

CUS-33

コロナ禍におけるフードデリバリーの 需要拡大への負荷対策

岡田 泰弘

株式会社出前館

プロダクト開発本部 開発管理部



自己紹介

岡田 泰弘（おかだ やすひろ）

- 所属

プロダクト開発本部

開発管理部 インフラグループ

- 出前館会員ランク

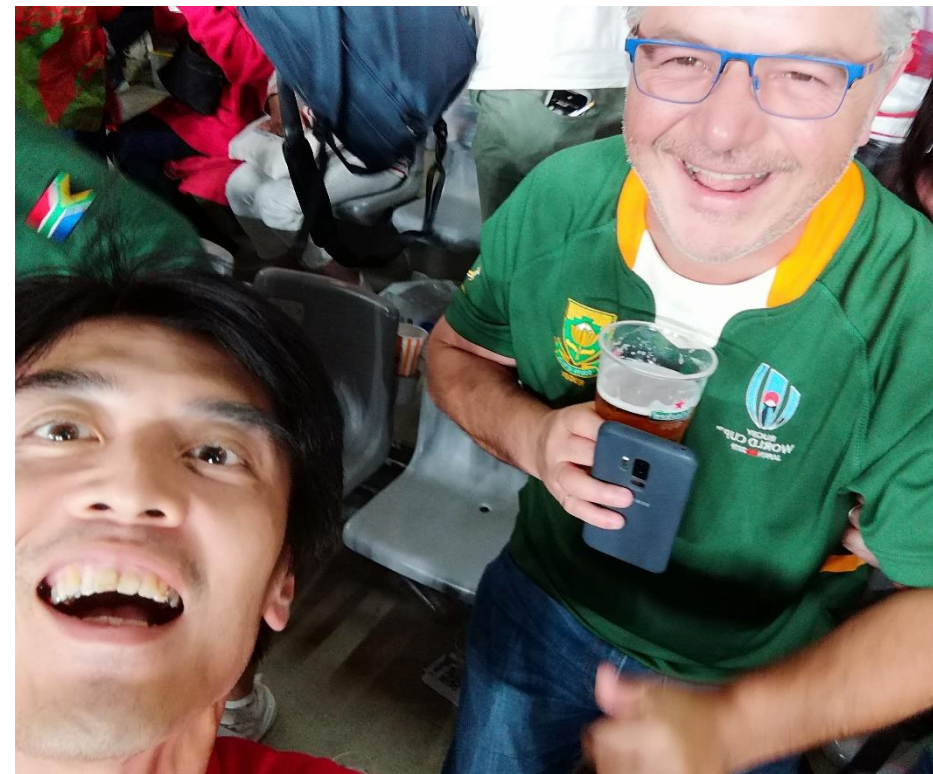
ゴッド （10回以上注文/3か月）

- 好きなアマゾン ウェブ サービス (AWS)のサービス

Amazon Relational Database Service (Amazon RDS)

AWS Database Migration Service (AWS DMS)

Amazon Route 53



目次

- 01 出前館について
- 02 ユーザー数の急増とキャパシティ限界
- 03 AWS本格利用によるスケーラビリティの改善
- 04 レスポンスタイム向上への取り組み
- 05 現在の取り組みと今後の計画

01 出前館について

02 ユーザー数の急増とキャパシティ限界

03 AWS本格利用によるスケーラビリティの改善

04 レスポンスタイム向上への取り組み

05 現在の取り組みと今後の計画

出前館について

2000年にサービスを開始した日本最大級のオンラインフードデリバリーサービスです。

2020年6月 LINEとの資本業務提携（3月発表）

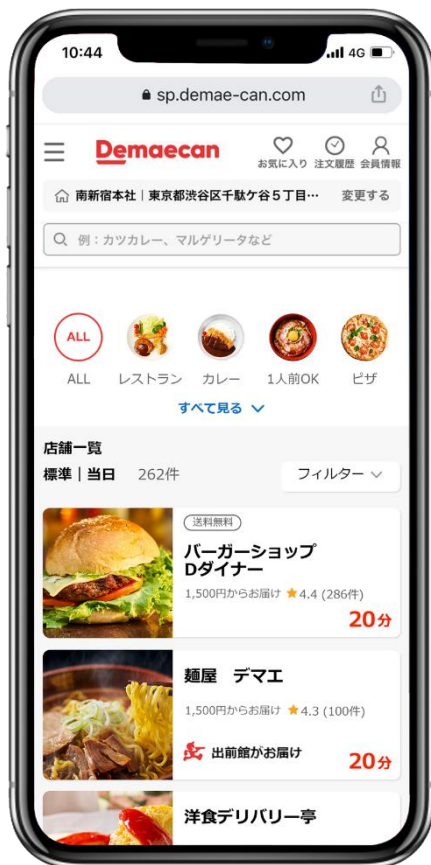
- ・サービスブランド（LINEデリマ）の統合
- ・出前館、LINEのアカウント連携開始
- ・新経営体制がスタート



