</a> )の業界標準機能を提供
MySQL Enterprise Firewall	SQLインジェクション対策
MySQL Enterprise Audit	ユーザ処理の監査、Oracle DBと同じツールで管理可能
MySQL Enterprise Scalability	Thread Poolプラグインによる性能拡張性の向上
Oracle Enterprise Manager for MySQL	Oracle Enterprise ManagerからMySQLを統合管理可能
Oracle Premier Support	24x7, インシデント無制限、コンサルティングサポート

# MySQL Enterprise Monitor

- 複数のMySQLサーバを一括監視可能なダッシュボード
- システム中のMySQLサーバやレプリケーション構成を自動的に検出し監視対象に追加
- ルールに基づく監視と警告
- **問題が発生する前に通知**
- 問題のあるSQL文の検出、統計情報の分析が可能なQuery Analyzer

参照: [MySQL Enterprise Monitor](#)

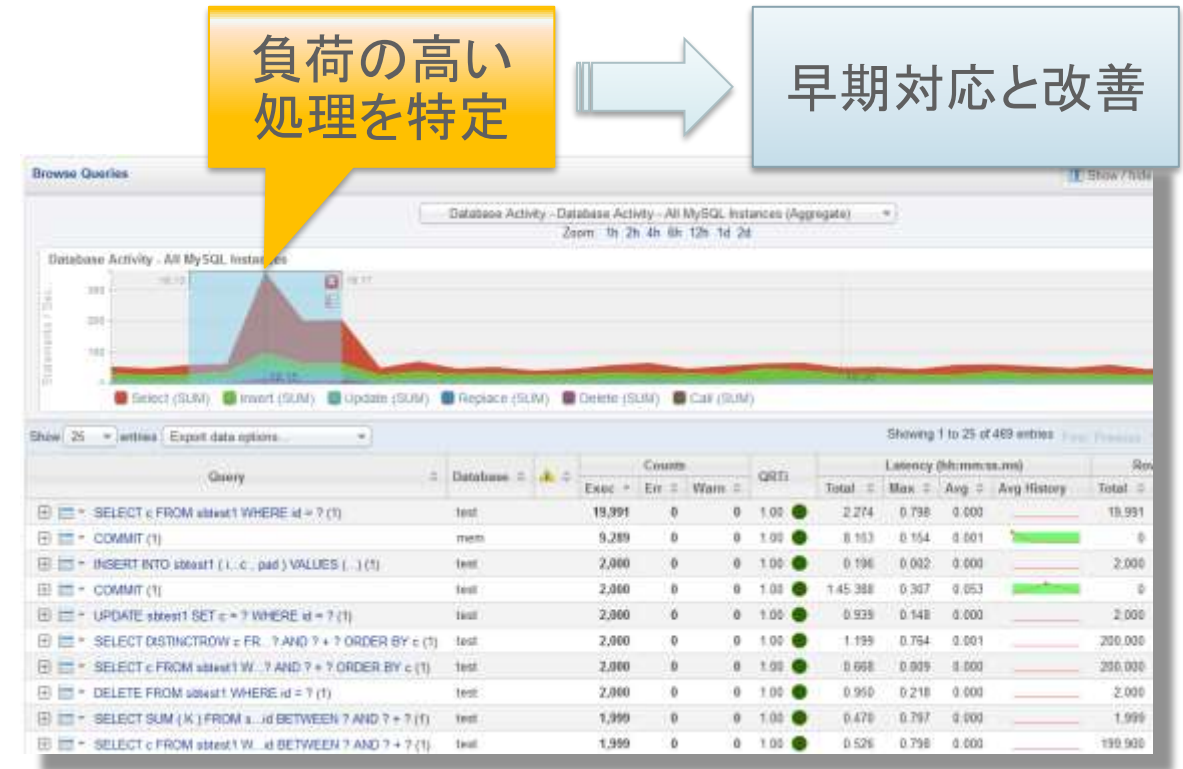
Diagnostic Reportを利用してサポートとの情報共有を容易にする事も可能



"バーチャルなMySQL DBA"

# クエリ解析機能 - MySQL Query Analyzer

- 全てのMySQLサーバの  
全てのSQL文を一括監視
- vmstatなどのOSコマンドやMySQLの  
SHOWコマンドの実行、  
ログファイルの個別の監視は不要
- クエリの実行回数、エラー回数、  
実行時間、転送データ量などを  
一覧表示
- チューニングのための  
解析作業を省力化



