

ORACLE

MySQL JavaScript ストアドプログラムと、 MySQL Enterprise Edition、 デベロッパーライセンスの紹介

Open Developer Conference 2024
2024年9月7日

大塚 恒平/Kohei OTSUKA

日本オラクル株式会社

MySQL Principal Solution Engineer

MySQL Global Business Unit



大塚 恒平 (Kohei OTSUKA)

所属

- 日本オラクル株式会社MySQL GBU

役割

- MySQLのプリセールス、普及促進活動など

専門分野

- GIS、地図、地理などの業界で20年

Github

- kochizufan

出身地

- 姫路市

趣味

- オープンソース開発
- 地方史研究（群馬、奈良など）
- 石造文化財研究

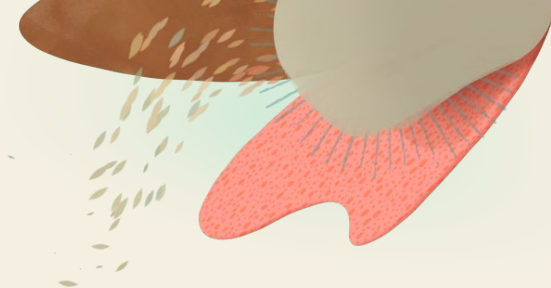


アジェンダ

1. MySQL とは
2. アプリケーション開発の効率化
3. MySQL Server 内での JavaScript サポート
4. JavaScript 対応した MySQL を試すには
5. HeatWave Always Free Tier を使ってみたレポート
6. 本セッションのまとめ

1. MySQL とは

—



MySQL とは

世界で最も普及しているオープンソースデータベース

デュアルライセンス

- コミュニティ版
- 商用版

Oracle が開発・提供

- 専任開発者
- バージョンリリース
- パッチ提供

MySQLを利用いただいているお客様

Social media	E-commerce	Finance	Tech	Car Manufacturers
				
				
				
				



MySQL リリース形態の変更

イノベーション・リリース (IR)

バグ修正と新機能追加を行うリリース
MySQL 8.1, 8.2, 8.3, 9.0, 9.1 ...

• リリース方針

バグ修正
セキュリティ・パッチ
新機能追加
機能やパラメータの非推奨化および削除

• リリースサイクル

3ヶ月毎
次バージョンのリリースでEOL

本番運用想定テスト済み

LTS (Long-Term Support) リリース

バグ修正のみを行うリリース
MySQL 8.4

• リリース方針

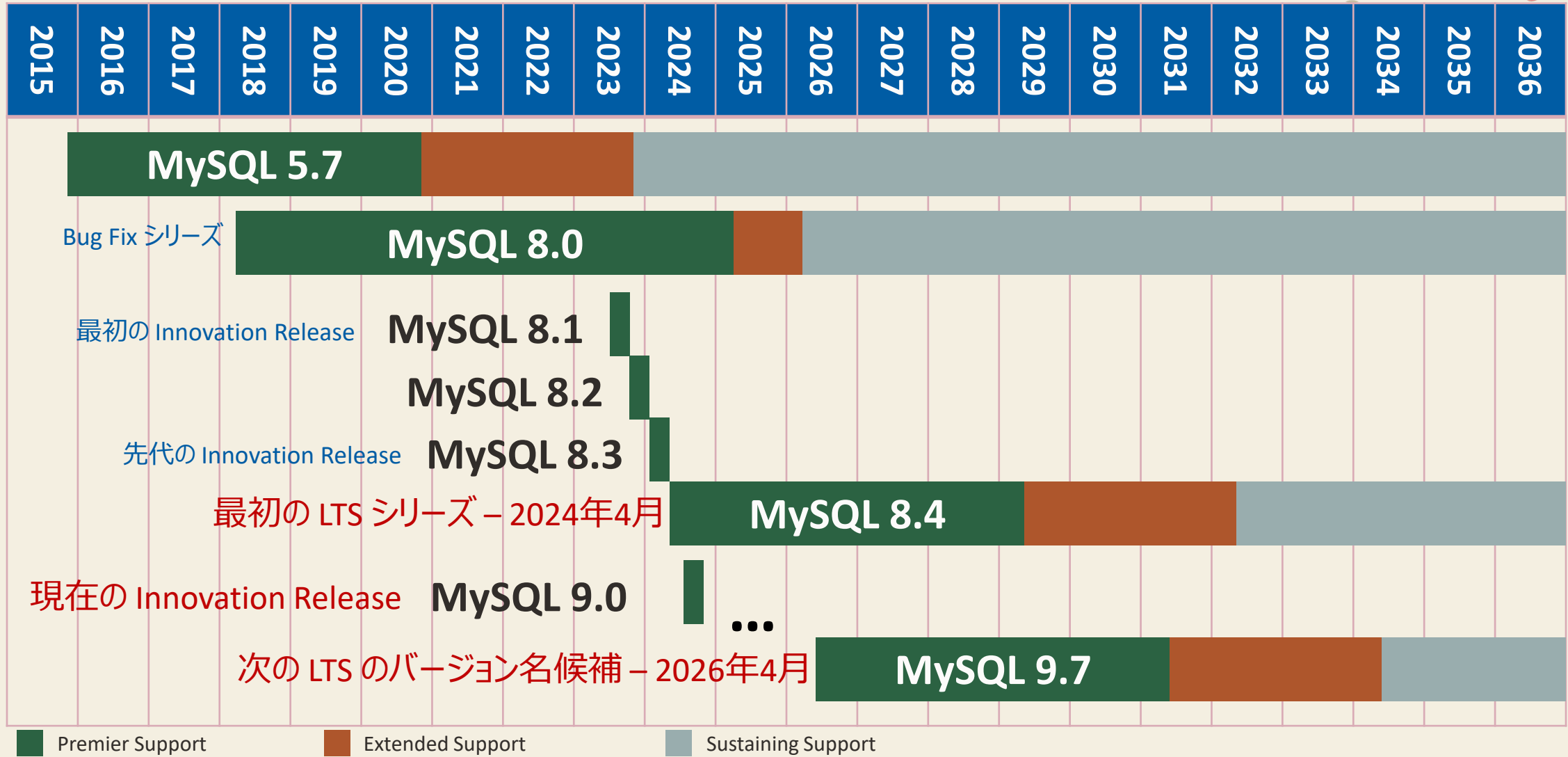
バグ修正
セキュリティ・パッチ
バージョン間の互換性重視

• リリースサイクル

リリース後8年間サポート
複数のLTSリリースをサポート予定

本番運用想定テスト済み

MySQL リリースとサポートのタイムライン



ベクトルデータ型サポート

データをベクトル化して保存し、ベクトル同士の近接性や類似性をベースに検索

- ベクトルカラムを持つテーブルを作成する標準SQLインタフェース

```
mysql> CREATE TABLE wikipedia (  
    id INT,  
    title VARCHAR(1024)  
    page_data TEXT  
    page_list TEXT,  
    page_url TEXT,  
    page_embedding VECTOR(1024) ENGINE_ATTRIBUTE='{"model":"cohere"}')  
ENGINE=lakehouse, SECONDARY_ENGINE=rapid;
```

- 距離関数
 - L1/MANHATAN
 - L2/EUCLIDIAN
 - L1^2/MANHATAN_SQUARED
 - L2^2/EUCLIDIAN_SQUARED
 - COSINE
 - DOT
 - HAMMING

<https://dev.mysql.com/doc/refman/9.0/en/mysql-nutshell.html>

mysql_native_password 削除

SHA-1アルゴリズムからの脱却

MySQL8.0

- caching_sha2_password がデフォルトの認証プラグインに変更(SHA-256)

MySQL8.4 LTS

- mysql_native_passwordのプラグイン化
- mysql_native_password は非推奨
- デフォルトではプラグインとしてロードされないように変更

MySQL9.0 IR

- mysql_native_password を削除

<https://blogs.oracle.com/mysql/post/mysql-90-its-time-to-abandon-the-weak-authentication-method>

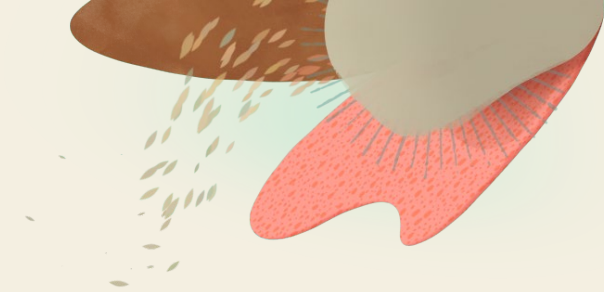
2. アプリケーション開発の効率化

—

アプリケーション開発の効率化

- db tech showcase 2024での発表: [MySQL アプリケーション開発を爆速にする最新手法](#)
- 年々複雑化するソフトウェア開発
- **高品質な製品をより速く、より効率的に**提供することに、開発者は力を尽くす
- ソフトウェアの作成、設計、テストの効率化に必要なのは：
 - 簡単化
 - 自動化
 - 最適化
- [MySQL を利用したアプリケーション開発効率化の新しいテクニック](#)を紹介

MySQL を用いた開発の効率化



1. MySQL と VS Code の統合

- MySQL 用 SQL IDE
- MySQL Shell for VS Code 拡張
- SQL、JavaScript、Python をサポート
- 管理API
- 接続管理
- OCI、MySQL HeatWave との統合

2. MySQL REST Service を用いた Web 開発

- MySQL Router の機能として提供された、MySQL の REST サービス
- スキーマからの作成の自動化
- VS Code 内での Rest API 開発の統合