出前館について

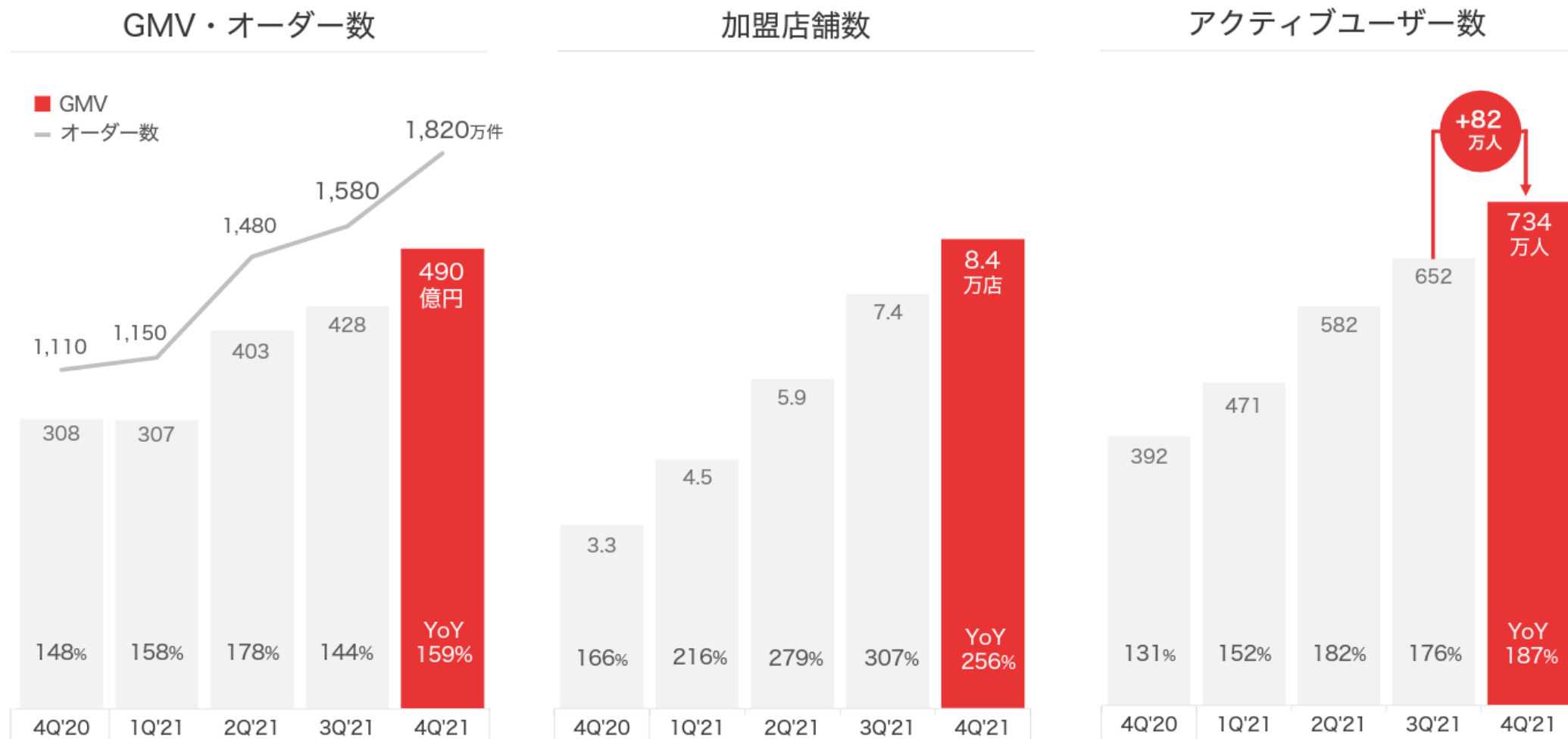
ユーザーからの注文を受け、配達代行のスタッフが店舗で商品を受取、ユーザーへお届けします。

