



ついにWindow関数も！！ MySQL開発最新動向のご紹介

updated : 2017/11/10

Yoshiaki Yamasaki / 山崎 由章

MySQL Global Business Unit
MySQL Senior Sales Consultant

ORACLE

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 RC 新機能
- 3 MySQL Group Replication、MySQL InnoDB Cluster
- 4 MySQL Enterprise Edition
- 5 参考情報

アジェンダ

- 1 Oracle MySQL Cloud Service
- 2 MySQL 8.0 RC 新機能
- 3 MySQL Group Replication、MySQL InnoDB Cluster
- 4 MySQL Enterprise Edition
- 5 参考情報

Oracle MySQLクラウドサービスによる TCOの最適化





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: 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^{GB}

メモリー

170^{GB}

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 ?

Service Description ?

* SSH Public Key [編集](#) ?

* Compute Shape ?

Configuration

* Usable Database Storage (GB) ?

* Administration User ?

* Administration Password ?

* Confirm Administration Password ?

* Database Schema Name ?

Configure Enterprise Monitor ?

* Manager User ?

* Manager Password ?

* Confirm Manager Password ?

* Agent User ?

* Agent Password ?

* Confirm Agent Password ?

Backup and Recovery Configuration

Backup Destination ?

* Cloud Storage Container ?

* Cloud Storage User Name ?

* 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

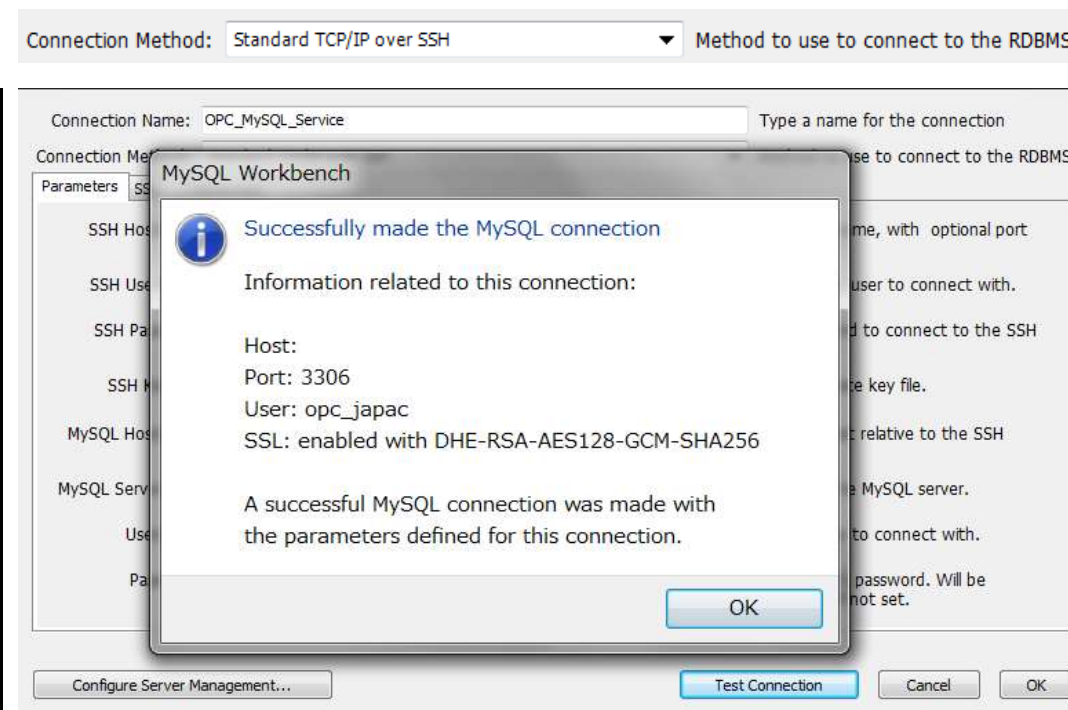
STEP3)

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。

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

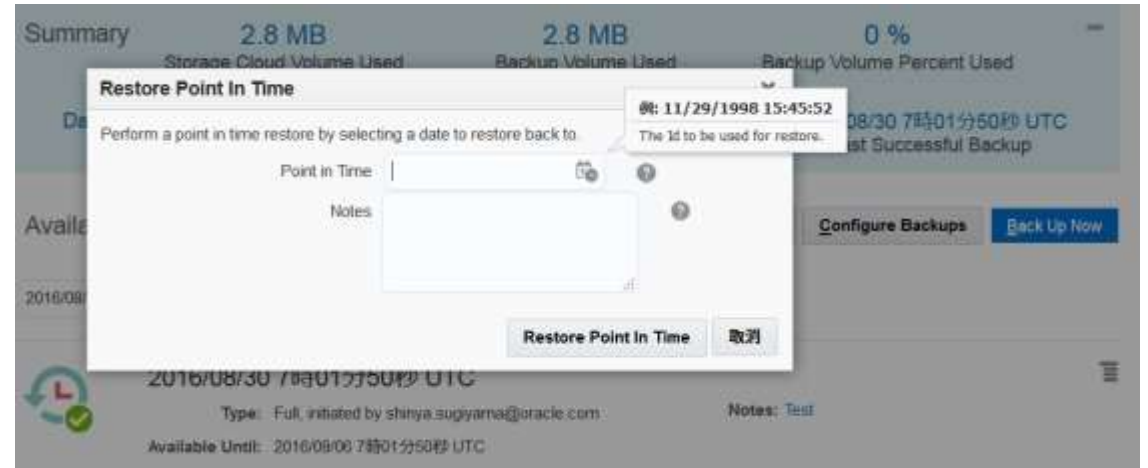
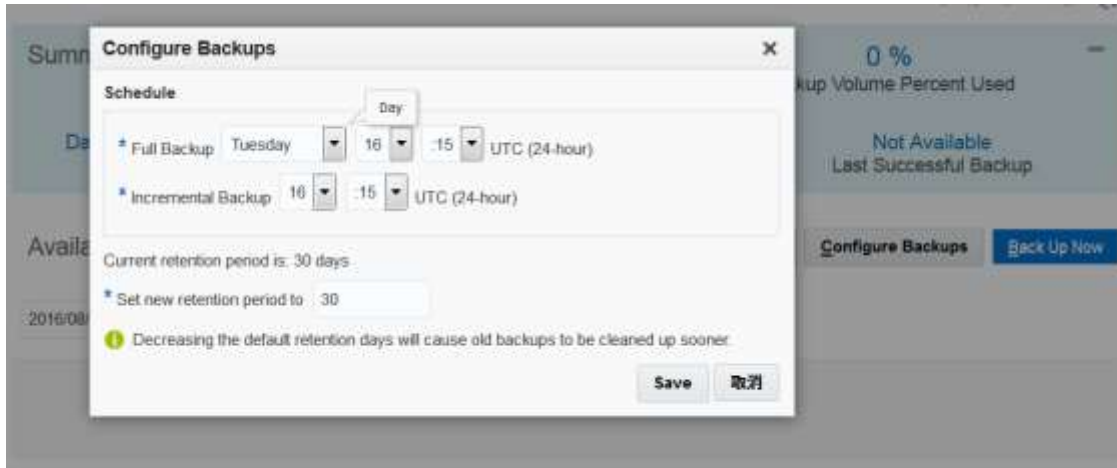
Available Backups

[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EXPORT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

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。

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



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/mysql-cloud/CSMCS/toc.htm>

<http://docs.oracle.com/en/cloud/iaas/compute-iaas-cloud/stcsa/toc.htm>



詳細情報 @ cloud.oracle.com/mysql

ORACLE Cloud

Sign In English Free Trial

Applications Platform Infrastructure Resources

MySQL Try It

Overview Features Pricing Learn More

MySQL in the Oracle Cloud for Your Enterprise Needs.

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.

Watch Video

MySQL™

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.

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 RC 新機能
- 3 MySQL Group Replication、MySQL InnoDB Cluster
- 4 MySQL Enterprise Edition
- 5 参考情報

RC

MySQL 8.0

MySQL 8.0 RC (リリース候補版)

- 2017年11月10日時点の最新版はMySQL 8.0.3 RC
- 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/>

The MySQL 8.0 Story

“Making MySQL Better for Modern Applications”

- Mobile First
- Developer Centric
- Premium placed on time to Market

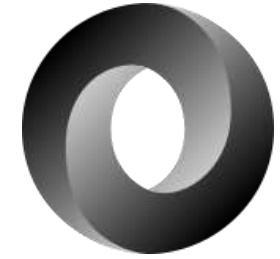
GIS機能の拡張



- Geography サポート
 - st_distance() 関数
- Spatial Data、Spatial Index、Spatial関数のSRID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)

JSON Functions

MySQL 5.7 and 8.0



JSON_ARRAY_APPEND()

JSON_ARRAY_INSERT()

JSON_ARRAY()

JSON_CONTAINS_PATH()

JSON_CONTAINS()

JSON_DEPTH()

JSON_EXTRACT()

JSON_INSERT()

JSON_KEYS()

JSON_LENGTH()

JSON_MERGE_[PATCH]()

JSON_OBJECT()

JSON_QUOTE()

JSON_REMOVE()

JSON_REPLACE()

JSON_SEARCH()

JSON_SET()

JSON_TYPE()

JSON_UNQUOTE()

JSON_VALID()

JSON_PRETTY()

JSON_STORAGE_SIZE()

JSON_STORAGE_FREE()

JSON_MERGE_PRESERVE()

JSON_ARRAYAGG()

JSON_OBJECTAGG()

JSON_TABLE() *labs

```
SELECT * FROM seats,  
  JSON_TABLE(doc, "$.properties.amenities[*]" COLUMNS (  
    id for ordinality,  
    amenity_type VARCHAR(100) PATH "$.type",  
    distance float PATH '$.distance_in_meters')  
  ) AS amenities  
WHERE seats.id = 28100  
  AND amenities.amenity_type IN ('snacks', 'bar')  
ORDER BY amenities.distance;
```

```
+-----+-----+-----+  
| id    | amenity_type    | distance |  
+-----+-----+-----+  
|  2    | bar             | 100.538 |  
|  3    | snacks          | 136.647 |  
+-----+-----+-----+  
2 rows in set (0.00 sec)
```

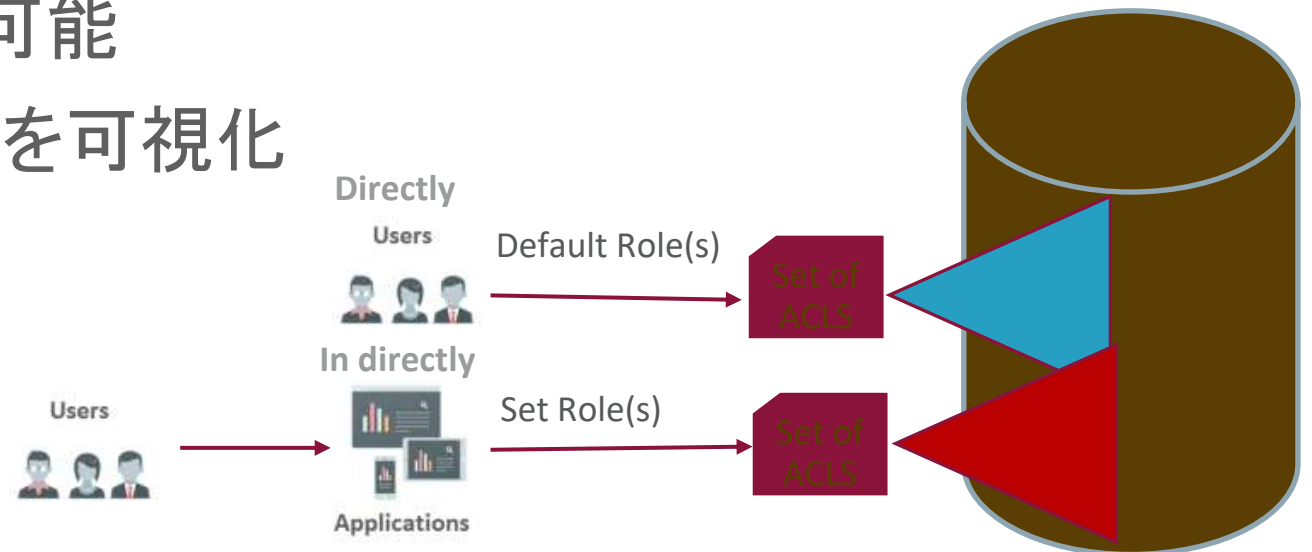
UTF-8

- 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
 - 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
 - Ks: カナセンシティブ('あ'、'ア'、'ァ'を区別)