3. MySQL Server 内での JavaScript サポート

- MySQL のストアードプログラム (プロシージャ、ファンクション) を JavaScript で記述
- VS Code 内での開発、デバッグ、デプロイ

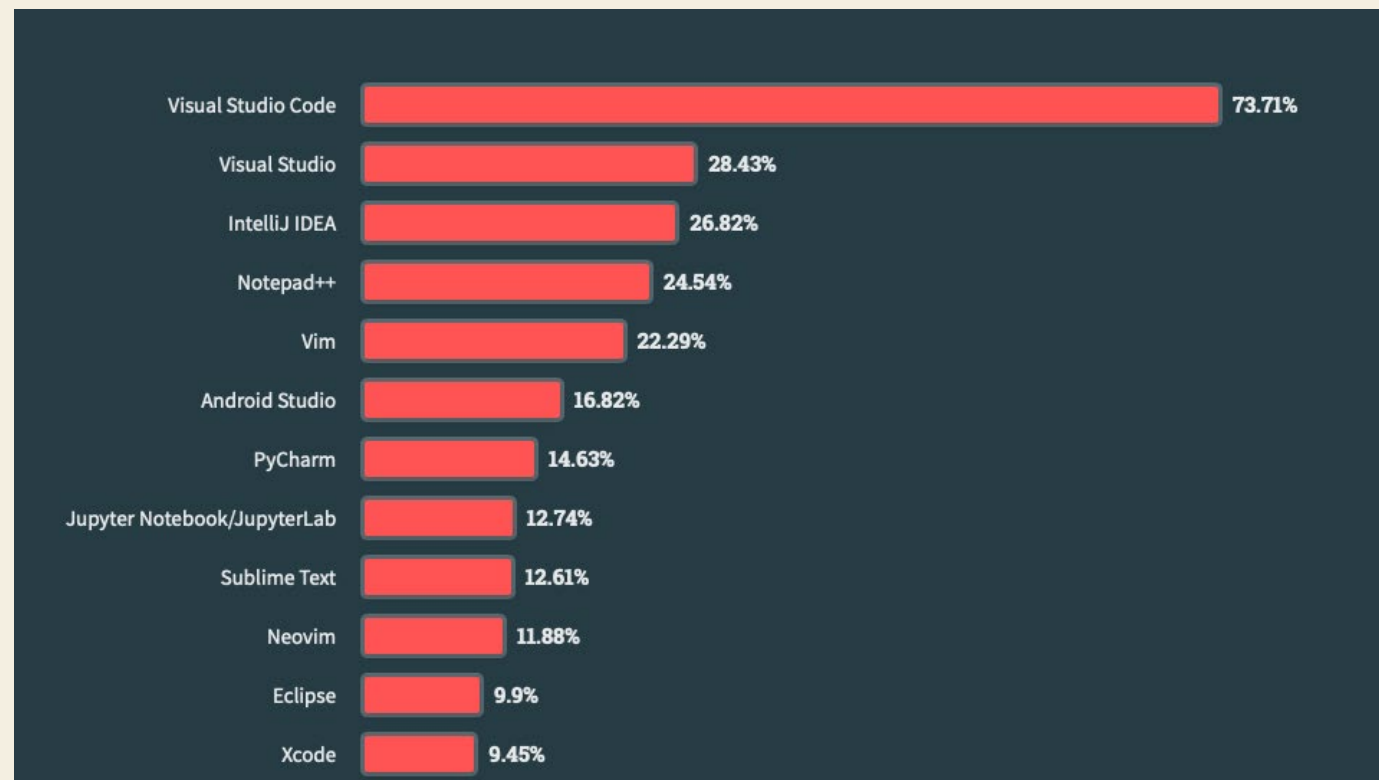
4. MySQL Operator for Kubernetes

- 自動化の実現
 - デプロイ
 - 構成
 - 可用性
- 開発、テスト、運用の一貫した環境
- 開発者はコードに集中できる



1. MySQL と VS Code の統合

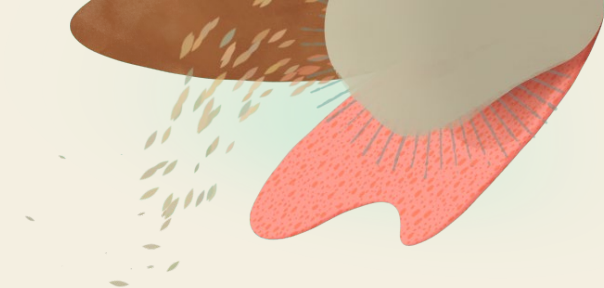
- VS Code IDE
 - 拡張
 - マルチ開発言語対応
 - マルチプラットフォーム対応
 - ソースコード管理の統合
- MySQL Shell 拡張を使うと：
 - スキーマ管理
 - SQL エディタ
 - SQL ワークシート
 - データ / 検索結果のグリッド表示
 - OCI 統合



Stack Overflow 2023 Developer Survey



2. MySQL REST Service を用いた Web 開発



MySQL REST Service

Restful Web サービス

- テーブル、ビュー、プロシージャの自動REST化
- {JSON} による応答
- 結果のページング
- 開発者サポート (GUI、CLI、API)

MySQL Shell for VS Code

- MRS管理用のGUIフロントエンド
- RESTfulなWebサービス作成
- 対話的なドキュメント
- CLIとスクリプト実行のサポート

ユーザー管理機能の搭載

- 一般的なOAuth2認証のサポート
- ロール、グループ、階層管理の使用
- ユーザー管理GUI
- CLIとスクリプト実行のサポート

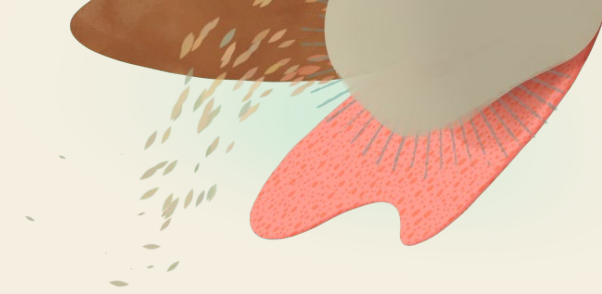


3. MySQL Server 内での JavaScript サポート

- JavaScriptのストアプログラムをGraalVMで実行
- SQL ベースのストアプログラム同様に動作し、加えて次のような特徴があります
 - 開発者エクスペリエンスの向上
 - セキュリティが核となる
 - 素早い開発
 - クラウドサービスとオンプレミスの両方に対応
- 以下の環境で使えます
 - Oracle Technology Network ([OTN](#)) で MySQL Enterprise Edition のプレビュー提供
 - OCI、AWS、Azure 向け Heatwave データベース・サービス (Always Free あり)



4. MySQL Operator for Kubernetes



組織の課題

たくさんの機材

たくさんのワークロード

それらを効率的に扱える
と

開発者はアプリケーションでの高い価値の実現に注力できる

複雑さの隠蔽化

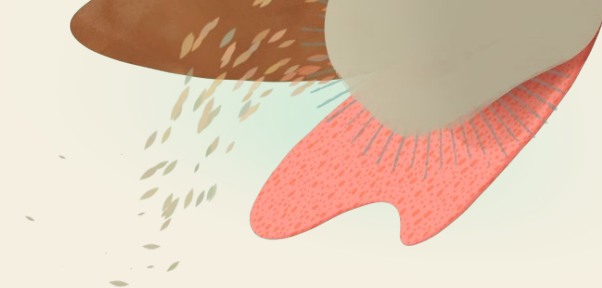
開発者はただ、あるホスト名 (あるいはIP) とポートで MySQL にアクセスできる簡単さを望む

Kubernetes ソリューション

コンテナ化されたワークロードとサービスを管理できる、ポータブルで拡張可能なプラットフォーム

宣言型の設定と自動化の両方を容易にする

MySQL を用いた開発の効率化



1. MySQL と VS Code の統合

- MySQL 用 SQL IDE
- MySQL Shell for VS Code 拡張
- SQL、JavaScript、Python をサポート
- 管理API
- 接続管理
- OCI、MySQL HeatWave との統合

2. MySQL REST Service を用いた Web 開発

- MySQL Router の機能として提供された、MySQL の REST サービス
- スキーマからの作成の自動化
- VS Code 内での Rest API 開発の統合

3. MySQL Server 内での JavaScript サポート

- MySQL のストアードプログラム (プロシージャ、ファンクション) を JavaScript で記述
- VS Code 内での開発、デバッグ、デプロイ

4. MySQL Operator for Kubernetes

- 自動化の実現
 - デプロイ
 - 構成
 - 可用性
- 開発、テスト、運用の一貫した環境
- 開発者はコードに集中できる



3. MySQL Server 内での JavaScript サポート

—

MySQL を用いた JavaScript アプリケーション

- JavaScript アプリケーションが主流になってきている
 - フロントエンドやサーバーサイドの軽量なアプリケーションに威力を発揮
- データ集約型のユースケースに対応
 - データの検証（バリデーション）
 - JSON 型や文字型のデータ処理 / フォーマット
 - データのクレンジングと変型
 - サーバーとクライアント間のデータ移動を最小限に
- MySQL Enterprise Edition 又は OCI クラウドの HeatWave サービスでのみ動作
 - Community Edition では動作しないのに注意
 - 試してみるための情報は、後半で紹介