# MySQL Enterprise Backup

## 高速、オンラインバックアップ & リカバリ

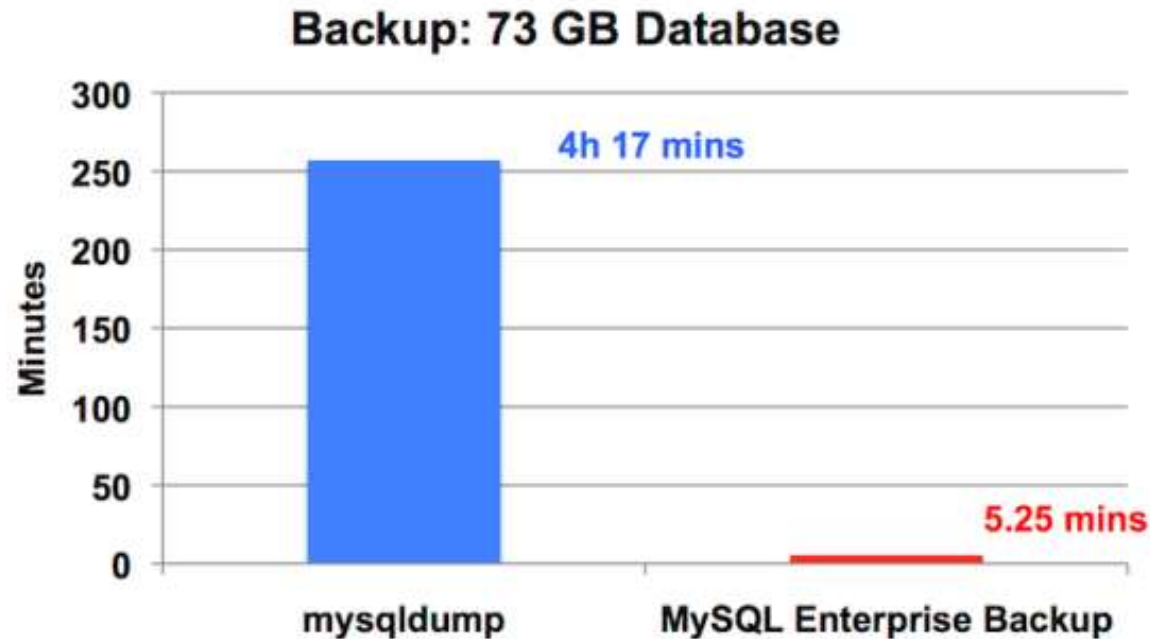
- InnoDBのオンラインバックアップツール
- フル、増分、部分バックアップ(圧縮可能)
- マルチスレッドによる並列バックアップ & リカバリ処理
- クラウドストレージとの直接の連携 (S3, Swift API)
- バックアップの暗号化 – AES 256
- Oracle Secure Backupとの連携