ロール(権限をまとめて付与/剥奪)

- アクセス制御の改善
- ロール作成/削除、ロールへの権限付与
- ユーザー/ロールに対してロールの付与
- デフォルトロールを定義、特定のホストのみロールを使用可能
- ROLES_GRAPHML()関数でロールを可視化



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



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)

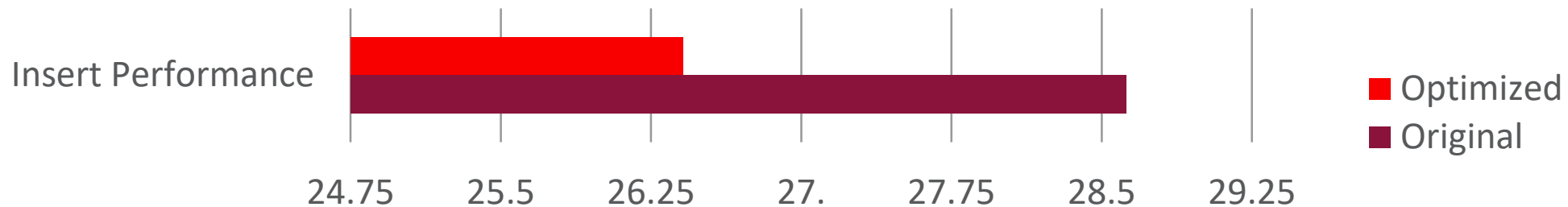


UUID_TO_BIN の最適化

- 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

Common Table Expressions (WITH句)

- WITH句
- 利点
 - 再帰的なクエリの記述
 - 複雑なSQLの簡素化
 - パフォーマンス向上

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



再帰的CTEのシンプルな例

Print 1 to 10 :

```
WITH RECURSIVE qn AS
( SELECT 1 AS a
  UNION ALL
  SELECT 1+a FROM qn WHERE a<10
)
SELECT * FROM qn;
```

```
1
2
3
4
5
6
7
8
9
10
```

Window関数



Window関数の例: 各部門ごとの合計給与を計算

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

Name	dept_id	salary	dept_total
Newt	NULL	75000	75000
Dag	10	NULL	370000
Ed	10	100000	370000
Fred	10	60000	370000
Jon	10	60000	370000
Michael	10	70000	370000
Newt	10	80000	370000
Lebedev	20	65000	130000
Pete	20	65000	130000
Jeff	30	300000	370000
Will	30	70000	370000

Window関数の種類

- 集計
 - COUNT, SUM, AVG, MAX, MIN + more to come
- ランキング
 - RANK, DENSE_RANK, PERCENT_RANK,
 - CUME_DIST, ROW_NUMBER
- 分析
 - NTILE, LEAD, LAG
 - NTH_VALUE, FIRST_VALUE, LAST_VALUE

新しいオプティマイザー・コストモデル

・バッファプールのヒット率を意識した改善



```
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、全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コストが低い
※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。
INNODB_CACHED_INDEXESからヒット率を判断し、適切な実行計画を選択

ヒストグラム

- データが偏っている場合のクエリーの精度向上
- ヒストグラムはインデックスを作成するよりもコストが低い
- 適応ケース
 - JOIN対象行数の正確な見積り
 - より適切なJOIN順序の判断
 - インデックスが付いている列で絞り込んだ時に、指定する値によってインデックススキャン/テーブルスキャンを切り替える(※)



※MySQL 8.0.3時点ではインデックスがついている列にヒストグラムが取得されないため未対応。
今後拡張予定。

降順索引 (Descending Indexes)

For B+tree indexes

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

```
mysql 8.0> EXPLAIN SELECT * FROM t1 ORDER BY a DESC, b ASC;
```

```
.. +-----+-----+-----+-----+-----+-----+  
.. | key          | key_len | ref  | rows | filtered | Extra      |  
.. +-----+-----+-----+-----+-----+-----+  
.. | a_desc_b_asc | 10      | NULL | 10   | 100.00   | Using index |  
.. +-----+-----+-----+-----+-----+-----+
```

索引a_desc_b_ascにより、
Using filesortを避けられている

```
mysql 8.0> EXPLAIN SELECT * FROM t1 ORDER BY a ASC, b ASC;
```

```
.. +-----+-----+-----+-----+-----+-----+  
.. | key          | key_len | ref  | rows | filtered | Extra      |  
.. +-----+-----+-----+-----+-----+-----+  
.. | a_desc_b_asc | 10      | NULL | 10   | 100.00   | Using index; Using filesort |  
.. +-----+-----+-----+-----+-----+-----+
```

降順索引 (Descending Indexes)

For B+tree indexes

以下構成でのテスト結果 ※MySQL 5.7はASC、ASCの索引のみが存在

```
CREATE TABLE t1  
(a INT,b INT,  
 INDEX a_desc_b_asc (a DESC, b ASC),  
 INDEX a_asc_b_asc (a ASC, b ASC),  
 INDEX a_asc_b_desc (a ASC, b DESC),  
 INDEX a_desc_b_desc (a DESC, b DESC)  
);
```



※詳細情報

<http://mysqlserverteam.com/mysql-8-0-labs-descending-indexes-in-mysql/>

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Invisible Indexes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
- Invisible Indexesは索引採用のテスト容易にする
 - 準備段階でInvisible Indexes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
 - SQL文単位でInvisible Indexesを有効化する方法は、次々ページ参照

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


Invisible Indexes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
```

オプティマイザヒントの拡張

- SET_VARヒント

- SQL単位でシステム変数を変更できるヒント
- セッション単位で変更可能なシステム変数をSQL単位で変更可能に
(max_allowed_packetなど一部のセッション変数は変更不可)

使用例

```
SELECT /*+ SET_VAR(sort_buffer_size = 16M) */ name FROM people ORDER BY name;
INSERT /*+ SET_VAR(foreign_key_checks=OFF) */ INTO t2 VALUES(2);
SELECT /*+ SET_VAR(optimizer_switch='use_invisible_indexes=ON') */ name,region
FROM country WHERE region='Eastern Asia';
```

オプティマイザヒントの拡張

- JOIN_FIXED_ORDERヒント
 - STRAIGHT_JOINヒントの置き換え
 - STRAIGHT_JOINヒントのようにFROM句を修正する必要なし！
- JOIN_ORDER
 - JOINの順番を指定
- JOIN_PREFIX
 - 最初にJOINするテーブルを指定
- JOIN_SUFFIX
 - 最後にJOINするテーブルを指定

オプティマイザヒントの拡張

- INDEX_MERGE/NO_INDEX_MERGEヒント
 - インデックスマージを選択させる/選択させない
- MERGE/NO_MERGEヒント
 - 外部クエリーブロックへderivedテーブル/ビューをマージさせる/させない
- RESOURCE_GROUPヒント
 - リソースグループを指定

※MySQL 5.0以降は、オプティマイザがインデックスマージも選択可能になっています。
また、MySQL5.6以降はインデックスマージの制限が緩和され、以前のバージョンよりもインデックスマージを選択しやすくなっています。

パフォーマンス・スキーマ・ヒストグラム

例: mysqlslap実行時のクエリー実行時間の分布



Feature Request
from DBAs

bucket	visualization	count
0us+	#	1253
10us+	#####	43102
100us+	#####	25013
1ms+	#	2003
10ms+		325
100ms+		17
1s+		0
10s+		0

8 rows in set (0.08 sec)

```
query: INSERT INTO `t1` VALUES (...)  
db: mysqlslap  
total_latency: 54.43 s  
exec_count: 58377  
lock_latency: 1.70 s
```

..

```
digest: 4e0c5b796c4052b0da4548fd7cb694be  
first_seen: 2017-04-16 20:59:16  
last_seen: 2017-04-16 21:00:34
```

latency_distribution:

```
0us+  
10us+ #####  
100us+ #####  
1ms+ #  
10ms+  
100ms+  
1s+  
10s+
```

events_statements_histogram_globalと
CTEを利用して実行時間の分布をビジュ
アライズした例

クエリー単位の実行時間の
分布も確認可能

パフォーマンス・スキーマの拡張

ロック発生状況の確認

```
SELECT thread_id, object_name, index_name, lock_type, lock_mode, lock_data  
FROM performance_schema.data_locks WHERE object_name = 'seats';
```

thread_id	object_name	index_name	lock_type	lock_mode	lock_data
33	seats	NULL	TABLE	IX	NULL
33	seats	PRIMARY	RECORD	X	3, 5
33	seats	PRIMARY	RECORD	X	3, 6
33	seats	PRIMARY	RECORD	X	4, 5
33	seats	PRIMARY	RECORD	X	4, 6

5 rows in set (0.00 sec)

※data_locks導入に伴い、以下のテーブルは非推奨となり削除されたため注意

- INFORMATION_SCHEMA.INNODB_LOCKS
- INFORMATION_SCHEMA.INNODB_LOCK_WAITS



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 max_connections = 500;`
 - `SET PERSIST_ONLY innodb_log_file_size = 128*1024*1024;`
- 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`)

設定変更の永続化

設定変更の情報を確認可能

```
SELECT * FROM performance_schema.variables_info WHERE variable_source != 'COMPILED';
```

VARIABLE_NAME	VARIABLE_SOURCE	VARIABLE_PATH	MIN_VALUE	MAX_VALUE	SET_TIME	SET_USER	SET_HOST
autocommit	DYNAMIC		0	0	2017-04-16 20:56:53	msandbox	localhost
basedir	COMMAND_LINE		0	0	2017-04-16 21:08:11		
bind_address	EXPLICIT	[..]my.sandbox.cnf	0	0	2017-04-16 21:08:11		
character_set_client	DYNAMIC		0	0	2017-04-16 20:56:53	msandbox	localhost
character_set_results	DYNAMIC		0	0	2017-04-16 20:56:53	msandbox	localhost
collation_connection	DYNAMIC		0	0	2017-04-16 20:56:53	msandbox	localhost
datadir	COMMAND_LINE		0	0	2017-04-16 21:08:11		
foreign_key_checks	DYNAMIC		0	0	2017-04-16 20:56:53	msandbox	localhost
log_error	COMMAND_LINE		0	0	2017-04-16 21:08:11		
lower_case_table_names	EXPLICIT	[..]my.sandbox.cnf	0	2	2017-04-16 21:08:11		
pid_file	COMMAND_LINE		0	0	2017-04-16 21:08:11		
plugin_dir	COMMAND_LINE		0	0	2017-04-16 21:08:11		
port	COMMAND_LINE		0	65535	2017-04-16 21:08:11		
socket	COMMAND_LINE		0	0	2017-04-16 21:08:11		
sql_mode	DYNAMIC		0	0	2017-04-16 20:56:53	msandbox	localhost
sql_notes	DYNAMIC		0	0	2017-04-16 20:56:53	msandbox	localhost
time_zone	DYNAMIC		0	0	2017-04-16 20:56:53	msandbox	localhost
tmpdir	EXPLICIT	[..]my.sandbox.cnf	0	0	2017-04-16 21:08:11		
unique_checks	DYNAMIC		0	0	2017-04-16 20:56:53	msandbox	localhost

19 rows in set (0.00 sec)