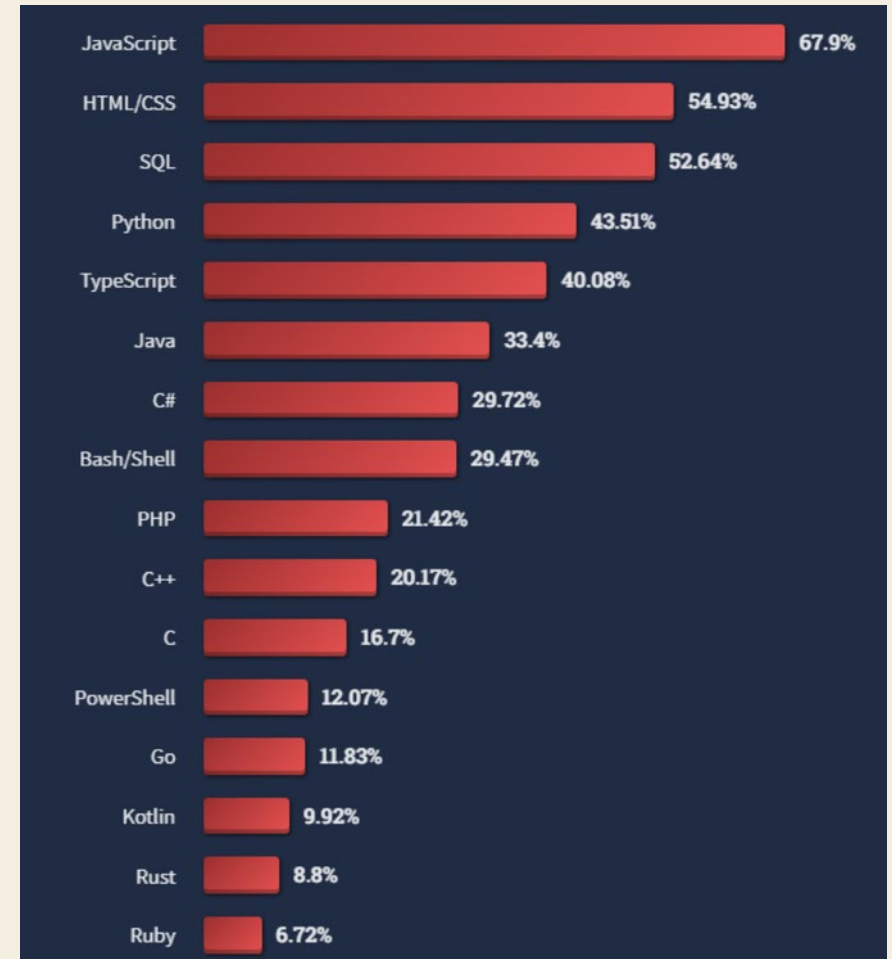
JavaScript

- 普遍性
 - 開発者にもっとも使われる開発言語の 1 つ*
 - 98% 以上の Web サイトが JavaScript を採用**
- 複数のランタイム
 - 全てのメジャーな Web ブラウザで採用
 - サーバーサイドでも多く使われるランタイム
 - Node.js
 - Deno
- 開発エコシステム
 - Npm レポジトリは、200 万件以上の JavaScript パッケージを持つ***
 - 1000 万人以上のユーザーが npm パッケージマネージャーを利用

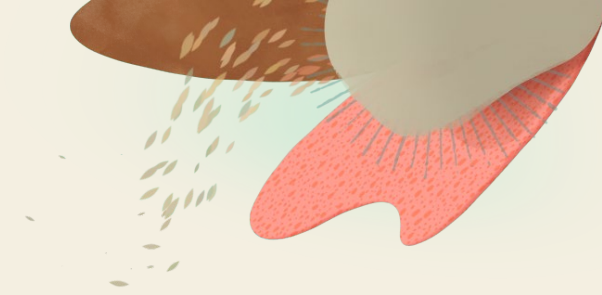
* Stack Overflow による 2024 年の調査

** <https://w3techs.com/technologies/details/cp-javascript>

*** <https://www.npmjs.com/>



MySQL ストアドプログラム – SQL と JavaScript の比較

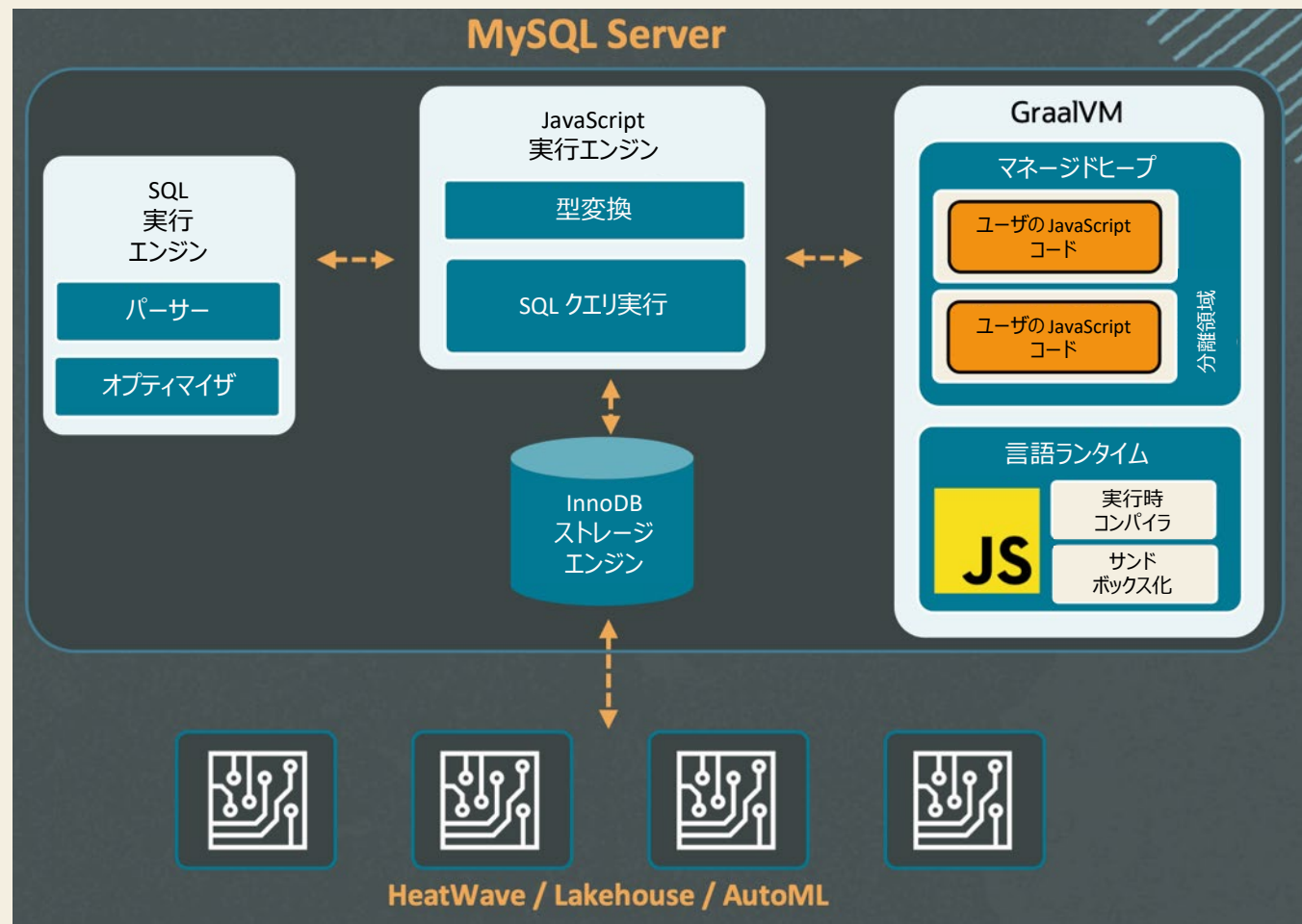


	SQL ストアドプロシージャ	JavaScript ストアドプロシージャ
表現力	✗ 使いにくく、コンテナデータ形式（たとえば配列やマップ）のような基本構成要素が欠けている	✓ 高い表現力と堅牢性
効率性	✗ インタプリタ経由の動作で最適化が難しい	✓ JS のコード解析ツールは多くある GraalVM により、JavaScript アプリは高速化、最適化
エコシステム	✗ 不十分：IDE、デバッガ、テストフレームワーク... などのサポートが不足	✓ JavaScript アプリケーション開発者向けの大規模な ツールのエコシステム
開発者の確保	✗ 経験豊富なプログラマーが少ない 特に MySQL エコシステムにおいて	✓ 1380 万人もの開発者 もっとも人気のある開発言語の 1 つ
再利用できる サードパーティライブラリ	✗ 少なく、ほとんどはコードサンプル	✓ 何千とある



MySQL Server 内での JavaScript

- MySQL の以下の機能と透過的に動作
 - InnoDB
 - 高可用性 / レプリケーション
 - HeatWave (OLAP向け)
 - HeatWave AutoML
 - HeatWave AutoPilot
 - HeatWave Lakehouse



JavaScript によるストアプログラムの定義

- 簡単な文法
 - LANGUAGE 節が JavaScript を受け入れるように
 - SQL 以外の言語の範囲を示すための文字列クォートの仕組み
 - AS \$\$... \$\$
 - AS \$JavaScript\$... \$JavaScript\$
- 関数環境
 - JavaScript で関数を再定義する必要はない
 - SQL で定義した引数識別子を JavaScript 内で直接利用可能
- 自動型変換
 - 透過的な SQL ⇔ JavaScript 間の型変換
 - INT、FLOATS、DATETIME、VARCHAR (utf8mb4) といった全ての変数型をサポート
- 外部ライブラリ (npmjs など) には未対応だが、対応予定あり

```
CREATE FUNCTION gcd_js (a INT, b INT)
RETURNS INT LANGUAGE JAVASCRIPT AS $$

    let [x, y] = [Math.abs(a), Math.abs(b)];

    while (y) [x, y] = [y, x % y];

    return x;

$$
```



