



RDS for MySQL/Aurora MySQL upgrade (major/minor version up)

アマゾン ウェブ サービス ジャパン株式会社
ソリューションアーキテクト

安井 通隆

2021/06/17



内容についての注意点

本資料では 2021年6月1日時点のサービス内容および価格についてご説明しています。最新の情報はAWS公式ウェブサイト (<http://aws.amazon.com/>) にてご確認ください。

- 資料作成には十分注意しておりますが、資料内の価格とAWS公式ウェブサイト記載の価格に相違があった場合、AWS公式ウェブサイトの価格を優先とさせていただきます
- 価格は税抜表記となっています。日本居住者のお客様がご利用される場合、別途消費税をご請求させていただきます

AWS does not offer binding price quotes. AWS pricing is publicly available and is subject to change in accordance with the AWS Customer Agreement available at <http://aws.amazon.com/agreement/>. Any pricing information included in this document is provided only as an estimate of usage charges for AWS services based on certain information that you have provided. Monthly charges will be based on your actual use of AWS services, and may vary from the estimates provided.

アジェンダ

- RDS MySQLのバージョンについて
- Aurora MySQLのバージョンについて
- アップグレード時の注意点(RDS for MySQL)
- アップグレード時の注意点(Aurora MySQL)
- まとめ

RDS MySQLのバージョン

RDS MySQLのバージョン

- MySQLのバージョン番号は x.y.z として編成されており、x.y をメジャーバージョン、z をマイナーバージョンを示しています
- RDS MySQLでは現在以下のメジャーバージョンが利用可能です
 - MySQL 8.0
 - MySQL 5.7
 - MySQL 5.6
- MySQL 5.6は廃止プロセスの途上です。最新状況はAWSフォーラムやAWS blogの告知をご確認ください
- MySQLの新しいマイナーバージョンは、AWSのデータベースエンジニアリングチームでのテストを経てRDSで利用可能となります。

詳細 : https://docs.aws.amazon.com/ja_jp/AmazonRDS/latest/UserGuide/CHAP_MySQL.html#MySQL.Concepts.VersionMgmt

MySQL 5.7での新機能・変更点

- ネイティブでのJSON対応
- 生成カラム(Generated columns)に対応
- パフォーマンススキーマの拡充とsysスキーマのサポート
- GTIDベースのレプリケーションに対応
- パラレルレプリケーションの改善
- 透過的圧縮の実装変更
- InnoDBでのネイティブ空間インデックスのサポート
- テンポラリテーブルがMyISAMからInnoDBに変更

詳細 : <https://aws.amazon.com/jp/about-aws/whats-new/2016/02/amazon-rds-now-supports-mysql-5-7/>

MySQL 8.0での新機能・変更点

- ウィンドウ関数と共通テーブル式のサポート
- JSONサポートのパフォーマンス向上
- パフォーマンス向上
- DDLオペレーションがアトミックでクラッシュセーフになり信頼性向上
- Graviton2インスタンスタイプが利用可能

詳細 : <https://aws.amazon.com/jp/about-aws/whats-new/2018/10/amazon-rds-now-supports-mysql-8/>

RDSのバージョン廃止ポリシーについて

“AWS では、定期的にメジャーまたはマイナーのエンジンバージョンの廃止を行います。メジャーバージョンについては、そのバージョンが延長サポートに移行された場合や、ソフトウェアの修正やセキュリティアップデートが行われなくなった場合に廃止します”

RDSでのバージョン廃止ガイドラインより

- MySQL 5.7 のExtendedサポートの終了予定 2023年10月
- MySQL 8.0のExtendedサポートの終了予定 2026年4月

“Amazon RDS でデータベースエンジンのメジャーバージョンが廃止される場合、廃止の発表から少なくとも 6 か月の期間が設定されるため、(中略) 廃止されるバージョンを実行しているインスタンスには、次のメジャーバージョンへの自動アップグレードが適用され、スケジュールされたメンテナンスウィンドウ中にアップグレードされます”