トランザクショナルなデータディクショナリ

- クラッシュセーフなデータベース
 - サーバーとInnoDBで共通のデータディクショナリ
- クラッシュセーフ&アトミックなDDL
 - 一連のメンテナンス操作をアトミックに処理可能
 - 例) CREATE USER <userlist>, DROP DATABASE
 - レプリケーション障害への対応をシンプルに
- 外部キー(FK)のメタデータ
 - InnoDBからサーバーレイヤーに移行
- スケーラブルなインフォメーションスキーマ
 - テーブル上に作成されたビューとして動作するようになった
 - パフォーマンスの大幅な向上

デフォルト値の変更

サーバーコア

- event_scheduler=ON
- デフォルトキャラクタセット
 - utf8mb4
- パフォーマンス・スキーマがデフォルトで取得する情報を追加
 - Memory
 - Transactions
 - MDL
- InnoDB
 - autoinc_lock_mode=2
 - flush_neighbors=0
 - max_dirty_pages_pct_lwm=10
 - max_dirty_pages_pct default=90
- back_log
 - 自動計算方法の改善
- max_error_count=1024
- max_allowed_packet=64M

デフォルト値の変更

レプリケーション

- log_bin=ON
- log_slave_updates=ON
- master_info_repository=TABLE
- relay_log_info_repository=TABLE
- transaction_write_set_extraction=XXHASH64
- slave_rows_search_algorithms='INDEX_SCAN,HASH_SCAN'
- expire_logs_days=30 (※)
- server_id=1

※現在は非推奨。別途binlog_expire_logs_secondsが存在し、8.0.3時点では

「expire_logs_days+binlog_expire_logs_seconds」が有効になる。binlog_expire_logs_secondsのデフォルト値は0。

InnoDB Dedicated Server

- システムメモリー量を確認し、ログファイルサイズ、バッファプールサイズ、フラッシュメソッドを自動的に調整
- 使用方法

```
SET PERSIST_ONLY innodb_dedicated_server = TRUE
```

InnoDB Dedicated Server

- MySQL 8.0.3での設定内容

```
[innodb_buffer_pool_size]
server_memory < 1G ? 128M (現在のデフォルトと同じ)
server_memory <= 4G ? server_memory * 0.5
server_memory > 4G ? server_memory * 0.75
```

```
[innodb_log_file_size]
server_memory < 1G ? 48M (現在のデフォルトと同じ)
server_memory <= 4G ? 128M
server_memory <= 8G ? 512M
server_memory <= 16G ? 1024M
server_memory > 16G ? 2048M
```

```
[innodb_flush_method]
O_DIRECT_NO_FSYNC
```

レプリケーション関連

- バイナリログにメタデータ追加
 - ログ内容の解析や他システムでのバイナリログ利用の簡素化
 - Transaction length, Commit timestamps, Table map Typesなど
- マルチソース・レプリケーション・フィルター
- JSONドキュメントの部分レプリケーション
- デフォルト設定の変更
 - バイナリログ取得およびスレーブでの変更点のログ記録(log_slave_updates)がONに
 - バイナリログを30日後に自動削除
 - binlog_expire_logs_secondsで設定、expire_logs_days非推奨
 - master.infoおよびrelay-log.info非推奨 テーブルでのメタデータ管理がデフォルト

他にも多くの点を改良

- Source code now documented with Doxygen
- New Plugin Infrastructure
- Improved BLOB Storage
- Improved Memcached Interface
- InnoDB Auto Increment Persists
- Parser Refactoring
- Improvements to Temporary Tables
- C++11 and Toolchain Improvements
- GTID_PURGED always settable
- Undo tablespaces now dynamic
- In memory storage engine
- InnoDB Cats lock scheduler algorithm
- Improved Parallel Replication
- SQL Grouping Function
- Optimizer Trace detailed sort statistics
- Smaller Package Downloads
- Improved usability of cost constant configuration

アジェンダ

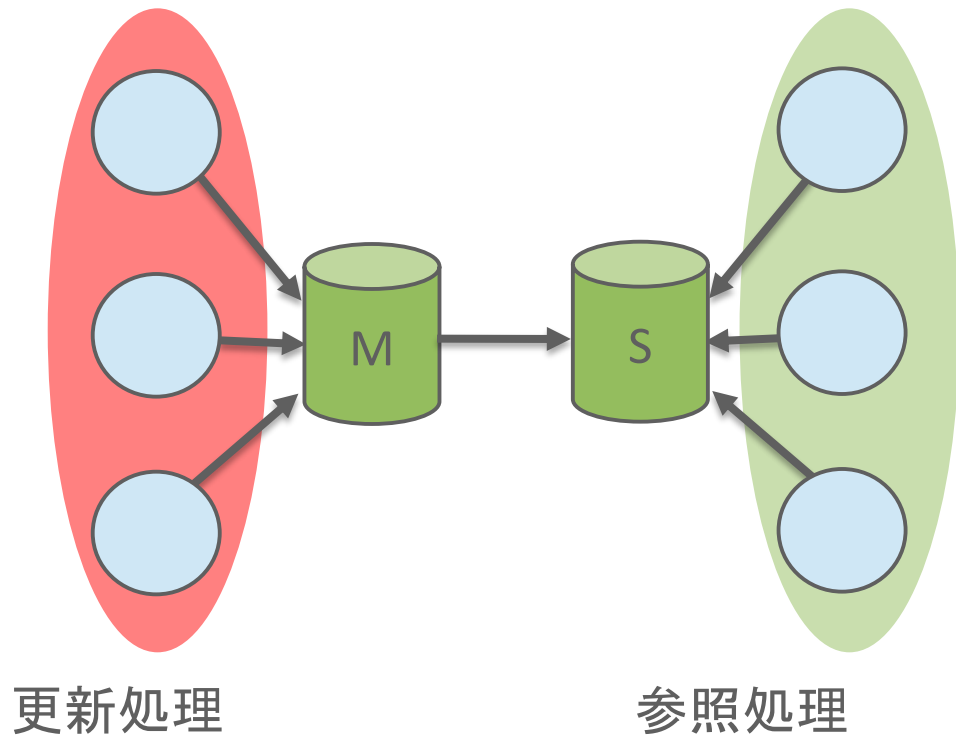
- 1 Oracle MySQL Cloud Service
- 2 MySQL 8.0 RC 新機能
- 3 **MySQL Group Replication、MySQL InnoDB Cluster**
- 4 MySQL Enterprise Edition
- 5 参考情報

グループレプリケーションとは？

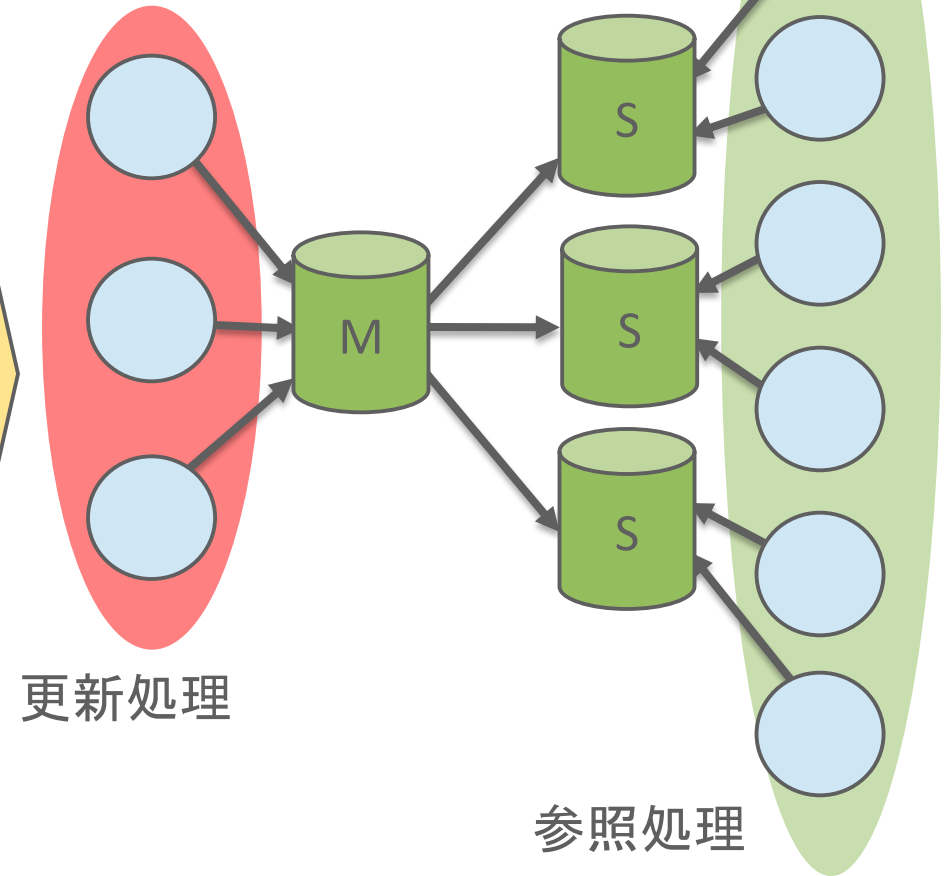
- MySQL5.7以降で利用可能な仮想同期レプリケーション
- MySQLがサポートする全てのプラットフォームに対応
 - Linux, Windows, Solaris, OSX, FreeBSD
- 手軽に高可用性構成を実現可能
 - 複数台でグループを組み、全台が同じデータを持つ
 - 3台以上の奇数でグループを構成することを推奨
- シングルプライマリーモード(デフォルト)とマルチマスターモードが使用可能

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

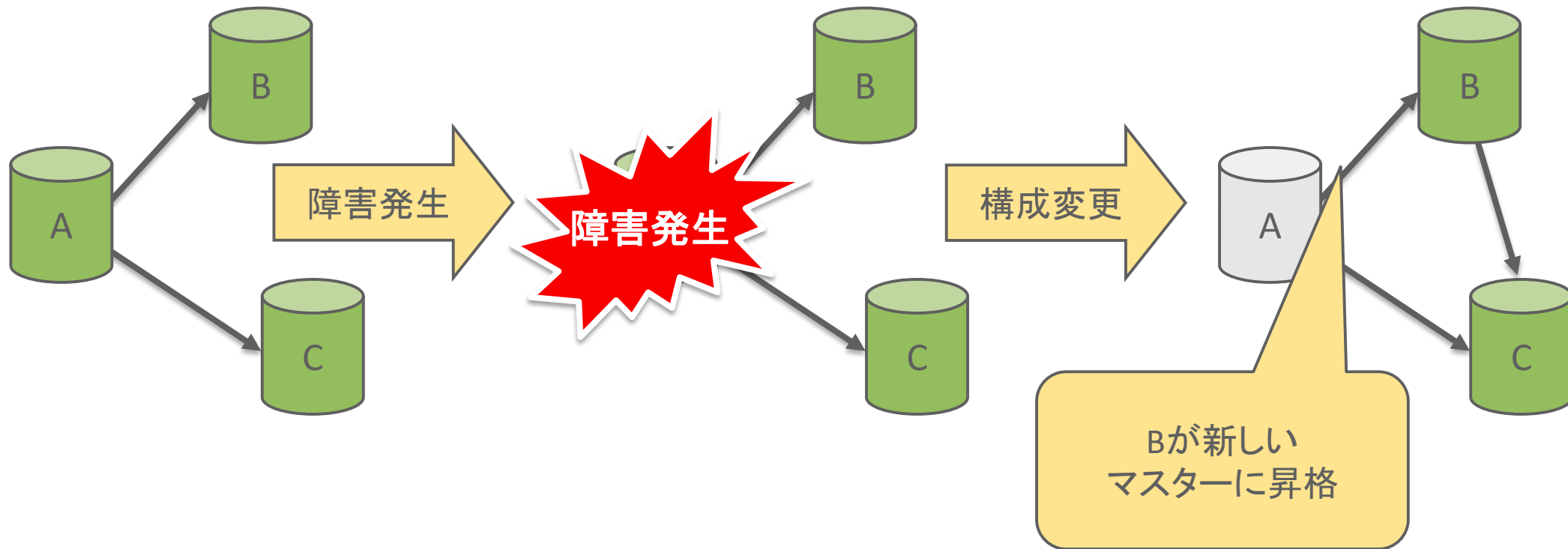


参照増加?
スレーブ
追加!