SQL 内での JavaScript 対応

- SELECT 文
 - SQL ストアドプログラムが使える場所ならどこでも使用可能
 - LANGUAGE 節が JavaScript を受け入れるように
 - 式、射影、WHERE 節、GROUP BY、JOIN、ORDER BY、HAVING 等
- DML、DDL、VIEW 等
 - DML (INSERT, UPDATE, DELETE) 内で利用可能
 - CREATE TABLE AS SELECT を含む DDL も対応
 - VIEW 内でも使える
- 相互運用性
 - 既存の SQL ストアドプログラムの中で、JavaScript ストアドプログラムを実行できる
 - JavaScript と SQL のストアドファンクションを、互いに入出力を引数経由で入力とすることで、チェーン実行することが可能

```
SELECT col1, col2, gcd_js(col1,col2)
FROM my_table
WHERE gcd_js(col1, col2) > 1
ORDER BY gcd_js(col1, col2);
CREATE TABLE gcd_table
AS SELECT gcd_js(col1,col2) FROM
my_table;

CREATE TABLE gcd_table
AS SELECT gcd_js(col1,col2) FROM
my_table;
```



JavaScript 内での SQL 対応

- 対応できる SQL の種別
 - 一般的な SQL 文
 - バインド値のプレースホルダーを含むプリペアドステートメント
- データアクセス API
 - XDevAPI を用いて JavaScript 内で SQL を実行
 - クエリ実行結果に対する、透過的な SQL ⇔ JavaScript 間の型変換
- セッションの状況
 - JavaScript 内部でトランザクションが継続
 - JavaScript 内から、セッション変数や一時テーブルなど、あらゆるセッションの状態にアクセス可能

```
CREATE PROCEDURE gen_random_age (IN row_count INT) LANGUAGE
JAVASCRIPT AS $$
  let insertStatement = session.prepare( "INSERT INTO
my_table(age) VALUES ( ? )" );
  for (let j = 0; j < row_count; j++) {
    let random_age = Math.trunc(Math.random() * 100);
    insertStatement.bind(random_age).execute();
  }
  $$
```

```
CREATE PROCEDURE average_age (OUT avg_age FLOAT) LANGUAGE
JAVASCRIPT AS $$
  let age_sum = 0, count = 0;
  let selectStatement = session.sql(
    "SELECT age FROM my_table");
  let result = selectStatement.execute(), row = null;
  while(row = result.fetchOne()) {
    age_sum += row[0]; count++;
  }
  avg_age = age_sum / count;
  $$
```



容易なデバグ

- 標準出力ストリーム
 - 言語の標準出力および標準エラー出力のストリームに MySQL 内からアクセス
- エラー処理
 - 未処理の JavaScript の例外を、MySQL のエラーに変換
 - 未処理のランタイムエラーが発生した際に JavaScript のスタックトレースにアクセスできる
 - JavaScript 内での SQL 文実行時に、MySQL のエラーと警告を JavaScript の例外に変換

```
CREATE PROCEDURE division (IN a INT, IN b INT, OUT
result DOUBLE) LANGUAGE JAVASCRIPT AS $$
    function validate(num) {
        console.log("validating input value: ", num);
        if (num === 0) throw ("Division by Zero!");
    }
    validate(b);
    result = a / b;
$$
```

```
CALL division( 5, 0, @res);
ERROR 6000 (HY000): JavaScript> Division by Zero!

SELECT mle_session_state("stdout");
validating input value: 0

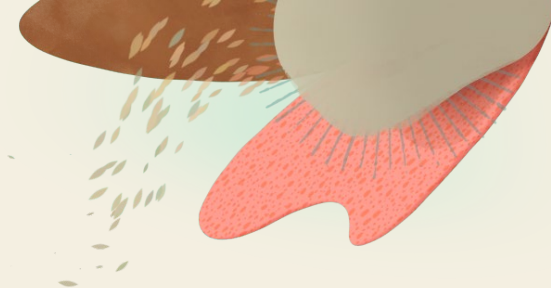
SELECT mle_session_state("stack_trace"); <js>
validate(division:9:187-214)
<js> division(division:11:222-232)
<js> :anonymous(division:15:256-265)
```



4. JavaScript 対応した MySQL を試すには

MySQL 内の JavaScript サポートは Enterprise Edition 以上

オンプレミスで使うには、MySQL デベロッパーライセンスを無料利用可能



- MySQL Enterprise Edition への完全なアクセス

- Enterprise Server
- Backup
- Router
- Shell
- Connectors
- JavaScript

MySQL デベロッパーライセンス

 MySQL Enterprise Masking De-identify, Anonymize Sensitive Data	 MySQL Enterprise TDE AES 256 encryption, Key Management	 MySQL Enterprise Authentication External Authentication Modules	 MySQL Enterprise Encryption Public/Private Key Cryptography, Asymmetric Encryption
 MySQL Enterprise Firewall Block SQL Injection Attacks, Intrusion Detection	 MySQL Enterprise Audit User Activity Auditing, Regulatory Compliance	 MySQL Enterprise Thread Pool Performance & Scalability for enterprise workloads	 MySQL Enterprise Backup Secure Backups, AES 256 encryption

ダウンロードは以下から

<https://www.oracle.com/mysql/technologies/mysql-enterprise-edition-downloads.html>

- 学習、開発、プロトタイプの使用にのみ利用可能（本番運用には利用不可）



MySQL Enterprise Editionが解決するお客様の課題

年間サブスクリプション 82.9万円 (1サーバーあたり)



ハードウェア削減によるコスト抑制

- クライアント接続スレッドの共有化による、リソースの競合抑制
- これにより少ないハードウェア構成で性能向上

→ **MySQL Enterprise Scalability**



運用負荷軽減によるコスト抑制

- MySQLサーバーの稼働状況やSQL性能の詳細な監視
- 監視スクリプトのメンテナンスや監視項目抽出の省力化
- 障害分析や課題解決への作業工数の削減

→ **MySQL Enterprise Monitor**
Oracle Premier Support for MySQL



機会損失削減とデータ保護

- システム障害による機会損失や信用低下の回避
- 大規模災害時のシステムへの影響の最小化
- 障害復旧までの時間的・人的なコストの削減

→ **MySQL Enterprise High Availability**
MySQL Enterprise Backup



包括的なセキュリティ対応

- セキュリティ対策に関する法規制やガイドラインへの準拠
- 情報漏えい予防による企業の信頼性やブランディングの保護

→ **MySQL Enterprise Security**



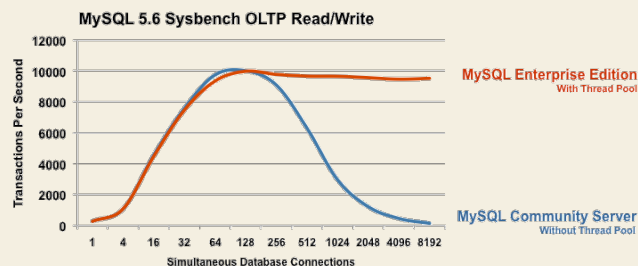
MySQL Enterprise Editionによるハードウェア/運用コスト削減

MySQL Enterprise Scalability & Oracle Premier Support for MySQL

MySQL Enterprise Scalability

- 高度にスケーラブルなスレッド処理モデルを採用
- クライアント接続、SQL実行スレッドの管理オーバーヘッドを軽減
- 読み取りのみで18倍、読み取り書き込みで60倍のスケーラビリティ向上

大きな並列接続要求の元でのスレッド処理を最適化
多くの接続があっても高いトランザクション性能を維持



Oracle Premier Support for MySQL

- 24時間x365日サポート
- 日本の営業時間中は日本語によるサポート
- 無制限サポート・インシデント
- MySQL GBUによる、開発チームと一体化したサポート体制
- バグ修正、パッチ、アップデートの提供

的確で迅速な対応により問題の調査検証工数を大幅に削減しTCOの低減に貢献

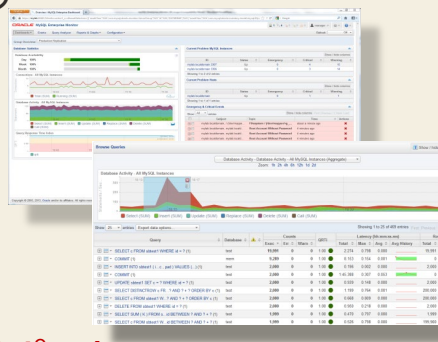


MySQL Enterprise Editionによる統合監視

MySQL Enterprise Monitorと後継ソリューション

MySQL Enterprise Monitor ※

- 複数のMySQLサーバを一括監視
- ルールに基づく監視と警告
- MySQL開発部隊とサポート部隊のノウハウを集約
- 問題が発生する前に通知

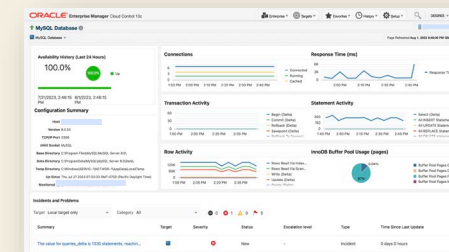


監視運用の属人化を排除し
DB管理工数の削減と安定稼働をサポート

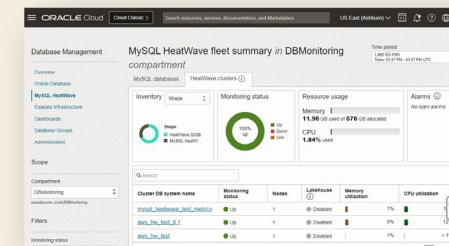
※ 2025年1月1日以降はMySQL Enterprise Monitorは
End of Life(EOL)に