## MySQL Enterprise Backup の特徴と利点

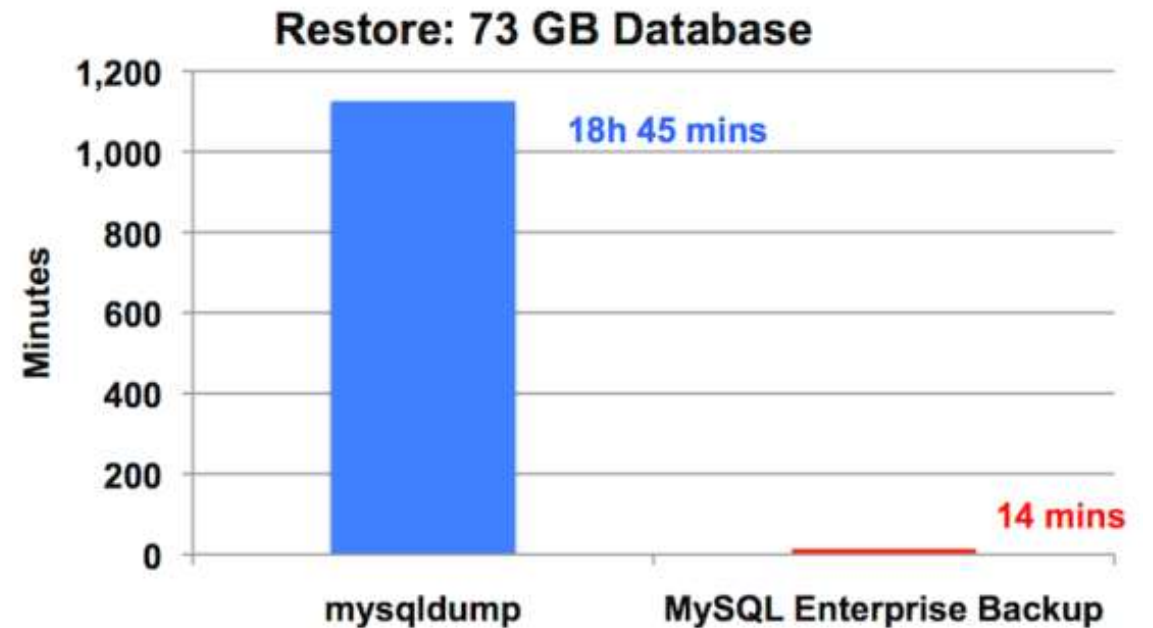


# 高速なバックアップとリカバリー

高速なオンラインバックアップ&リカバリー処理により、機会損失を最小限に抑える事が可能

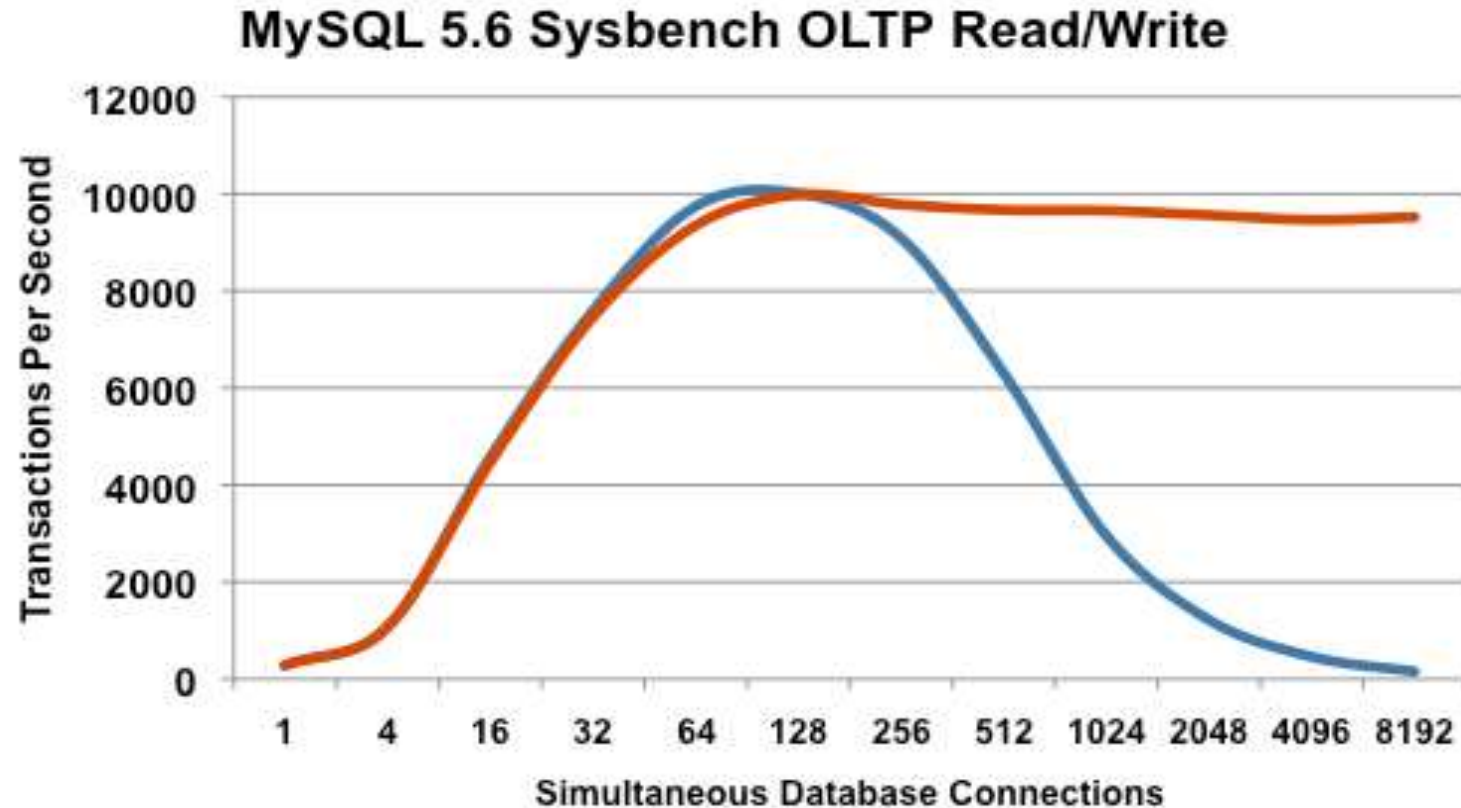


mysqldumpより49倍速い



mysqldumpより80倍速い

# MySQL Enterprise Scalability : Thread Pool



**MySQL Enterprise Edition**

Thread Pool有り

**MySQL Community Edition**

Thread Pool無し

MySQL 5.6.11  
Oracle Linux 6.3、Unbreakable Kernel 2.6.32  
4 sockets、24 cores、48 Threads  
Intel(R) Xeon(R) E7540 2GHz CPUs  
512GB DDR RAM

**Thread Poolでスケーラビリティが60倍向上**

参照: [MySQL Enterprise Scalability](#)

# MySQL Enterprise Edition 5.7: 統合されたセキュリティ機能

- MySQL Enterprise **Firewall**
  - SQLインジェクション攻撃をブロック
  - 侵入者を検知
- MySQL Enterprise **TDE**
  - 保存データの暗号化(透過的)
  - 鍵管理
- MySQL Enterprise **Encryption**
  - デジタル署名、データバリデーション
  - MySQL KeyRing (Oracle Key Vaultと連携)
- MySQL Enterprise **Authentication**
  - 外部認証モジュール
    - Microsoft AD, Linux PAMs
- MySQL Enterprise **Audit**
  - ユーザーアクティビティの監査、法令順守
  - テーブルレベルの監査
- MySQL Enterprise **Monitor**
  - データベース設定、ユーザー権限、スキーマ、パスワードの変更等のモニタリング
  - MySQL Enterprise Firewall監視
- MySQL Enterprise **Backup**
  - セキュアなバックアップ、AES 256暗号化