補足: レプリケーションの用途

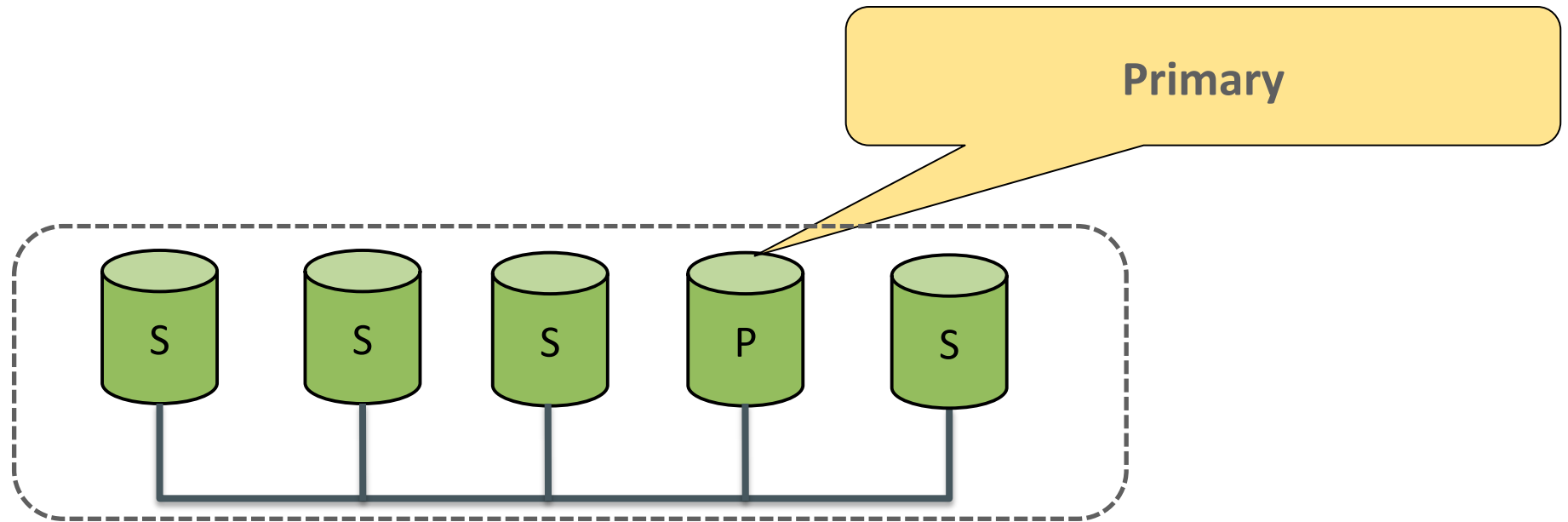
冗長性: マスターがクラッシュした場合, スレーブをマスターに昇格



シングルプライマリーモード

- 自動的なリーダー選択メカニズム

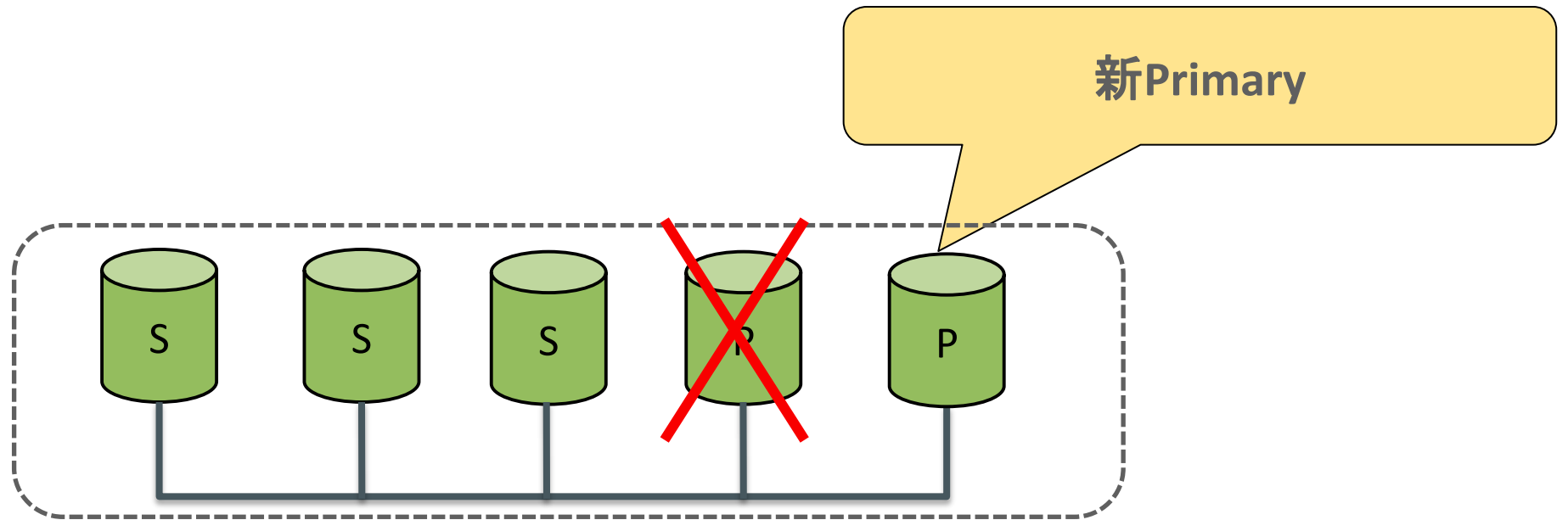
- Secondaryノードは参照のみ可能 (書き込んだ場合: ERROR 1290)