後継ソリューション

- Oracle Enterprise Manager for MySQL
 - Enterprise EditionおよびCarrier Grade Editionのご契約をお持ちのお客様はそのまま利用可能



- OCI Database Managementサービス
 - **今後**の機能拡張にてオンプレミスやIaaS上のMySQLサーバを監視可能となる予定



MySQL Enterprise Editionによる機会損失削減とデータ保護

MySQL Enterprise High Availability & MySQL Enterprise Backup

MySQL Enterprise High Availability

- MySQLサーバーの高可用性パッケージ
“MySQL InnoDB Cluster”
- 障害検出やフェイルオーバーの自動化
- シンプルな環境構築 & 管理
- 複数のクラスタを連携させた災害対策構成
- 外部ツール、スクリプト不要
- 統合されたコンポーネント
 - データ複製: グループ・レプリケーション
 - ソフトウェア・ルーター: MySQL Router
 - 管理クライアント: MySQL Shell

使いやすさと高い可用性を両立し
ビジネスの継続性を担保

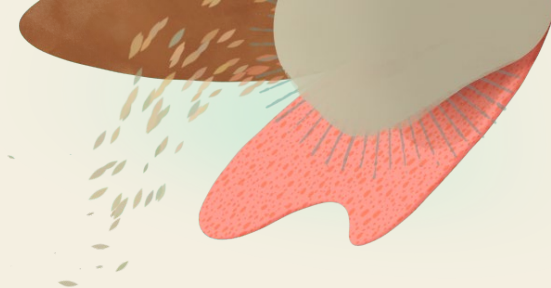
MySQL Enterprise Backup

- DB障害への高速かつ確実な対応
- コミュニティ版と比較して非常に高速な復旧
- ビジネスを止めないオンラインバックアップ
- 暗号化によるセキュリティ要件への対応
- クラウドストレージとの連携
- 多様なバックアップ・リカバリ方式
 - フル／増分／差分バックアップ
 - ポイントインタイム・リカバリ
 - Oracle Secure Backupとの連携

高速なバックアップ&復旧処理により
機会損失を最小限に抑える事が可能

MySQL Enterprise Editionによる包括的なセキュリティ対応

MySQL Enterprise Security各機能によるコンプライアンス要件対応



内部統制と法規制準拠のための監査

MySQL Enterprise Audit

- 柔軟な設定が可能なポリシー・ベース監査
 - 要件に応じDB内の全部または一部操作を記録
- 不正なアクセスの検出や不審な行為の抑制、
監査レポート提出に対応

DBへの攻撃に対する防御壁

MySQL Enterprise Firewall

- DB固有の攻撃に対するリアルタイム保護
- 正常アクセスの自動学習による設定の省力化

MySQL内で高速に動作し
外的脅威からデータを保護

データ漏洩対策としての暗号化

MySQL Enterprise Transparent Data Encryption

- ファイル不正アクセス時のデータ漏洩を阻止
 - 暗号化のためのアプリケーションの変更不要
- アプリケーションへの影響を抑えつつ
機密データの流出リスクを最小化

多角的なセキュリティ要素

MySQL Enterprise Authentication

- アクセス制御の一元化

MySQL Enterprise Masking

- 匿名化による機密データの不正使用からの保護

総合的なセキュリティ対策を提供

LINE株式会社: データベースの可用性、拡張性、およびセキュリティを改善 MySQL Community EditionをMySQL Enterprise Editionへアップグレード



製品およびサービス

MySQL Enterprise Edition

MySQL Enterprise Monitor

MySQL Query Analyzer

Oracle Premier Support for MySQL

アプリケーション

スマートフォンやPC端末から無料で1:1やグループでのチャット、音声通話及びビデオ通話を楽しめる人気コミュニケーションアプリ「LINE」及び関連サービスを提供

課題

- 新規サービス開始に向けた拡張性強化と作業効率化
- 10,000を超えるインスタンス管理・監視
- 可用性とセキュリティ向上

MySQL導入の効果

MySQL Enterprise Editionにアップグレードしたことで、数日を要していたデータベースバックアップが数時間で完了できるようになり、障害復旧に要する時間を1/10に短縮。新規サービスを追加する際の運用管理機能の開発を最小限にできることで迅速に開始できるようになり、将来にわたる事業成長をサポートする体制が整った

お客様の声

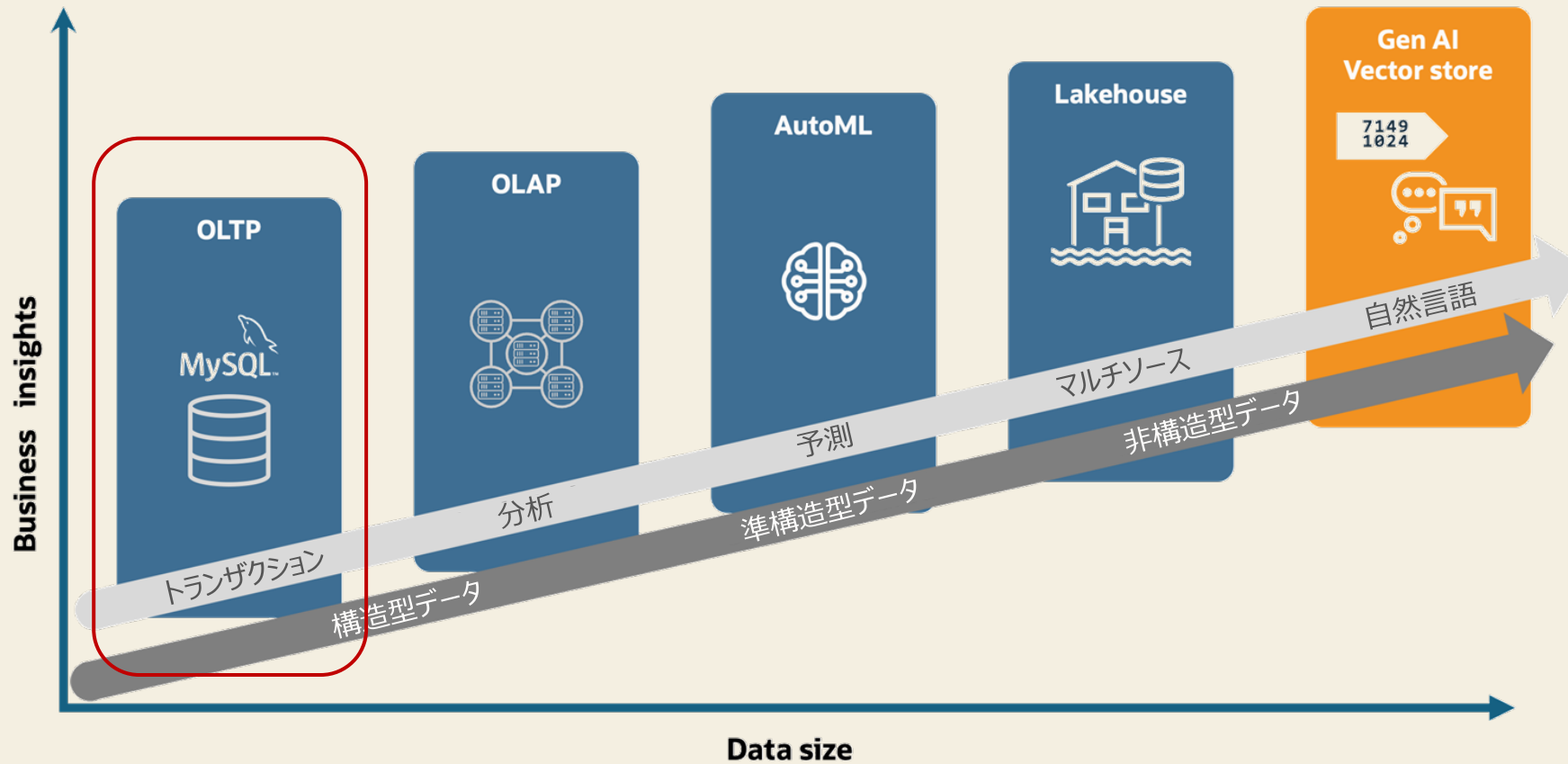
「他社製品も検討しましたが、日本語によるエンタープライズレベルのサポートを提供できるのはMySQL Enterprise Editionだけでした。加えて、長時間かかるメンテナンスや社内開発作業の削減に貢献する高度な監視、レプリケーション機能も重要な判断材料でした」

LINE株式会社 ITサービスセンター Database室
DB1チーム 北川 健太郎氏



HeatWave

OCI (オラクルのクラウド) 上に構築された、
MySQL Enterprise Editionベースの多用途マネージドデータベース



単独でトランザクション処理、データ分析、機械学習、データレイク、生成AIを実現



HeatWave データベースが OCI の Always Free 対象に

2024年8月から

Always Freeクラウド・サービス

期間の制限なく使用できるサービスです。

- Oracle APEX、Oracle SQL Developerなど、強力なツールを実装する2つのOracle Autonomous Database
- 2つのAMD Compute VM
- 最大4インスタンスのArm Ampere A1 Computeを1か月あたり3,000 OCPU時間と18,000GB時間で使用可能
- ブロック・ストレージ、オブジェクト・ストレージ、アーカイブ・ストレージ、ロード・バランサおよびデータ・エグレス、監視と通知

対象となるサービスのリストについては、以下を参照してください



30日間無料トライアル

300米ドルの無料クレジット。

- データベース、アナリティクス、コンピューティング、Container Engine for Kubernetesなど、幅広いOracle Cloudサービスへの30日間にわたるアクセス
- 利用可能なすべてのサービスで最大8つのインスタンス
- 最大5 TBのストレージ