More information available at : <http://www.mysql.com/products/enterprise/>

# MySQL Enterprise Edition 5.7: 統合されたセキュリティ機能

- MySQL Enterprise Edition を使うことで、セキュリティ対策を強化できます
- セキュリティ要件が厳しいシステムでも、是非MySQL Enterprise Editionの採用をご検討ください！！



# 技術サポート

# MySQL Enterprise Support

- 最大のMySQLのエンジニアリングおよびサポート組織
- MySQL開発チームによるサポート
- 29言語で世界クラスのサポートを提供
- メンテナンス・リリース、バグ修正、パッチ、アップデートの提供
- 24時間x365日サポート
- 無制限サポート・インシデント
- MySQL コンサルティング・サポート

～リモートDBAとして、是非ご活用ください！！～



Get immediate help for any MySQL issue, plus expert advice

# MySQL Supportの特徴

- 「パフォーマンス・チューニング」や「SQLチューニング」まで通常サポートの範囲内
  - コンサルティングサポートが含まれており、「クエリ・レビュー」、「パフォーマンス・チューニング」、「レプリケーション・レビュー」、「パーティショニング・レビュー」などに対応可能
  - 詳細はこちらを参照下さい  
<http://www-jp.mysql.com/support/consultative.html>
- ソースコードレベルでサポート可能
  - ほとんどのサポートエンジニアがソースを読めるため、対応が早い開発エンジニアとサポートエンジニアも密に連携している

サポート、  
コンサルテーティブ・サポート  
は共に回数制限が無い為、  
リモートDBAとして活用頂く事  
で、自社内の調査・検証工数  
を大幅に削減する事が可能。  
TCO削減が可能です。

# MySQL Supportの特徴

- **物理サーバー単位課金**

- CPU数、コア数に依存しない価格体系
- 4CPUまで(コア数は制限無し)同一料金、5CPU以上の価格は営業問合せ

- **コミュニティ版バイナリに対してもサポートを提供可能**

- サブスクリプションを契約することで、バイナリを入れ替えずにサポートを受けられる(バイナリはオラクルが提供しているものをご使用ください)
- 商用版の機能を使用する場合のバイナリ入れ替えの必要性については、P22参照
- Oracle CloudのMySQL Cloud Service以外のDBaaSはサポート対象外