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

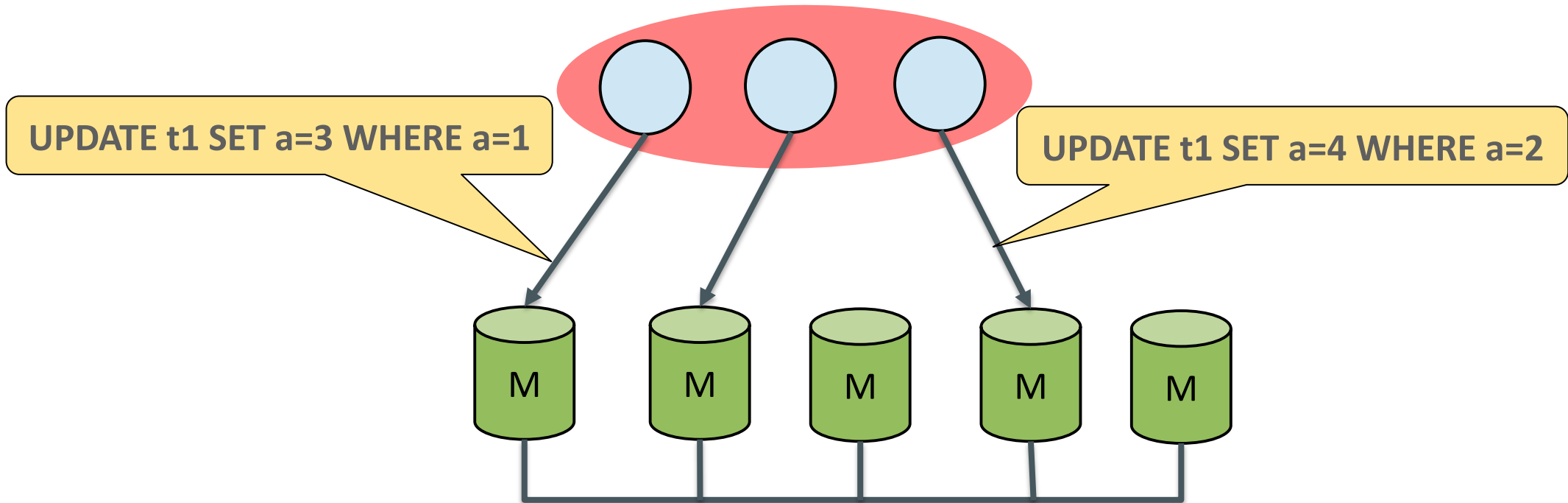
シングルプライマリーモード

- 自動的なリーダー選択メカニズム
 - 障害発生時は自動的にフェイルオーバー



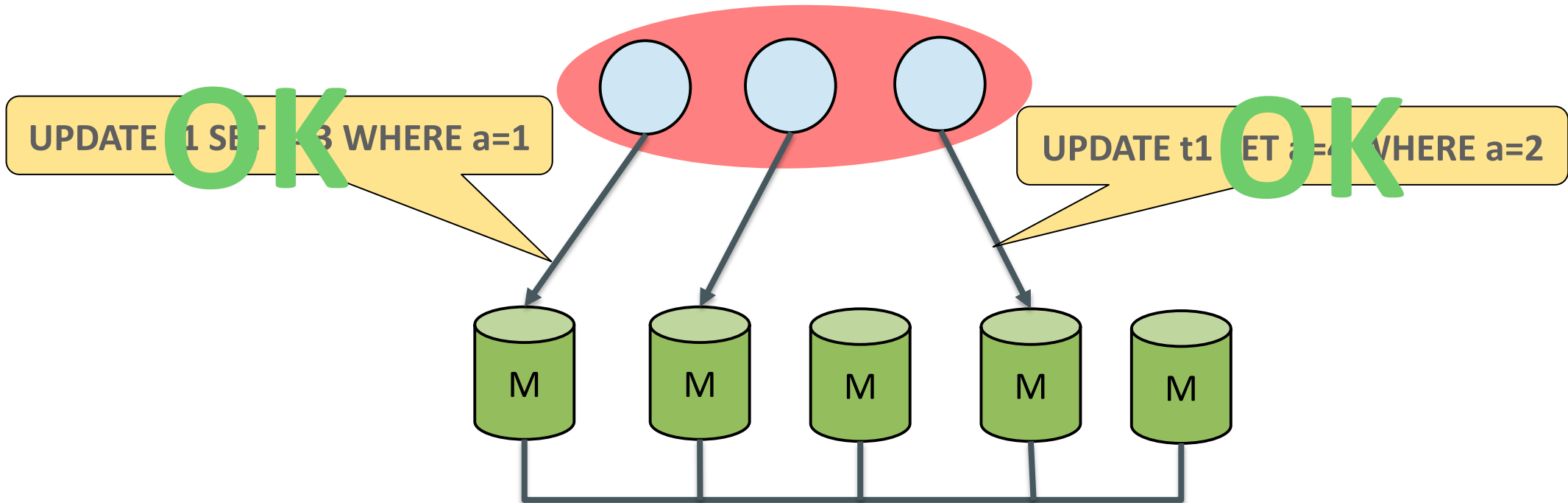
マルチマスターモード:どこでも更新可能

- 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



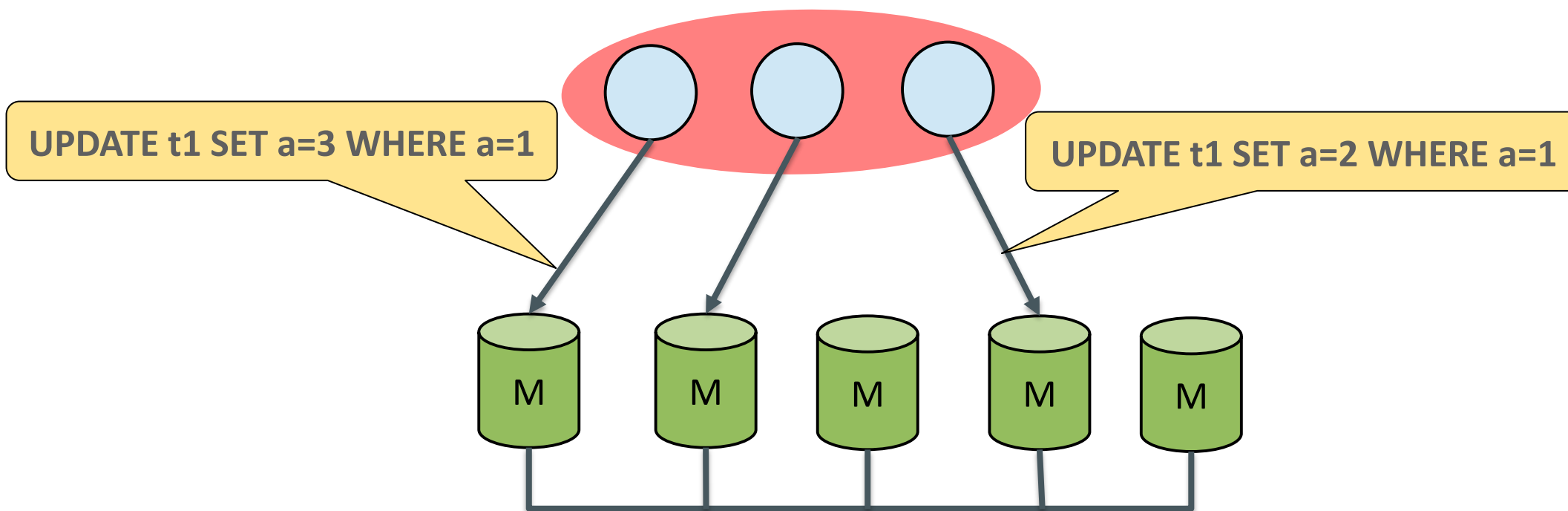
マルチマスターモード:どこでも更新可能

- 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



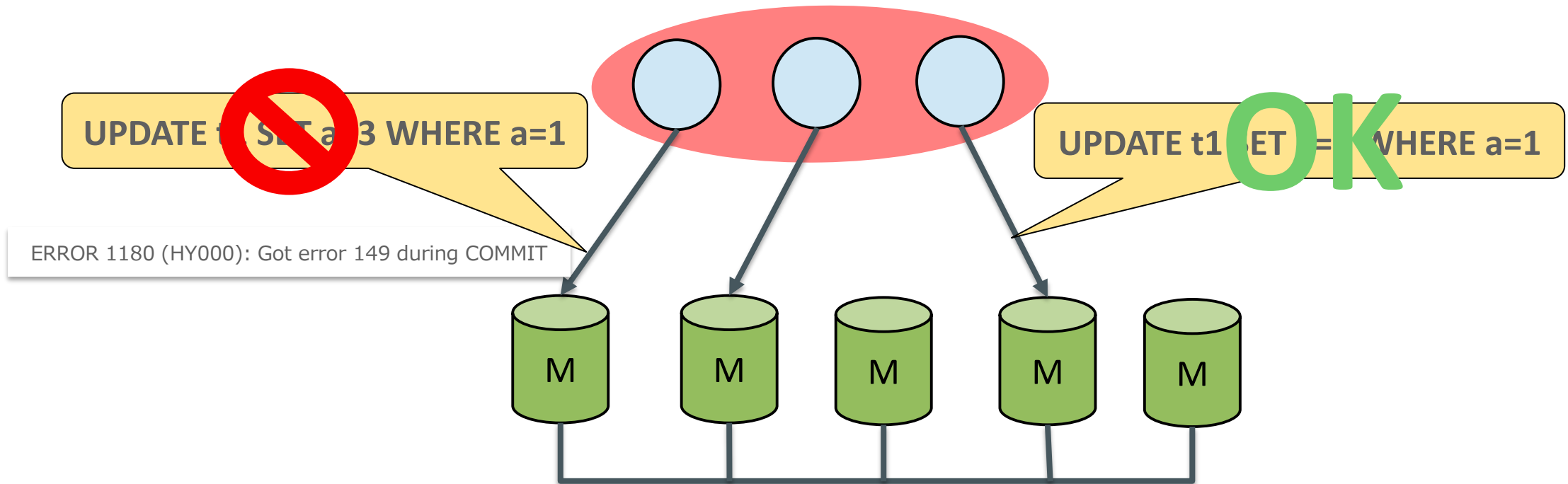
マルチマスターモード:どこでも更新可能

- 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

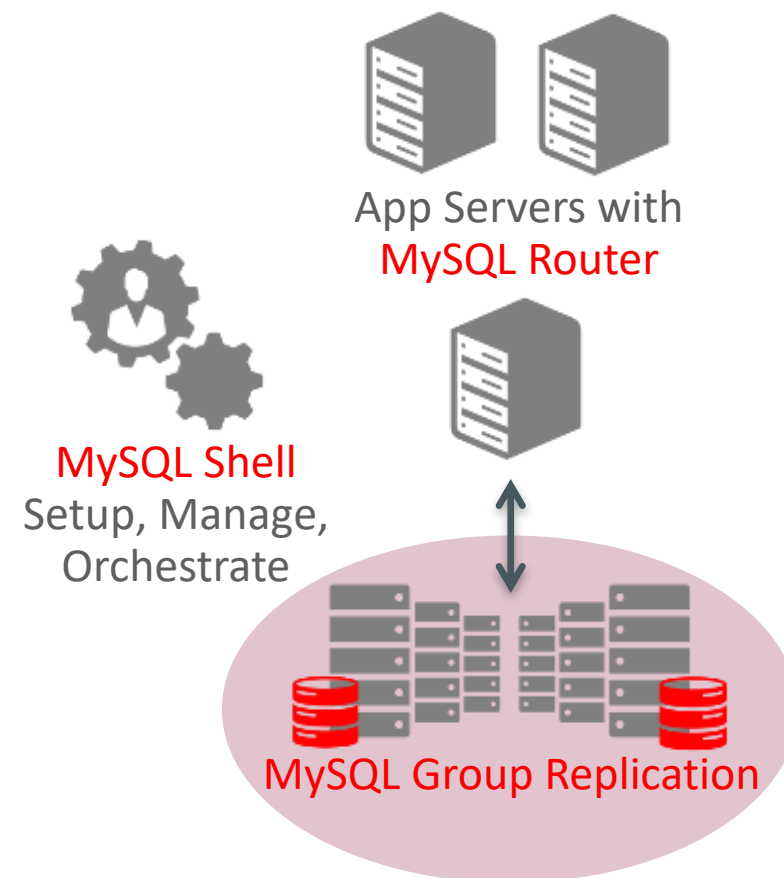
- 異なるサーバー上での2つのトランザクションは、同じデータを更新可能
- 競合が検出された場合、自動的に対処される
 - 先にコミットしたトランザクションが優先される



MySQL InnoDB Cluster

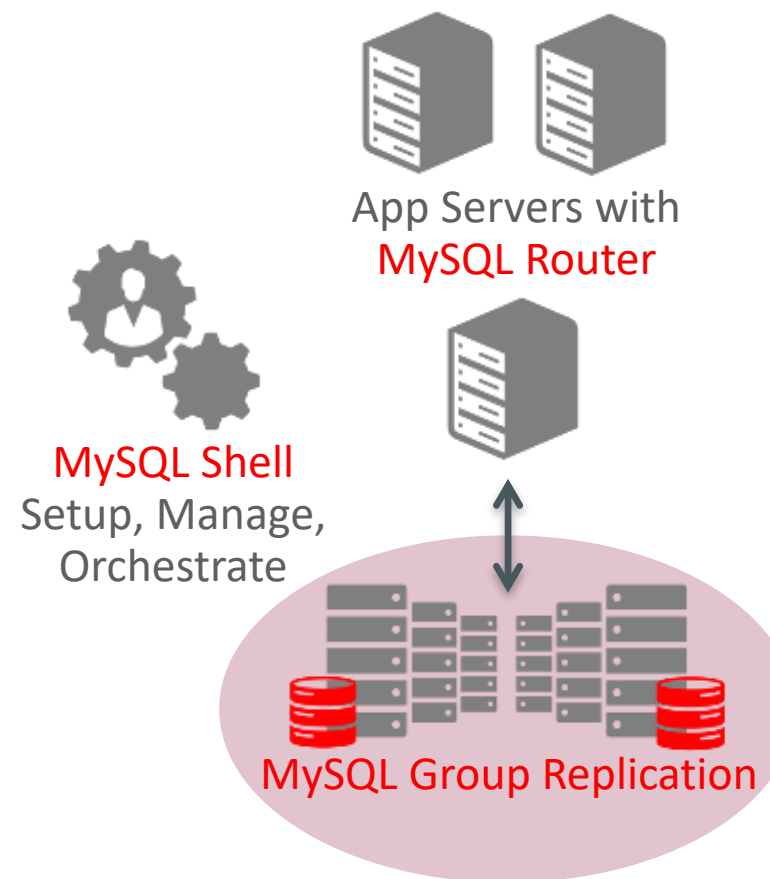
MySQL InnoDB Clusterとは？

- 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 5.7.20 (2017-10-16)
 - MySQL Router 2.1.4 (2017-07-24)
 - MySQL Shell 1.0.10 (2017-07-28)