RDS DB のエンジンバージョンが廃止されるとどうなりますか? より

Aurora MySQLのバージョン

Aurora MySQLのバージョン

- Aurora MySQLでは以下の形式でバージョンを表します
<major version>.<minor version>.<patch version>
 - major version: 互換対象のMySQLとの対応関係あり
 - 1.y.z: MySQL 5.6との互換性
 - 2.y.z: MySQL 5.7との互換性
- APIやCLIなどからは以下のようなバージョン表記を採用しています

mysql-major-version.mysql_aurora.**aurora-mysql-version**

例: 5.7.mysql_aurora.2.03.2

- 以下のSQLでバージョン確認が可能

```
=>SELECT AURORA_VERSION();
```

Aurora MySQLの長期サポートリリース

- Aurora MySQLでは特定のバージョンを長期サポート(LTS)として指定しています
- 指定されたバージョンは利用可能になってから少なくとも1年間提供を継続いたします
- LTSバージョンの新しいパッチレベルでは新機能は含まれておらず、重要な問題の修正を含みます
- 重要なパッチ以外のアップグレードを適用したくないなどの場合にはこのバージョンを選択ください
- 2021年6月 現在のLTSバージョン
 - MySQL 5.7互換 2.07.*
 - MySQL 5.6互換 1.22.*

詳細 :

https://docs.aws.amazon.com/ja_jp/AmazonRDS/latest/AuroraUserGuide/AuroraMySQL.Updates.Versions.html#AuroraMySQL.Updates.LTS

アップグレード時の注意点 (RDS MySQL)

アップグレード

- メジャー/マイナーバージョンアップグレードが可能
- 注意事項/推奨事項
 - アプリケーションの互換性の確認などのために事前にスナップショットなどを用いて、事前に十分な検証をおこなってください
 - アップグレードしたインスタンスは元のバージョンに戻す事はできません
 - 切り戻しを行う場合、アップグレード時に取得されるアップグレード前のバージョンでのスナップショットから、新たなインスタンスを復元ください

詳細 : https://docs.aws.amazon.com/ja_jp/AmazonRDS/latest/UserGuide/USER_UpgradeDBInstance.MySQL.html

メジャーバージョンアップグレード

- メジャーバージョンアップは一つ新しいメジャーバージョンのみがサポートされます
 - MySQL 5.6 -> MySQL 5.7
 - MySQL 5.7 -> MySQL 8.0
- 注意事項/推奨事項
 - DBエンジンの動作の変更点** : [MySQL 5.7での変更点](https://dev.mysql.com/doc/refman/5.7/en/upgrading-from-previous-series.html)、[MySQL 8.0での変更点](https://dev.mysql.com/doc/refman/8.0/en/upgrading-from-previous-series.html) で事前に影響を受ける可能性がある変更点がないかを確認ください
 - アップグレード後のパラメータグループの準備** : アップグレード先のパラメータグループを作成
 - slow_log と general_log をテーブルに出力されている場合** : アップグレードで中身が空になりますので、事前に退避ください
 - MySQL 5.6 からMySQL 5.7 へのアップグレードは時間がかかる場合があります(後述)
詳細 : <https://dev.mysql.com/doc/refman/5.7/en/upgrading-from-previous-series.html>,
<https://dev.mysql.com/doc/refman/8.0/en/upgrading-from-previous-series.html>

マイナーバージョンアップグレード

- マイナーバージョンアップグレード概要
 - 手動 or 自動で実施可能
 - ※ いずれの場合もダウンタイムが発生するためプロダクション環境では手動での実施を推奨
- 自動マイナーアップグレード(AMVU)オプション
 - 本オプションが有効の場合、新しいバージョンがテストおよび承認されると自動でアップグレードを実施
- 注意事項/推奨事項
 - **リードレプリカを利用している場合** : MySQL DB インスタンスでリードレプリケーションを使用している場合は、ソースインスタンスのマイナーバージョンのアップグレード前に、すべてのリードレプリカをアップグレードする必要があります。