飲食以外のデリバリーも始めました。



出前館のデリバリー需要について

近年、新型コロナウイルスにより、利用者数や加盟店舗数が急激に伸びています。



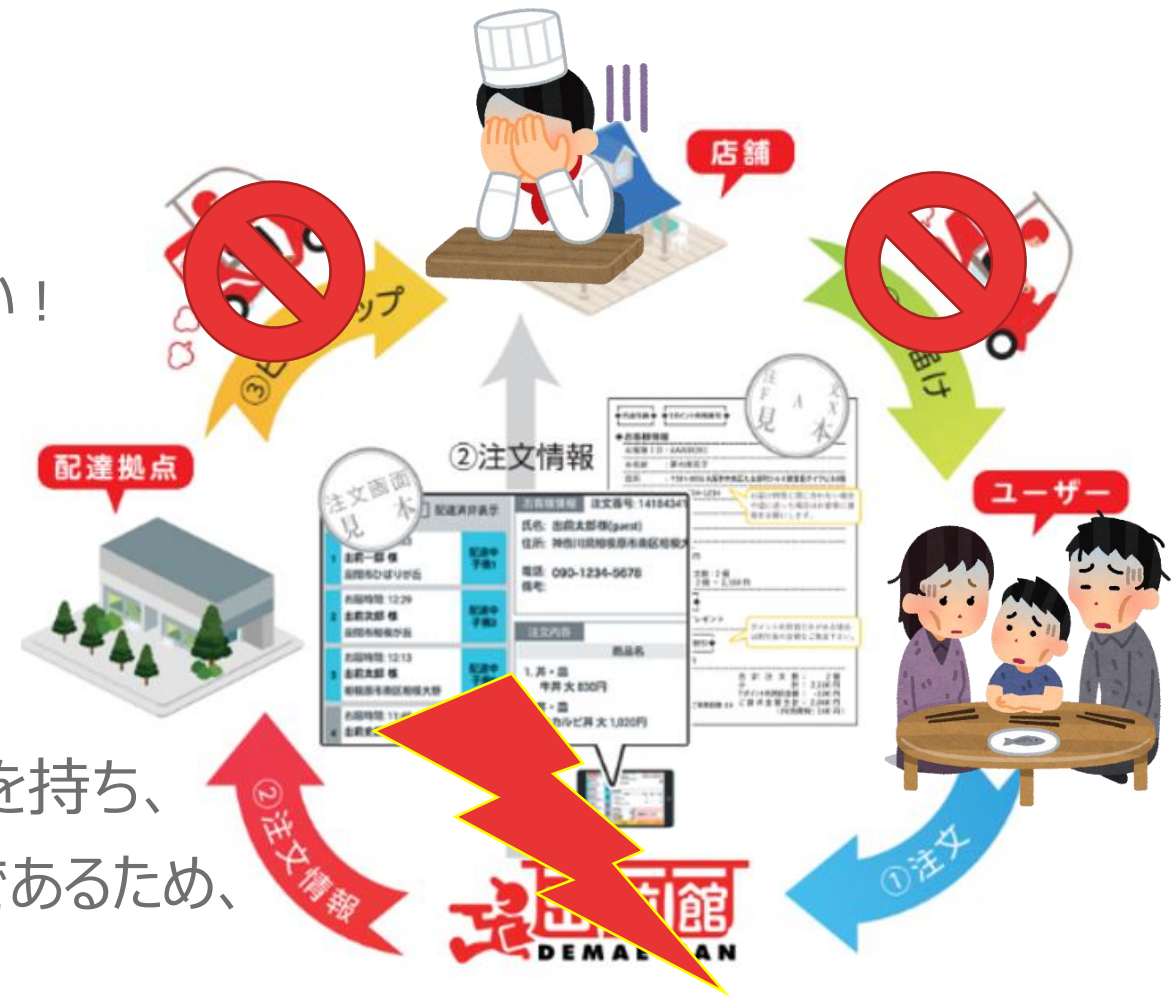
※8月期決算 4Q : 6~8月

出前館利用のピークタイム

食事のタイミング 昼食（12:00～14:00）、夕食（18:00～20:00）の直前に、システム負荷のピークが来ます。

ピークタイムの障害、配達遅延は絶対に避けたい！
空腹は怒りを生む・・・

出前館システムは、ミッションクリティカルな特性を持ち、
1日の中でもシステム負荷の差が大きいシステムであるため、
クラウド利用に適したサービスと言えます。



01 出前館について

02 ユーザー数の急増とキャパシティ限界

03 AWS本格利用によるスケーラビリティの改善