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

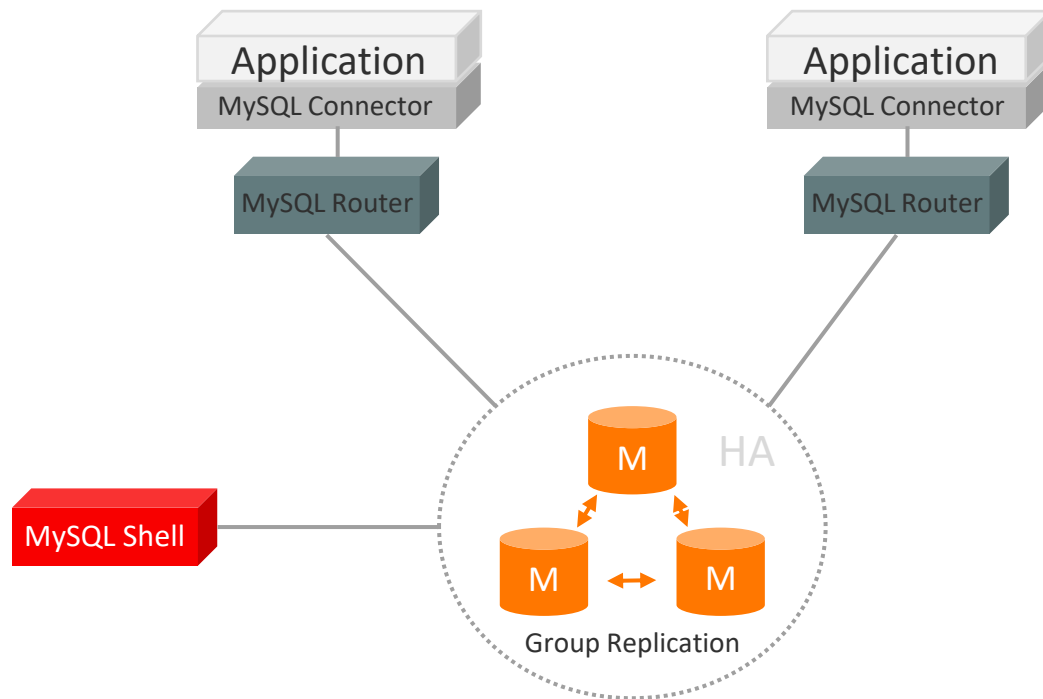
Out-of-Box Solution

- 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

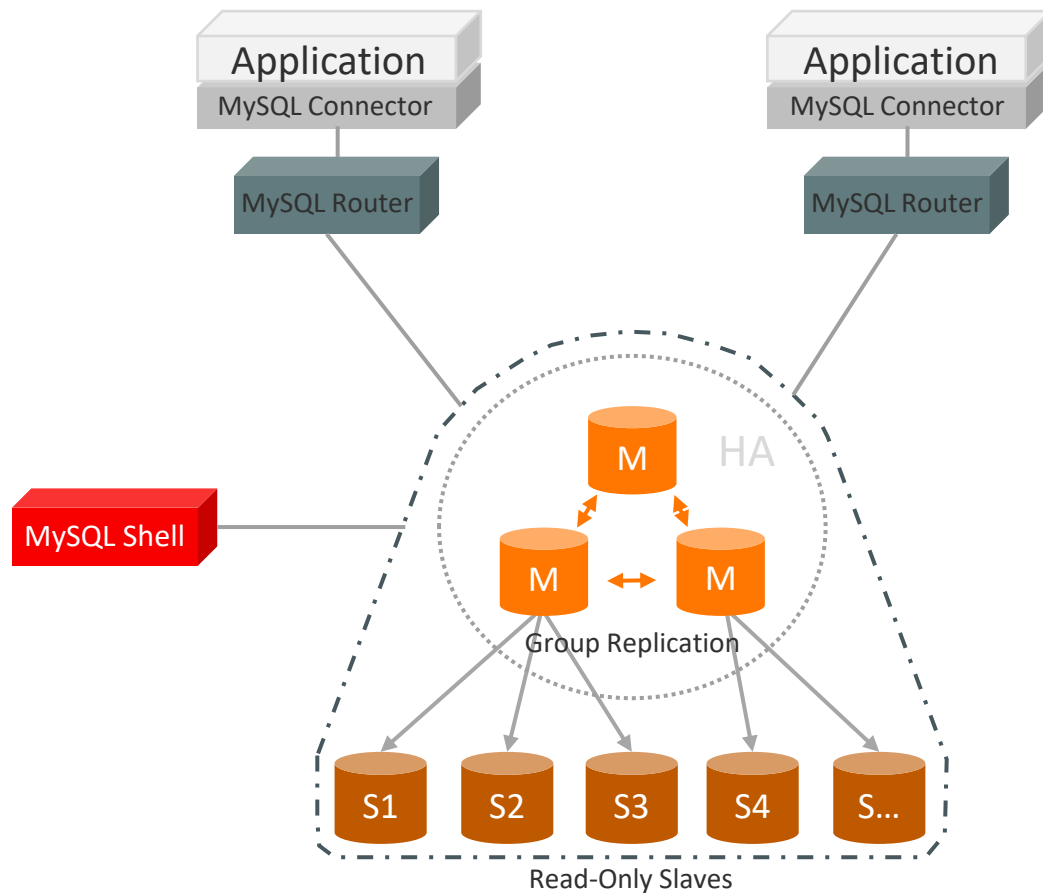
Scale-Out

- 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

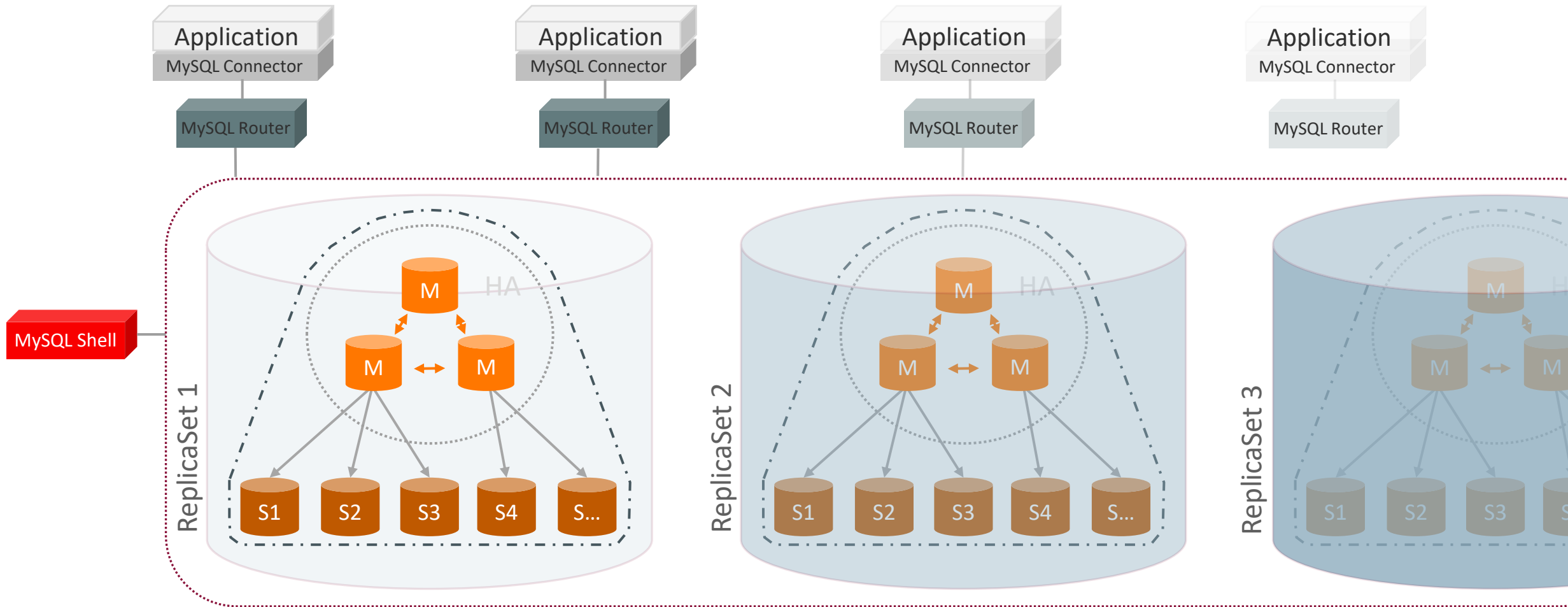
MySQL InnoDB Cluster: Architecture – Step 1



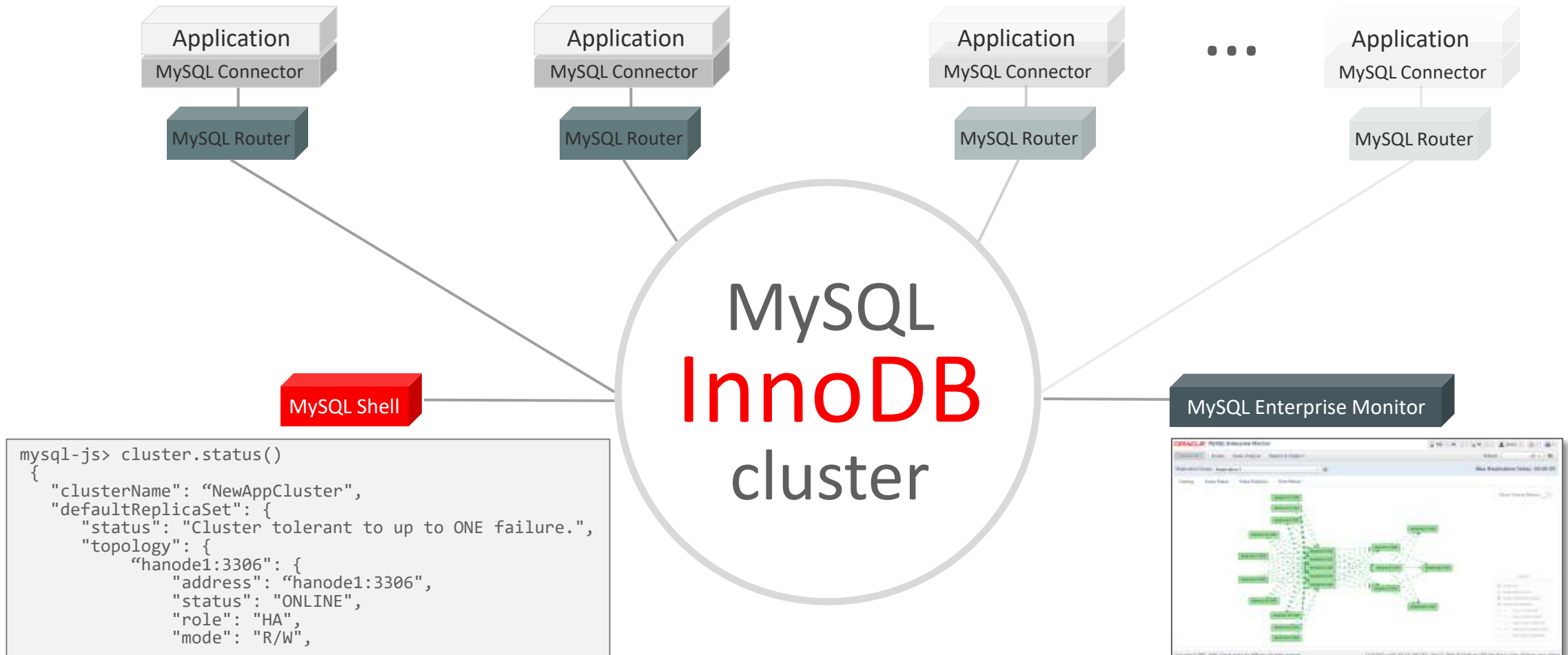
MySQL InnoDB Cluster: Architecture – Step 2



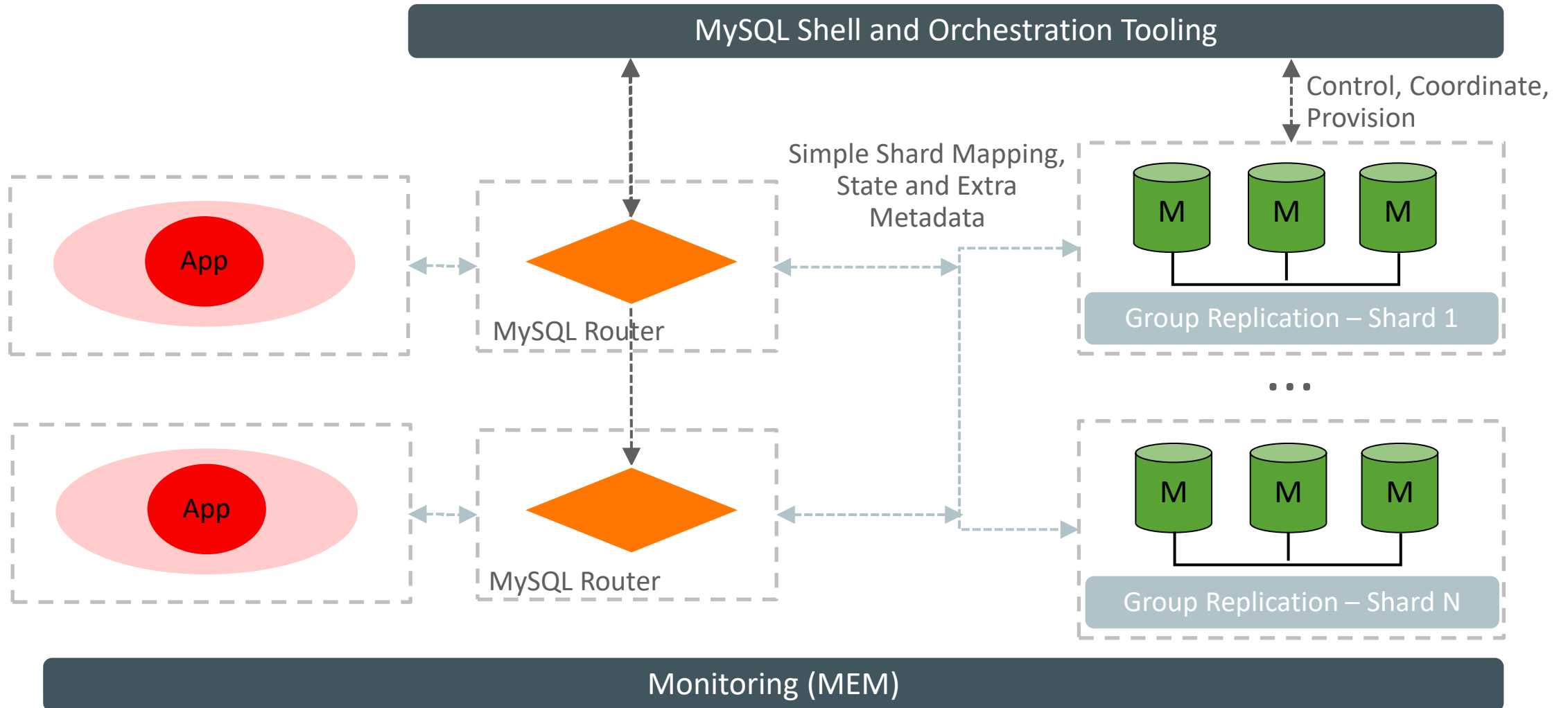
MySQL InnoDB Cluster: Architecture – Step 3