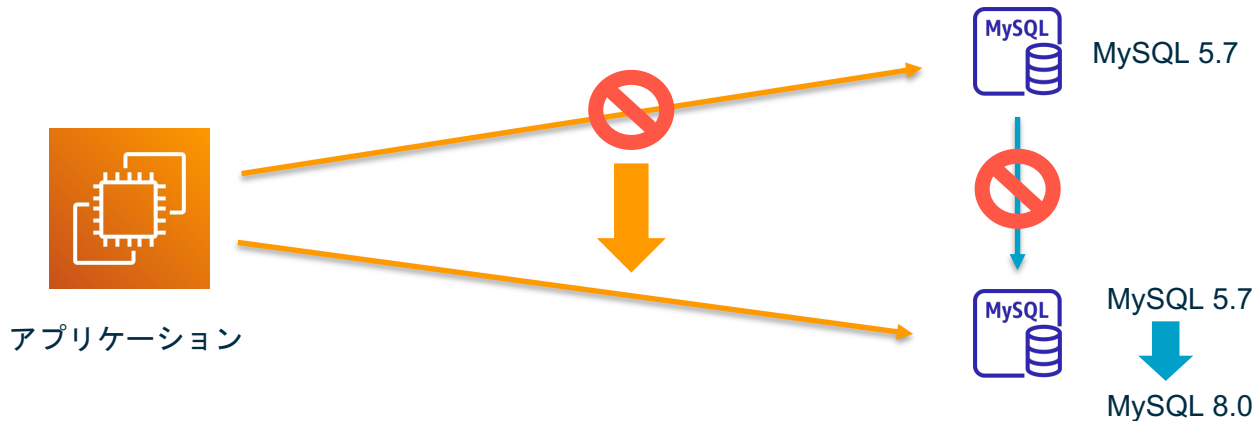
MySQL 5.7へのアップグレード

- MySQL 5.6.4から `datetime`、`time`、`timestamp` が秒未満の精度に対応しました。これに伴い、内部形式が変更になりました
- 便宜上、秒までの精度を旧形式。秒未満の精度も保持できる形式を新形式と呼びます
- MySQL 5.7へのアップグレード時に実行される `mysql_upgrade` が旧形式のテーブルを新形式に変換します。変換にはテーブルコピーが必要で、テーブルのサイズに応じた時間がかかります
- RDS MySQLの場合、MySQL 5.5からアップグレードした MySQL 5.6を5.7にアップグレードする場合に該当する可能性があります
- アップグレード前に旧形式のテーブルが存在するかをご確認ください(確認のクエリは詳細のリンクをご参照ください)

詳細 : https://docs.aws.amazon.com/ja_jp/AmazonRDS/latest/UserGuide/USER_UpgradeDBInstance.MySQL.html#USER_UpgradeDBInstance.MySQL.DateTime57

リードレプリカを使ったダウンタイムの短縮

- アップグレード時のダウンタイムを短くするために、リードレプリカを作り、レプリカをアップグレードした後に昇格と切り替えを行う



詳細 : https://docs.aws.amazon.com/ja_jp/AmazonRDS/latest/UserGuide/USER_UpgradeDBInstance.MySQL.html#USER_UpgradeDBInstance.MySQL.ReducedDowntime

リードレプリカを使ったアップグレード手順の例

1. 既存のDBインスタンスのリードレプリカを作成する
2. リードレプリカをアップグレード
3. アップグレード完了後、レプリケーションが追いついていることを確認
4. アプリケーションからの書き込みを停止
5. リードレプリカの昇格、これによりレプリケーションが解消されます
6. アプリケーションからの書き込み先をリードレプリカから昇格したインスタンスに向けて再開

リードレプリカを使ったアップグレード時の注意点

- リードレプリカをアップグレードしている間、レプリケーションは一時的に停止します。そのためレプリケーションソース上のバイナリログの削除も停止します。バイナリログのストレージ使用量(BinLogDiskUsage)とストレージの空き容量(FreeStorageSpace)に注意してください
- リードレプリカはデフォルトではシングルAZで作成されます。現在使っているインスタンスがマルチAZでメンテナンス後に昇格したリードレプリカで置き換える事が前提の場合、作成時点でマルチAZ構成にされることをお勧めします
- アップグレード後にリードレプリカへのレプリケーションが追いついていることを ReplicaLag などから確認後、昇格の操作を実行ください
- アップグレード前にアップグレード対象以外のリードレプリカを使用している場合、昇格するリードレプリカをソースとして改めて作成ください

アップグレード時の注意点 (Aurora MySQL)

アップグレード

- メジャー/マイナーバージョンアップグレードが可能
- 注意事項/推奨事項
 - DBクラスター単位でのアップグレードとなるため、同じDBクラスターのwriter/readerインスタンスが更新されます
 - 事前にAuroraクローンかスナップショットで別クラスターを作り、テスト/リハーサルを実施し問題ないことを検証されることをお勧めします