対象となるサービスのリストについては、以下を参照してください

出典：<https://www.oracle.com/jp/cloud/free/>

HeatWave データベースが OCI の Always Free 対象に

自律型取引処理

3,338時間、2 TBのOracle Exadataストレージ。ミッションクリティカルなトランザクション処理を簡単に。インクラウドデータベースの未来。

NoSQL Database Cloud

53億の読み取り、53億の書き込み、450 GBのストレージ。読み取り、書き込み、またはストレージユニットを動的にプロビジョニングします。数分で迅速に稼働します。

Database Backup

5 TBのOracle Databaseバックアップ。Oracle Databaseをクラウドにバックアップするための、セキュアでスケラブルなオンデマンド・ストレージ・ソリューション。

Autonomous JSON Database

ドキュメントAPIとACIDトランザクションを備え、開発者向けに構築されたJSONデータベース。

MySQL HeatWave

5,000時間、550GBのストレージ。世界で最も普及しているオープンソースデータベース。MySQLチームが100%開発、管理、サポートを行っています。

Data Safe

無制限のクラウドターゲット。1つの支払済ターゲット。ターゲットごとに100万件の監査レコード。Oracle Data Safeは、Oracle Cloudおよびオンプレミスデータベースに不可欠なセキュリティ機能を提供して、リスクを軽減し、セキュリティを向上させます。

Autonomous Data Warehouse

3,338時間、2 TBのOracle Exadataストレージ。データウェアハウスを数秒でデプロイできます。フルマネージドで事前構成および最適化済み。

データベース

3,338時間、600GBのストレージ。あらゆる展開を対象とした、最も完全で統合されたセキュアなデータベース。

出典 : <https://www.oracle.com/jp/cloud/free/>



Always Free で HeatWave MySQL を使用する方法

- 以下URLからオラクルクラウドのトライアルアカウントを作成します
<https://signup.cloud.oracle.com/>
 - ホームリージョンは後から変更できないので注意
 - 日本には現在東京リージョンと大阪リージョンがあります
- 以下URLのチュートリアルを参考にし、HeatWave MySQL環境を構築する
 - OCIチュートリアル 入門編：その9 - クラウドでMySQL Databaseを使う
<https://oracle-japan.github.io/ocitutorials/beginners/creating-mds/>

※本日時点で、上記チュートリアルのスクリーンショットは、まだAlways Freeリリース後のものに更新されていませんが、基本的な手順は確認可能です

Always Free で HeatWave MySQL を使用する際の制限事項

- 最新バージョンのみ使用可能 (本日時点では、9.0.1)
- MySQL.Freeシェイプ、HeatWave.Freeのみ使用可能 (Always Free専用のスペックがあまり高くないシェイプ)
- ストレージサイズは50GB
- HeatWaveクラスタノードは1台のみ追加可能
- HeatWave AutoML と HeatWave Lakehouse は使用可能
- HeatWave GenAI は使用不可
- レプリケーション機能は使用可能
- 高可用性、リードレプリカは使用不可
- 自動バックアップは1日だけ取得される
- 手動バックアップやポイントインタイムリカバリは使用不可
- Database Management and Ops Insights サービスは使用不可 (データベースの監視ツール)

※原文 : <https://docs.oracle.com/en-us/iaas/mysql-database/doc/features-mysql-heatwave-service.html#MYAAS-GUID-772BD870-57C1-4B21-9205-FFC5B4290044>

5. HeatWave Always Free Tier を 使ってみたレポート

—
JavaScript ストアドプロシージャに、HeatWave クラスタノードを添えて

JavaScript ストアドプロシージャがどのようなことに使えるか？

趣味と実益を兼ねて考えてみる

- 発表者の趣味は、地域の石造文化財データベース化など

趣味

- オープンソース開発
- 地方史研究（群馬、奈良など）
- 石造文化財研究

- 石造物の判明製作時期にはあいまいさがある

- 正確に製作年月日がわかっているものもあれば、月までしかわからないもの、年までしかわからないもの、元号での「〇〇年間」までしかわからないものなどある
- あいまいな時期情報をそのまま記録しつつ、その範囲検索や順序並べ替えなどもできる工夫を実現したい

大門地蔵尊（鼓阪 / 手貝町
害門北側）

[修正提案をする](#)
種別: 地蔵



手貝町 鼓阪の大門地蔵

年代: 永禄
付帯情報: きたまちまつぶ お地蔵さんに「地蔵」あり 現地では名称なし 石造物調査報告書では雑司町 五輪塔残欠、五輪塔残欠、一尊像等多数あり
材質: 石造
刻銘: <@a> 尊□□
永禄□□
十一月廿八日
最終現地調査日: 2023-07-23
現況: 現存

大道寺寒念仏地蔵尊（館林 /
本町2丁目 / 大道寺）

[修正提案をする](#)
種別: 地蔵



大道寺寒念仏地蔵尊

年代: 寛延4年 (1751)
形状: 丸彫 立像
付帯情報: 西向き
材質: 石造
総高: 181cm
部分高: 98cm
刻銘: (表)
施主 信男信女
寒念佛供養
谷越町講中
(右)
寛延四未八月廿四日
最終現地調査日: 2022/12/12
現況: 現存
言及資料:



JavaScript ストアドプロシージャがどのようなことに使えるか？

趣味と実益を兼ねて考えてみる

- あいまいな時期情報を、期間として定義してみる
 - 時期情報自体は、テキスト情報として持つ
 - 例： 1996年2月14日 => “1916-02-14”、2020年10月 => “2020-10”、2024年 => “2024”、
平成年間 => “平成”
 - テキストで定義された時期情報から、2つのDATE型による期間情報（start、end）を生成する
 - 例：

“1996-02-14” =>	“1996-02-14” ~ “1996-02-14”
“2020-10” =>	“2020-10-01” ~ “2020-10-31”
“2024” =>	“2024-01-01” ~ “2024-12-31”
“平成” =>	“1989-01-08” ~ “2019-04-30”
- データの挿入/更新時に、トリガで時期情報から期間情報（start、end）を生成し、自動補完
 - ここで JavaScript ストアドプロシージャを試してみる

テストに使った環境とテーブル

- Oracle Cloud Infastructure の Always Free HeatWave インスタンス
- 以下のようなテーブルを作成し、JavaScript ストアドプロシージャとトリガを用いて、製作時期から期間情報を挿入できるようにする

cultural_prop (文化財テーブル)

id (PK) : INT	name (名前) : TEXT	time (製作時期) : TEXT	date_st (期間開始) : DATE	date_ed (期間終了) : DATE

mysql20240823185243 Always Free

編集 パフォーマンス・ハブ 起動 停止 More actions

DBシステム情報 接続 タグ

一般情報

OCID: ...mbv2n3afda 表示 コピー

説明: - 編集

Compartment: [REDACTED]

作成日: 2024年8月23日(金) 9:53:52 UTC

最終更新: 2024年8月30日(金) 10:43:50 UTC

アクティブ

関連付けられたサービス

データベース管理: 無効 ⓘ

Opsインサイト: 無効 有効化 ⓘ

高可用性

高可用性: 無効 ⓘ

HeatWave クラスタ

HeatWave クラスタ: 詳細 ⓘ

状態: ● アクティブ

HeatWave レイクハウス: 無効 ⓘ

削除プラン

削除保護: 有効 編集

自動バックアップの保持: 保持

最終バックアップ: 最終バックアップをス

DBシステム構成

シェイプ: MySQL.Free ⓘ

ECPU数: 1 ⓘ

メモリー: 8 GB ⓘ

自動ストレージ拡張: 無効

ストレージ・サイズ: 50 GB ⓘ

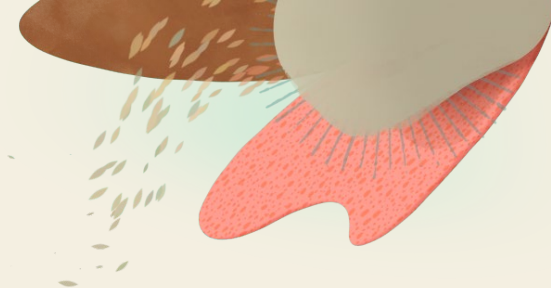
データベースのバージョン: 9.0.1 - 革新

構成: MySQL.Free.Standalone Always Free 編集

クラッシュ・リカバリ: 有効



時期情報から期間情報を生成する JavaScript ストアドプロシージャ

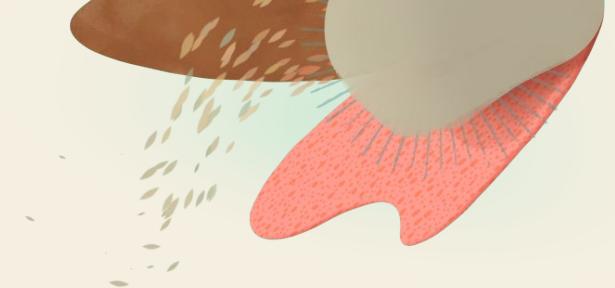


```
CREATE PROCEDURE fuzzyDate (IN date TEXT, OUT start DATE, OUT end DATE)
LANGUAGE JAVASCRIPT AS $$
  let sy, sm, sd, ey, em, ed;
  if (date == "平成") {
    sy = 1989; sm = 0; sd = 8;
    ey = 2019; em = 3; ed = 30;
  } else {
    date.match(/^(¥d{4})(?:-(¥d{2})(?:-(¥d{2}))?)?$/);
    if (!RegExp.$1) throw "Not date text";
    sy = parseInt(RegExp.$1);
    sm = RegExp.$2 ? parseInt(RegExp.$2) - 1 : 0;
    sd = RegExp.$3 ? parseInt(RegExp.$3) : 1;
    ey = parseInt(RegExp.$1);
    if (RegExp.$2) {
      if (RegExp.$3) {
        em = parseInt(RegExp.$2) - 1; ed = parseInt(RegExp.$3);
      } else {
        em = parseInt(RegExp.$2); ed = 0;
      }
    } else {
      em = 11; ed = 31;
    }
  }
  start = new Date(sy, sm, sd);
  end = new Date(ey, em, ed);
  $$;
```