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>

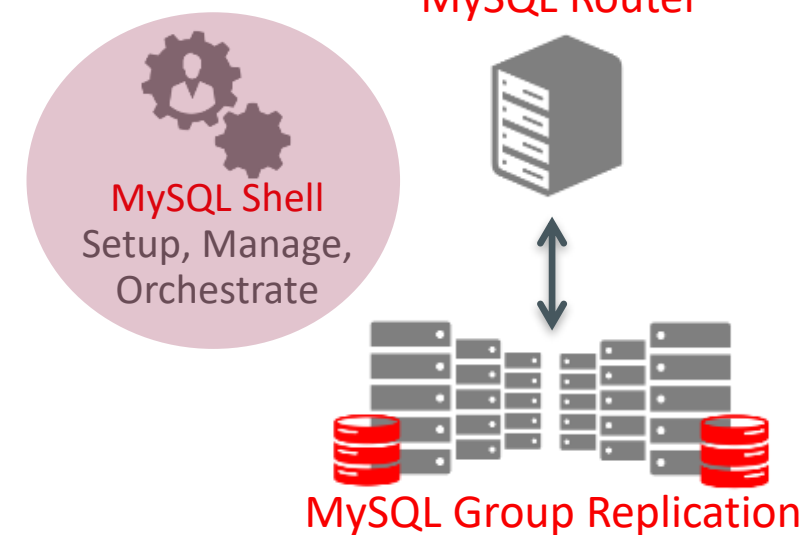
MySQL Shell

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!"

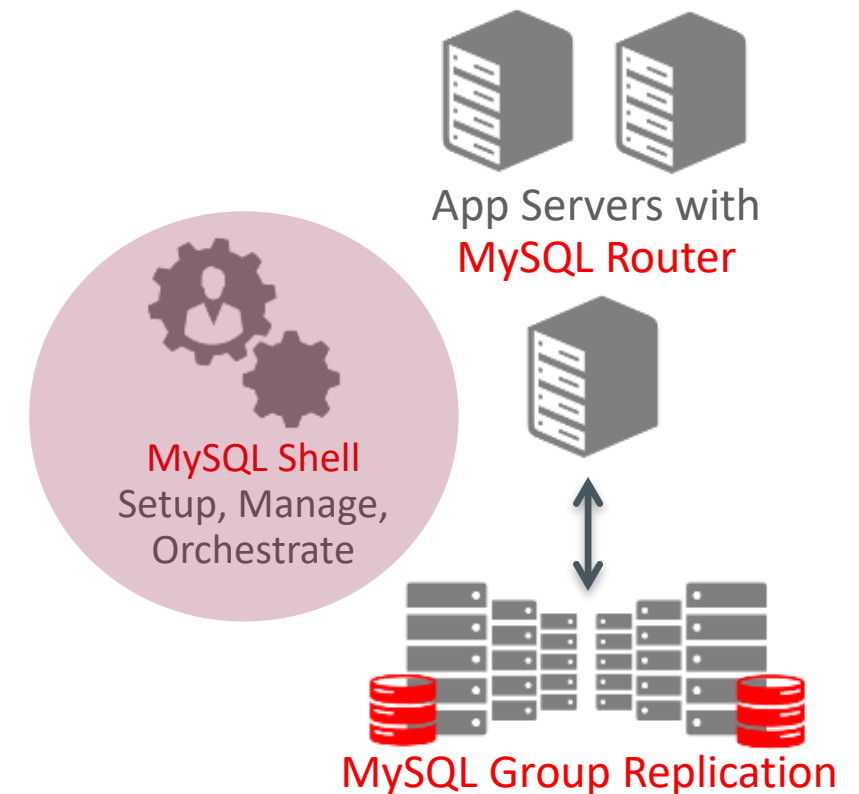
```
[root@misc01 admin]# mysqlsh --help | egrep -i "Start in"
--sql          Start in SQL mode using a node session.
--sqlc        Start in SQL mode using a classic session.
--js          Start in JavaScript mode.
--py          Start in Python mode.
[root@misc01 admin]#
```



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 – Add MySQL Router

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

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 – 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 RC 新機能
- 3 MySQL Group Replication、MySQL InnoDB Cluster
- 4 MySQL Enterprise Edition
- 5 参考情報

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・顧客満足の上



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

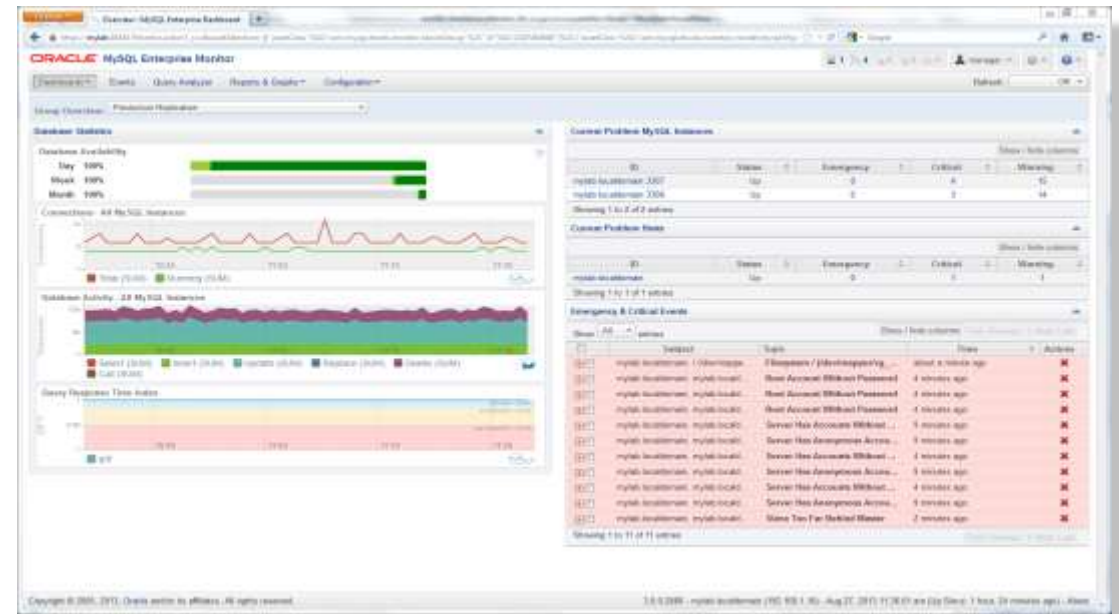
MySQL Enterprise Monitor	複数サーバの一括管理、クエリ性能分析
MySQL Enterprise Backup	高速なオンラインバックアップ、ポイントインタイムリカバリ
MySQL Enterprise Authentication	LDAPやWindows Active Directoryとの外部認証と統合管理
MySQL Enterprise TDE	データベース全体の暗号化(透過的)
MySQL Enterprise Encryption	非対称暗号化(公開鍵暗号)の業界標準機能を提供
MySQL Enterprise Firewall	SQLインジェクション対策
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](#)

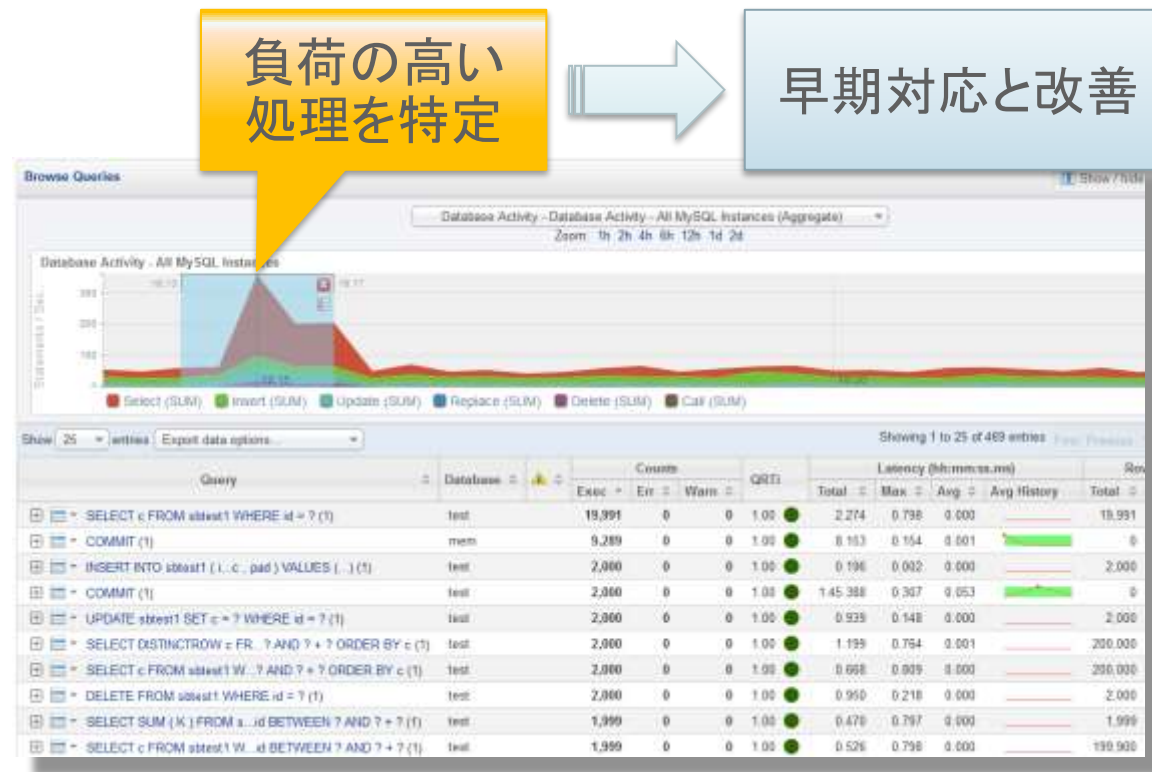
Diagnostic Reportを利用してサポートとの情報共有を容易にする事も可能