- **オラクルのライフタイムサポート**

- 詳細はこちらを参照下さい

<http://www.oracle.com/jp/support/lifetime-support/index.html>

<http://www-jp.mysql.com/support/>

# Oracle製品との動作保証

- Oracle Linux
- Oracle VM
- Oracle Solaris
- Oracle Clusterware
- Oracle Secure Backup
- Oracle Enterprise Manager
- Oracle Fusion Middleware
- Oracle GoldenGate
- Oracle Audit Vault & Database Firewall
- MyOracle Online Support

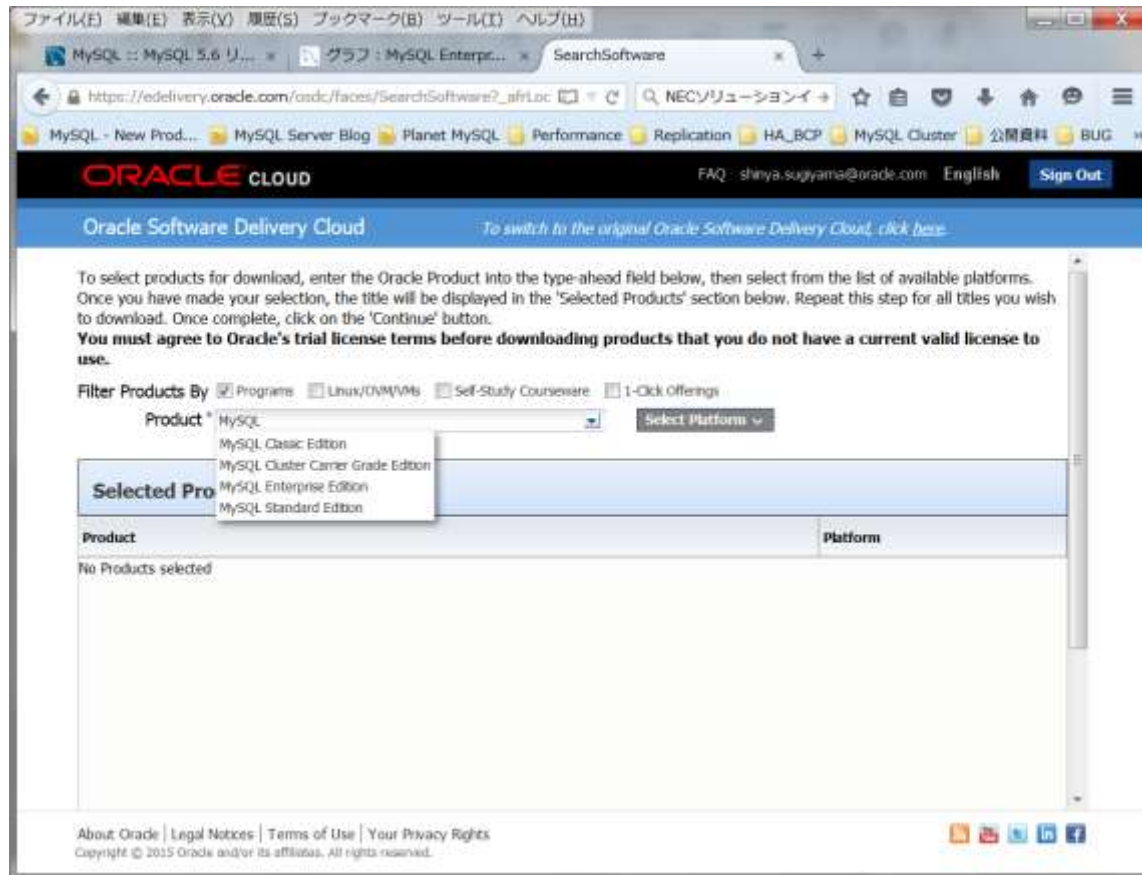
**MySQL Integrates into your Oracle Environment**



# MySQL Enterprise Editionの試用

# MySQL Enterprise Edition & Cluster CGEの試用

30日間トライアル



- Oracle Software Delivery Cloud

<http://edelivery.oracle.com/>

- 製品パックを選択:  
"Product" にMySQLと入力し、  
OSを選択し"Continue"

- 製品マニュアル

<http://dev.mysql.com/doc/index-enterprise.html>

# Integrated Cloud

## Applications & Platform Services



ORACLE®