元号指定は平成にのみ対応

西暦指定のみ対応
旧暦はとりあえず無視

行が挿入されると期間を生成するトリガ



```
> delimiter //  
  
> CREATE TRIGGER `ins_cult` BEFORE INSERT ON `cultural_prop` FOR EACH ROW BEGIN  
>   CALL fuzzyDate(NEW.time, NEW.date_st, NEW.date_ed);  
> END//  
  
> CREATE TRIGGER `upd_cult` BEFORE UPDATE ON `cultural_prop` FOR EACH ROW BEGIN  
>   CALL fuzzyDate(NEW.time, NEW.date_st, NEW.date_ed);  
> END//  
  
> delimiter ;
```

- あるカラムの値から別のカラムの値を生成するには、Generated Columnなどの方法も存在するが、ストアードプログラムはGenerated Columnでは使えないため、JavaScript ストアドを用いるにはトリガを使う必要がある
- 実際にデータを挿入すると以下のような結果になり、JavaScript ストアドの動作が確認できる

```
MySQL 10.0.30.141:3306 ssl cultural_prop SQL > INSERT INTO cultural_prop (name, time) VALUES ('青面金剛', '1875-02');  
Query OK, 1 row affected (0.1656 sec)  
MySQL 10.0.30.141:3306 ssl cultural_prop SQL > SELECT * FROM cultural_prop WHERE name = '青面金剛';  
+-----+-----+-----+-----+-----+  
| id      | name      | time      | date_st      | date_ed      |  
+-----+-----+-----+-----+-----+  
| 1100006 | 青面金剛 | 1875-02   | 1875-02-01   | 1875-02-28   |  
+-----+-----+-----+-----+-----+  
1 row in set (0.2767 sec)
```



期間の検索

Always Free でも使える HeatWave クラスタノードも試してみる

- 1300年から2024年までの、110万件のサンプルデータを作成し、検索してみる

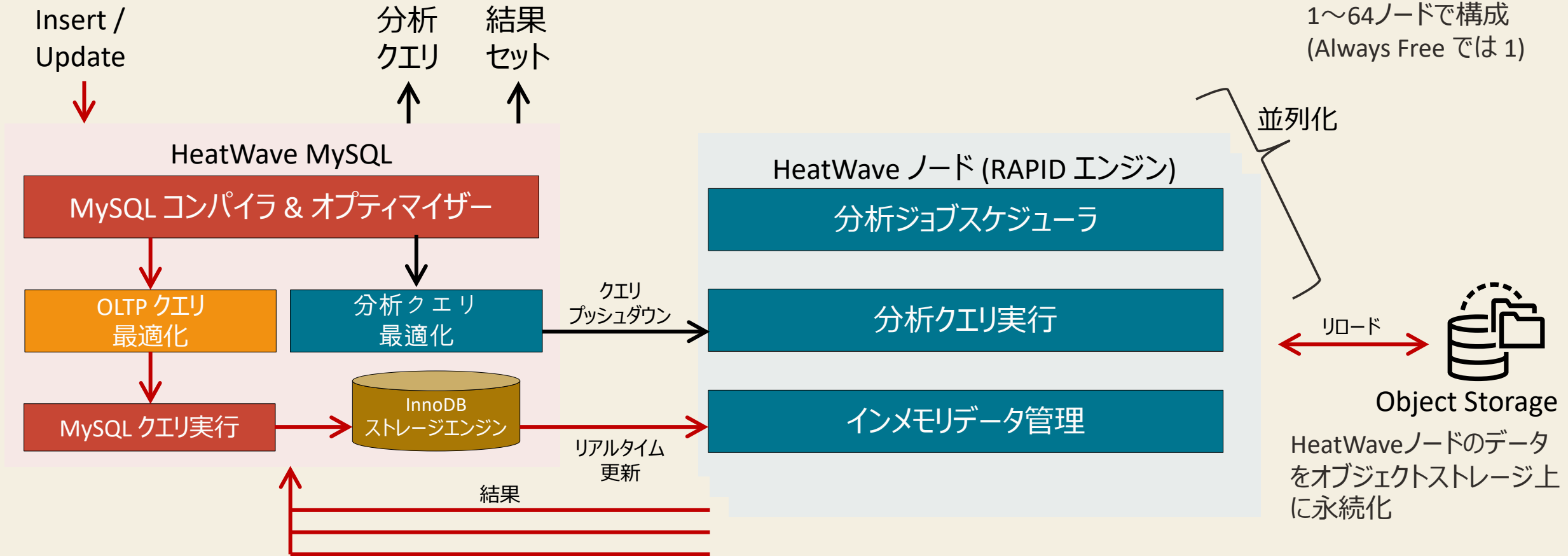
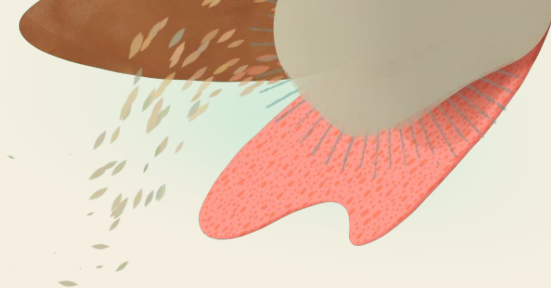
```
MySQL 10.0.30.141:3306 ssl cultural_prop SQL > SELECT COUNT(*), MIN(date_st), MAX(date_ed) FROM cultural_prop;
+-----+-----+-----+
| COUNT(*) | MIN(date_st) | MAX(date_ed) |
+-----+-----+-----+
| 1100006 | 1300-01-01   | 2024-02-29   |
+-----+-----+-----+
1 row in set (0.0837 sec)
```

- 検索結果比較の条件
 - インデックスなし
 - date_stとdate_edの複合インデックス
 - HeatWave クラスタノード (インメモリカラムナーDB) を用いた高速検索



補足: HeatWave クラスタノード (RAPID エンジン)

OLTP は InnoDB、OLAP は RAPID エンジンで動作する



HeatWaveノードは
1~64ノードで構成
(Always Free では 1)



補足: HeatWave クラスタノードの使い方

- HeatWave クラスタノードを起動
 - HeatWave のコンソール画面から、HeatWave クラスタを選択し、有効化する
- HeatWave クラスタノードの利用を宣言
 - ALTER TABLE [テーブル名] SECONDARY_ENGINE=RAPID; を実行
- HeatWave クラスタノードの利用を開始 (HeatWaveクラスタにデータを読み込み)
 - ALTER TABLE [テーブル名] SECONDARY_LOAD; を実行

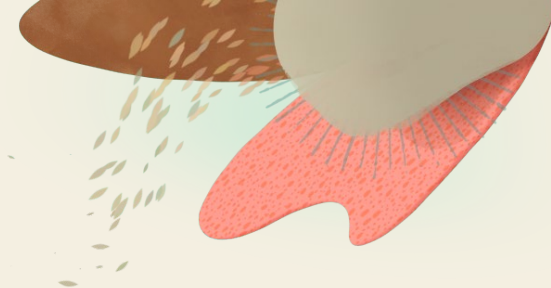
The screenshot displays the 'HeatWave クラスタ情報' (HeatWave Cluster Information) page. On the left, a navigation menu includes 'メトリック', 'エンドポイント', '読取りレプリカ', 'HeatWave クラスタ', 'バックアップ', 'チャネル', and '作業リクエスト'. The main content area shows cluster details: 'シェイプ: HeatWave.Free', '状態: アクティブ', 'ノード: 1', 'メモリー: 16 GB', and 'HeatWaveレイクハウス: 無効 有効化 ⓘ'. Below this, the creation and update timestamps are listed as '作成日: 2024年8月23日(金) 9:53:53 UTC' and '最終更新: 2024年8月23日(金) 9:53:53 UTC'. At the bottom, the 'HeatWave クラスタ・ノード' (HeatWave Cluster Nodes) table is shown with one node: 'cluster-node-1' in an 'アクティブ' state, created on '2024年8月23日(金) 10:08:24 UTC'. A red box highlights the cluster information and the node table.

```
MySQL 10.0.30.141:3306 ssl cultural_prop SQL > ALTER TABLE cultural_prop SECONDARY_ENGINE=RAPID;
Query OK, 0 rows affected (0.0106 sec)

Records: 0 Duplicates: 0 Warnings: 0
MySQL 10.0.30.141:3306 ssl cultural_prop SQL > ALTER TABLE cultural_prop SECONDARY_LOAD;
Query OK, 0 rows affected (0.8397 sec)
```



検索実行結果



```
SQL > SELECT COUNT(*) FROM cultural_prop WHERE date_st > "1854-03-24" AND date_ed < "1900-11-13";
+-----+
| COUNT(*) |
+-----+
|    71739 |
+-----+
1 row in set (0.1871 sec)
SQL > ALTER TABLE cultural_prop ADD INDEX idx_date (date_st, date_ed);
SQL > SELECT COUNT(*) FROM cultural_prop WHERE date_st > "1854-03-24" AND date_ed < "1900-11-13";
+-----+
| COUNT(*) |
+-----+
|    71739 |
+-----+
1 row in set (0.0454 sec)
SQL > ALTER TABLE cultural_prop ALTER INDEX idx_date INVISIBLE;
SQL > ALTER TABLE cultural_prop SECONDARY_ENGINE=RAPID;
SQL > ALTER TABLE cultural_prop SECONDARY_LOAD;
SQL > SELECT COUNT(*) FROM cultural_prop WHERE date_st > "1854-03-24" AND date_ed < "1900-11-13";
+-----+
| COUNT(*) |
+-----+
|    71739 |
+-----+
1 row in set (0.0094 sec)
```