"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

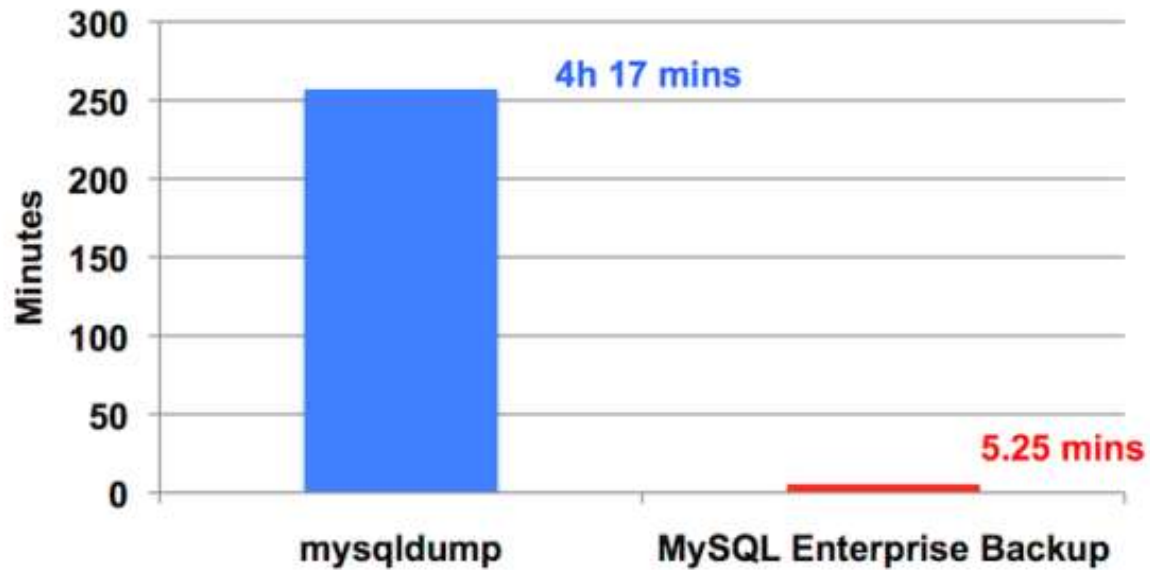
MySQL Enterprise Backup の特徴と利点



高速なバックアップとリカバリー

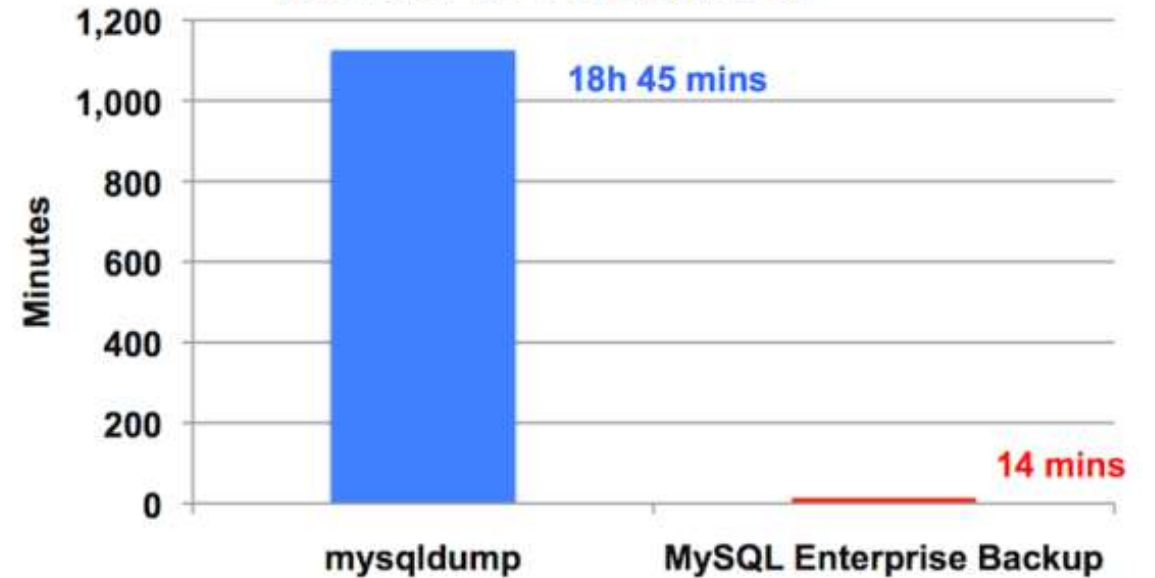
高速なオンラインバックアップ&リカバリー処理により、機会損失を最小限に抑える事が可能

Backup: 73 GB Database



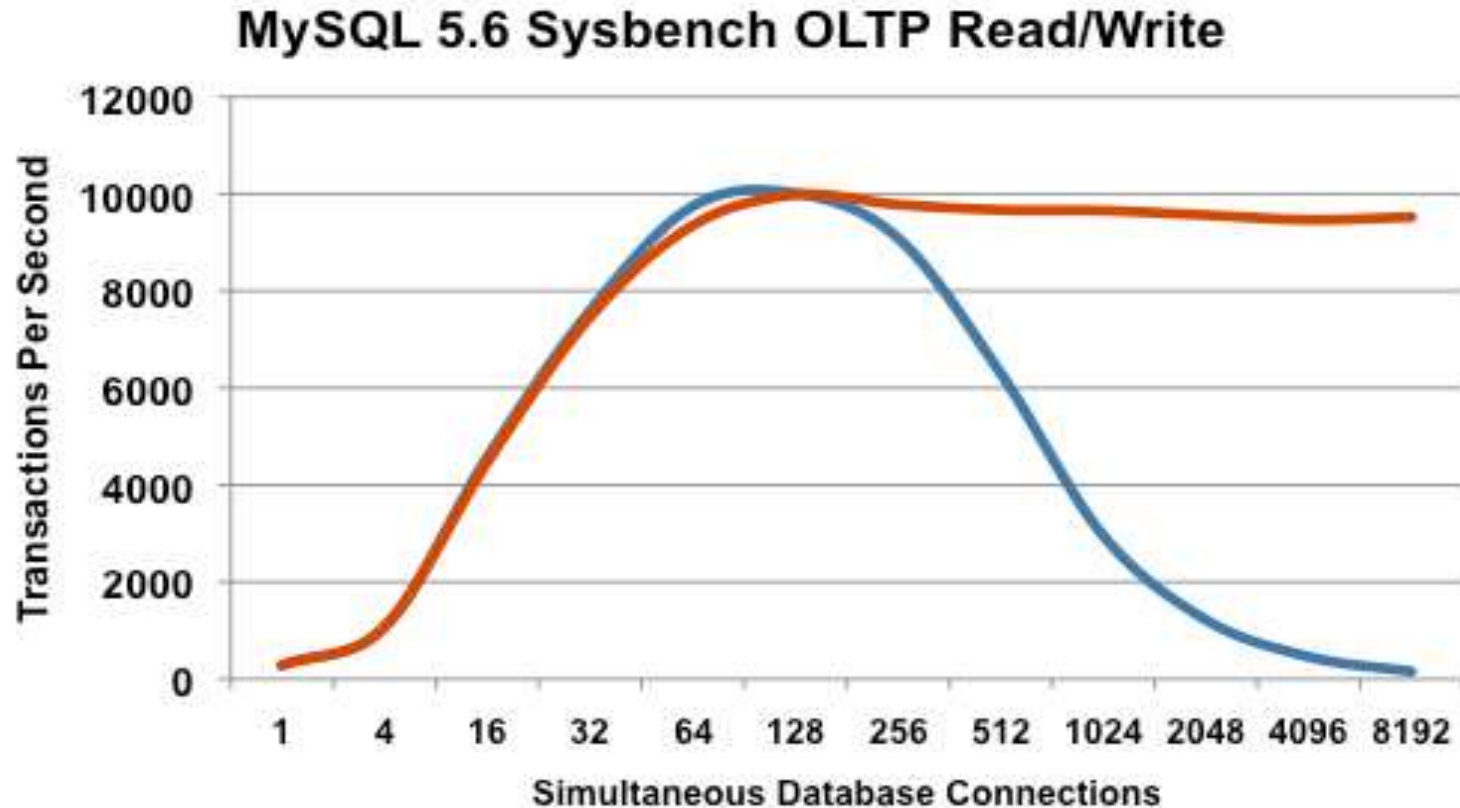
mysqldumpより49倍速い

Restore: 73 GB Database



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

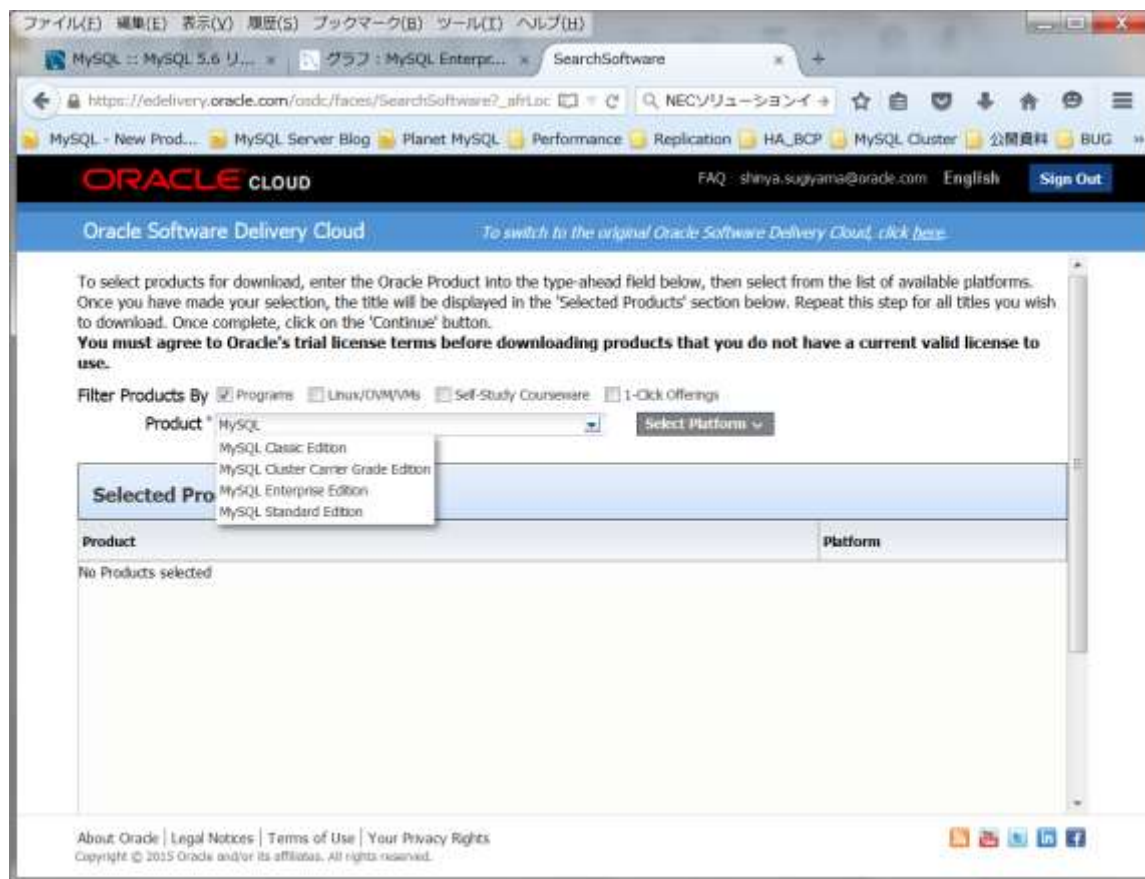
MySQL Integrates into your Oracle Environment



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>

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
- 5 参考情報

MySQL 8.0の参考資料

- MySQL 8.0 CTE & Window関数

- <https://www.mysql.com/jp/why-mysql/presentations/mysql-80-cte-window-function-201705-ja/>

- MySQL 8.0 Replication改善点

- <https://www.mysql.com/jp/why-mysql/presentations/mysql-80-replication-201705-ja/>

MySQLのイベント情報

- MySQLのイベント情報掲載ページ

– <https://www.mysql.com/jp/news-and-events/events/>

Oracle OpenWorld 2017の発表資料(英語資料)

- セッション情報確認 & ダウンロードURL

- <https://events.rainfocus.com/catalog/oracle/oow17/catalogoow17?search=mysql>

- 資料の一例

- CON7309: MySQL 8.0: What's New in the Optimizer

- CTEの使用例、ヒストグラム使用例、新しいコストモデルの使用例、など

- CON7292: Using MySQL Containers

- 公式MySQLコンテナの紹介、など

- CON3081: Using MySQL Flexible Schema (Document Store JSON) for IoT

- JSONデータをMySQLに格納して活用する方法、など

- HOL7299: MySQL Performance Tuning 101

- 基本的なチューニング方法、など

Integrated Cloud

Applications & Platform Services

ORACLE®