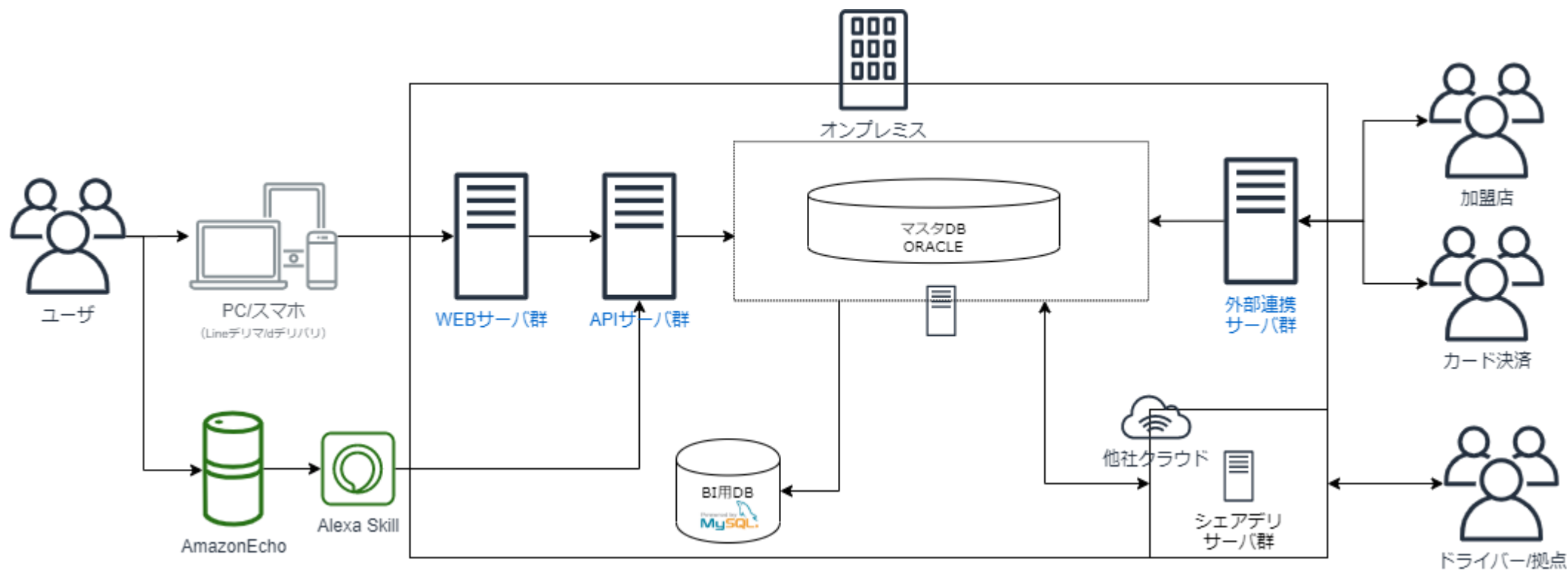
04 レスポンスタイム向上への取り組み

05 現在の取り組みと今後の計画

AWS移行前のシステム構成

従来システムはほぼオンプレミス上で構成されて以下の課題があった。

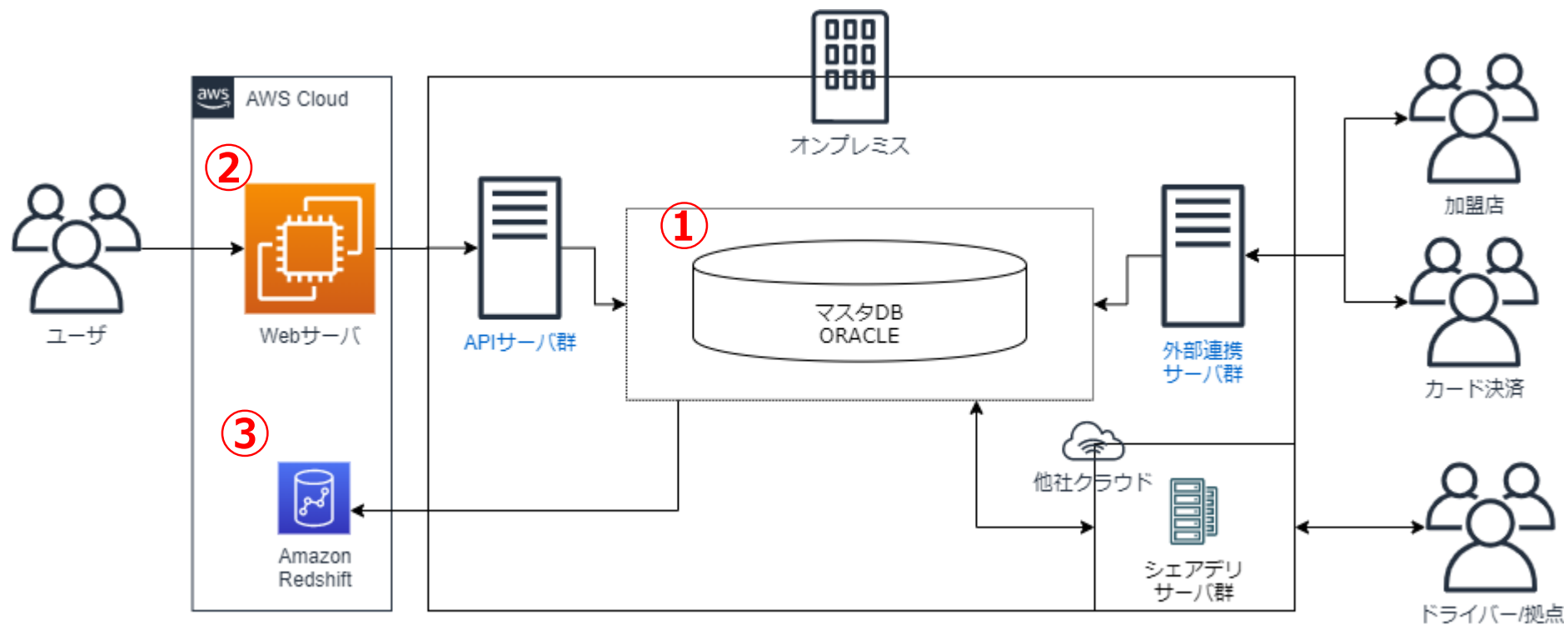
- スケールアウトできない
- CPUリソース不足でレスポンスが遅い
- 集計が遅くレスポンスも遅い



AWS利用初期段階のシステム構成

取り急ぎキャパシティの改善を図った。

- ① SQLチューニングと並行して Oracle のハードウェア換装を実施した。
- ② WebサーバをAWSに移設し、余剰となったサーバをAPIサーバに転用した。
- ③ BI環境をAmazon Redshiftに移行した。



フードデリバリー需要増加後のシステム稼働状況

DBサーバをスケールアップしたにもかかわらず、すでにCPU使用率は70%程度に達していた。
経営体制の変更や緊急事態宣言もあり、システムの課題が浮き彫りに…。

加盟店数：3倍
オーダ数：2倍
にしたい！！



LINE



ただし…。

【システム課題】

- ・DBサーバのCPUリソース不足
- ・DBサーバ以外もスケールアウトができない
- ・レスポンスが遅い。（店舗検索、商品検索）
- ・モノリシックな構成のため開発スピードが遅い…。
- ・CIが遅い…。
- ・全文検索が遅すぎて使われていない…。

リソースの拡張が容易なAWSへ移行開始！！

- 01 出前館について
- 02 ユーザー数の急増とキャパシティ限界
- 03 AWS本格利用によるスケーラビリティの改善**
- 04 レスポンスタイム向上への取り組み
- 05 現在の取り組みと今後の計画