0.2秒弱 (インデックスなし)

0.05秒弱 (複合インデックスあり)

0.01秒弱 (HeatWave クラスタノード)
--- インデックスなしの20倍、ありの5倍速

期間への複合
インデックス付与

HeatWave クラスタノード
の有効化とデータロード



6. 本セッションのまとめ

—

本セッションのまとめ

MySQL JavaScript ストアドプログラムと、HeatWave Always Freeの紹介

- MySQL Enterprise Editionでは、JavaScript によるストアドプログラムを利用可能
 - SQLによるストアドプログラムでは実装が困難な仕組みも簡単に実現可能
- JavaScript ストアドプログラムを試す方法:
 - オンプレミスでは、MySQL デベロッパーライセンスを使う
 - Enterprise Edition は高度なビジネス要件に対応できる機能やサポートを提供し、トータルコストを削減
 - 開発、学習などの用途に限って、デベロッパーライセンスで利用可能 => 実運用では使用不可
 - クラウドでは、OCI HeatWave の Always Free Tier を使う
 - 選べるイメージ、性能、インスタンス数などに制限はあるが、実運用でも利用可能
- Always Free Tier を試してみた
 - JavaScript ストアドプログラムを用いて、特殊なデータ型のような仕組みを作ることができた
 - 制限付きだが無償で使える、HeatWave クラスタノードもテストしてみた
 - 無償の範囲でも、インデックスなしでの検索の 20 倍、複合インデックス検索の 5 倍の検索性能
 - 無償でも複雑な検索条件の高速化に効果あり

Appendix.

MySQL 製品 / サービス / コミュニティのご紹介

柔軟なMySQLの利用方法

MySQLサーバーは全て共通のソースコードのためハイブリッド構成も可能

MySQLを自社で運用管理

オンプレミスでのMySQL

- バージョン選択や構成を最も柔軟に選択可能

IaaS上でのMySQL

- OCIのマーケットプレイスのイメージから簡単に環境構築

商用版MySQL

- コミュニティ版に加え、サポートやセキュリティに優れた商用版も

MySQLのマネージドサービス

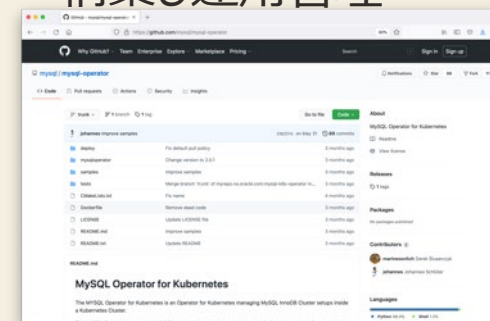
HeatWave Database Service

- MySQLチームが100%開発・提供するクラウド・サービス
- Amazon RDS (MySQL)の1/3以下のコスト
- データ損失なし、自動フェイルオーバーの高可用性機能をマネージドサービスで提供

クラウドネイティブなMySQL

MySQL Operator for k8s

- MySQLサーバーをKubernetes上に構築し運用管理



MySQL :: MySQL Operator for Kubernetes テクニカルアップデート

いずれの利用方法でもMySQL開発チームと連携した
専門部隊によるサポートサービスをご利用いただけます※



Oracle Premier Support for MySQL

- 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
- MySQL コンサルティング・サポート

開発チームと一体となったサポートサービス

⇒ 商用版MySQL サーバー及び HeatWave Database Serviceにより提供
年間サブスクリプション 74.9万円 (1サーバーあたり)



Get immediate help for any MySQL issue,
plus expert advice

MySQLの最新情報配信

MySQLホームページ

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

MySQL イベント

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

MySQLニュースレター 英語版&日本語版（月刊）

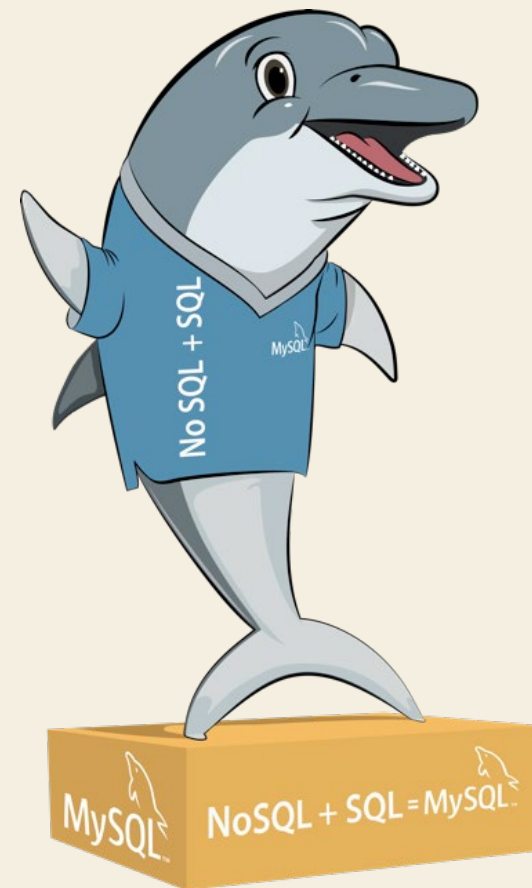
<https://www.mysql.com/jp/news-and-events/newsletter/>

The Oracle MySQL Japan ブログ

<https://blogs.oracle.com/mysql-jp/>

MySQL X（旧Twitter）日本語公式アカウント

[@mysql_jp](https://twitter.com/mysql_jp)



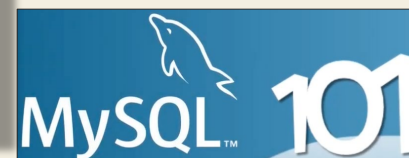
MySQL コミュニティ

- MySQL コミュニティの紹介
- MySQL への貢献
 - Oracle Contribution Agreement (OCA)
- MySQL 無償認証制度



MySQL コミュニティの紹介

- MySQL コミュニティへの貢献プロセスの運営
- MySQL ユーザーグループへの支援 <https://dev.mysql.com/community/mug/>
- 全世界でのサードパーティによるカンファレンスやイベントへの支援や参加
<https://dev.mysql.com/community/>
- 教育ビデオの作成
 - MySQL 短編動画 (MySQL Shorts)
 - MySQL 入門編シリーズ (MySQL 101 for Biginners)
 - <https://www.youtube.com/@mysql>
- MySQL RockStar プログラム
 - MySQLの利用促進に最も精力的に取り組んだ MySQL コミュニティ・メンバーへの表彰
 - 第1回: <https://blogs.oracle.com/mysql/post/mysql-rockstars-2022>
- MySQL ACE プログラム
 - MySQL プロジェクトでの ACE プログラムの運営
 - https://ace.oracle.com/pls/apex/ace_program/r/oracle-aces/home



MySQL への貢献

- MySQL オープンソースプロジェクトのコントリビューターコミュニティへの参加:
<https://forums.oracle.com/ords/apexds/post/contributing-code-to-mysql-8037>
- コントリビューターになるために持つべきこと
 - MySQL の機能を変更/修正したい、あるいは新しい機能を追加したいといった要望
 - MySQL ソースコードのダウンロード <http://dev.mysql.com/downloads/>
 - bugs.mysql.com のアカウント <http://bugs.mysql.com> or
 - 有効な GitHub アカウント <https://github.com>
- Oracle Contribution Agreement (OCA) への署名 <https://oca.opensource.oracle.com/>
 - OCAは、コントリビューターとオラクルの両方を法的攻撃から保護する短い法的契約です。OCAに署名することにより、コントリビューターはオラクルがコントリビューターのコードをオラクル・ソフトウェアで使用する事が法的に許可されていること、およびコントリビューターの知る限りにおいて、そのコードに特許的な問題がないことに同意することになります。

MySQL 無償認証制度

- MySQL コミュニティチームは、Oracle University および Oracle Academy と協力し、mylearn.oracle.com を介して、2ヶ月間の指定期間内に使用できる無料のトレーニングバウチャー/クレジットを受講者に提供します。
- ご興味のある方は、以下についての詳細をお知らせくだされば、MySQL コミュニティから連絡いたします。
 - 名前
 - 姓
 - Email アドレス
 - 居住国
- <https://education.oracle.com/>



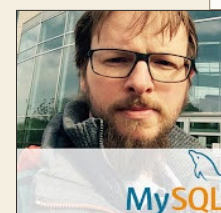
Become An
Oracle Explorer



Become
Oracle Certified

連絡先

- MySQL コミュニティとの CONTACT 先一覧:
- MySQL コミュニティページ, <https://dev.mysql.com/community/>
- MySQL Slack, <https://mysqlcommunity.slack.com>
- The Oracle MySQL ブログ, <https://blogs.oracle.com/mysql/>
- The Oracle MySQL Japan ブログ, <https://blogs.oracle.com/mysql-jp/>
- Planet MySQL, <https://planet.mysql.com/>
- LinkedIn, <https://www.linkedin.com/groups/60715/>
- ブログ, <https://lefred.be/>
- MySQL フォーラム, <http://lists.mysql.com/>
- ディスカッションフォーラム, <http://forums.mysql.com>



ORACLE