メジャーバージョンアップグレード

- メジャーバージョンアップグレード概要
 - アップグレードパス：
 - Aurora MySQL 1.x -> Aurora MySQL 2.x
- 注意事項/推奨事項
 - アップグレード前のバージョン：アップグレード前のバージョンが1.22.3未満のバージョンの場合、まず1.22.3へのアップグレードを行います
 - データベースのエンジン属性： aurora から aurora-mysql へ変更となります
 - Aurora Serverless、Aurora Multi-master、インスタンスを起動していないDBクラスターはインプレースでのアップグレードをサポートしません
 - Backtrackを使用しているDBクラスターでもインプレースアップグレードは可能ですが、アップグレード前に巻き戻すことはできません

詳細： https://docs.aws.amazon.com/ja_jp/AmazonRDS/latest/AuroraUserGuide/AuroraMySQL.Updates.MajorVersionUpgrade.html

マイナーバージョンアップグレード

- マイナーバージョンアップグレード概要
 - 手動 or 自動で実施可能
 - ※ いずれの場合もダウンタイムが発生するためプロダクション環境では手動での実施を推奨
- 自動マイナーアップグレード(AMVU)オプション
 - 本オプションが有効の場合、新しいバージョンがテストおよび承認されると自動でアップグレードを実施
- 注意事項/推奨事項
 - ダウンタイムのないパッチ適用：ベストエフォートで適用されます(後述)

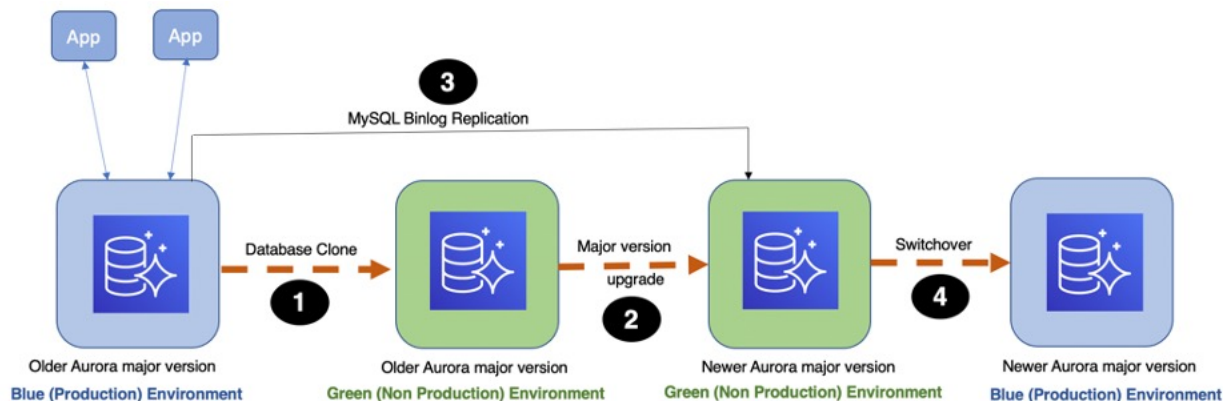
ダウンタイムのないパッチ適用

- Aurora MySQL 2.07.2～2.07.4と2.10.0以降のバージョンからのアップグレードで db.t2, db.t3のインスタンスのときに、ベストエフォートでダウンタイムのないパッチ適用(ZDP)が使用されます
- 2.10.0から、以降のバージョンへのアップグレードに関してはwriter/readerのいずれもがZDPの対象となります。それ以外のバージョンではwriterのみがZDPの対象で、readerは通常通り再起動いたします
- ZDPが実行されない条件
 - 長時間のトランザクションが実行中
 - SSLを使った接続が存在する
 - テンポラリテーブル、あるいはテーブルロックが実行されている
 - 未適用のパラメータ変更がある

詳細 : <https://docs.aws.amazon.com/AmazonRDS/latest/AuroraUserGuide/AuroraMySQL.Updates.Patching.html#AuroraMySQL.Updates.ZDP>

Blue/Green によるアップグレード

- RDS MySQLのレプリカを使ったメジャーバージョンアップグレードの方法の類似の手法として、Aurora MySQLのバイナリログレプリケーションを使って2つのクラスターでアップグレード時の切り替え時間を短縮する手法



詳細: <https://aws.amazon.com/jp/blogs/database/performing-major-version-upgrades-for-amazon-aurora-mysql-with-minimum-downtime/>

まとめ

- RDS MySQL/Aurora MySQLのバージョンの考え方を説明しました
- RDS MySQL/Aurora MySQLともにレプリケーションをうまく使うことで、アップグレード時のサービス中断を短くすることができます
- マネージドサービスであるため、バージョンのライフサイクルに合わせて余裕を持ったアップグレード計画を立案ください