データベースの負荷状況分析

課題

DBサーバのCPUリソース不足



分析①

特定のSQL、リクエスト元サーバなどの偏りによるものではない

分析②

DBサーバのハードウェア増強はできない

分析③

オーダー数の増加率よりも、CPU使用率、リクエスト数の増加率の方が大きい



負荷対策

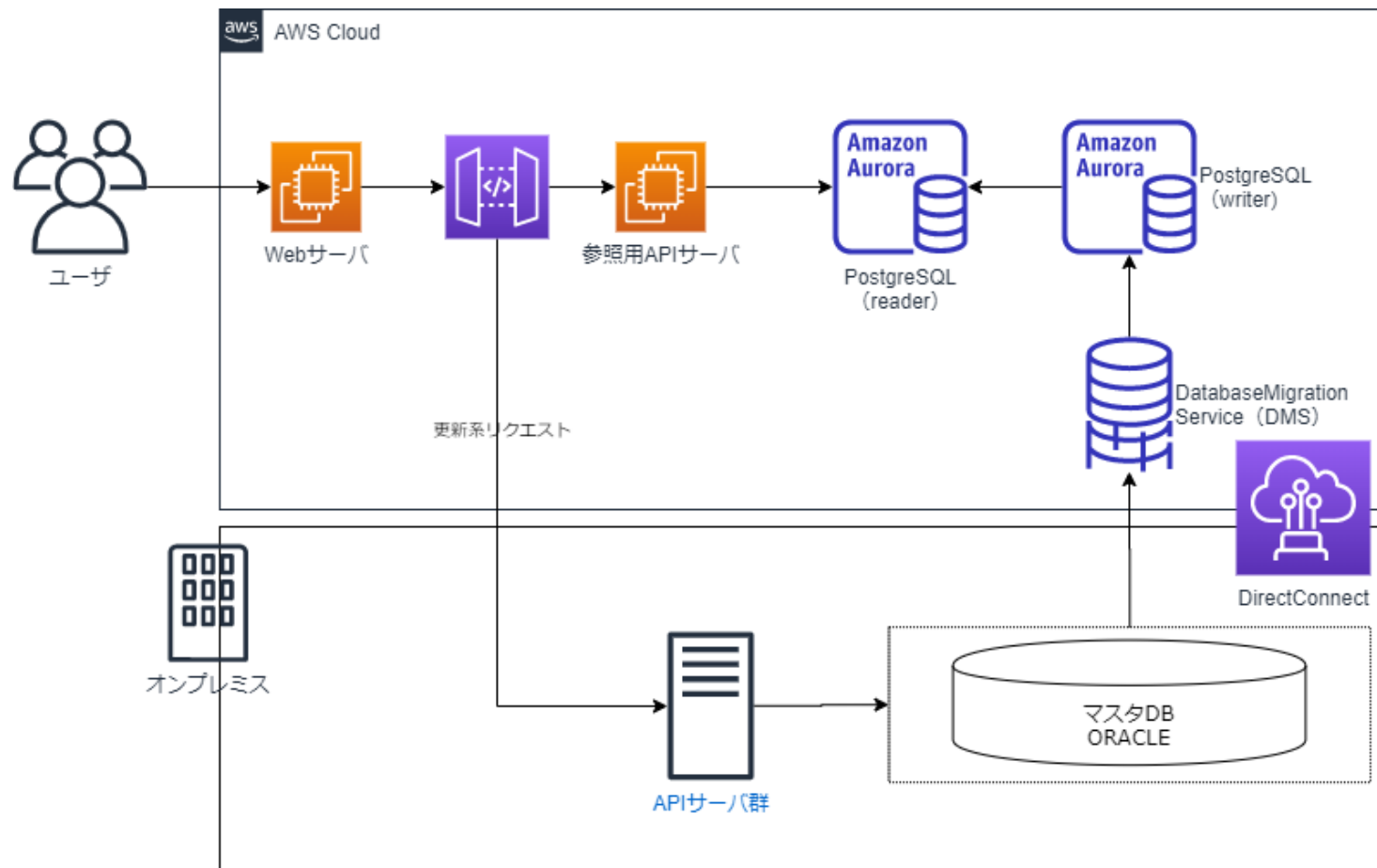
参照系のリクエストによる負荷を減らす

データベース負荷軽減の施策

- 1 AWS Database Migration Service の採用
- 2 Amazon Aurora PostgreSQL の採用
- 3 AWS プロフェッショナルサービス の採用
- 4 Amazon API Gateway の採用
- 5 アプリケーションのオートスケール の採用

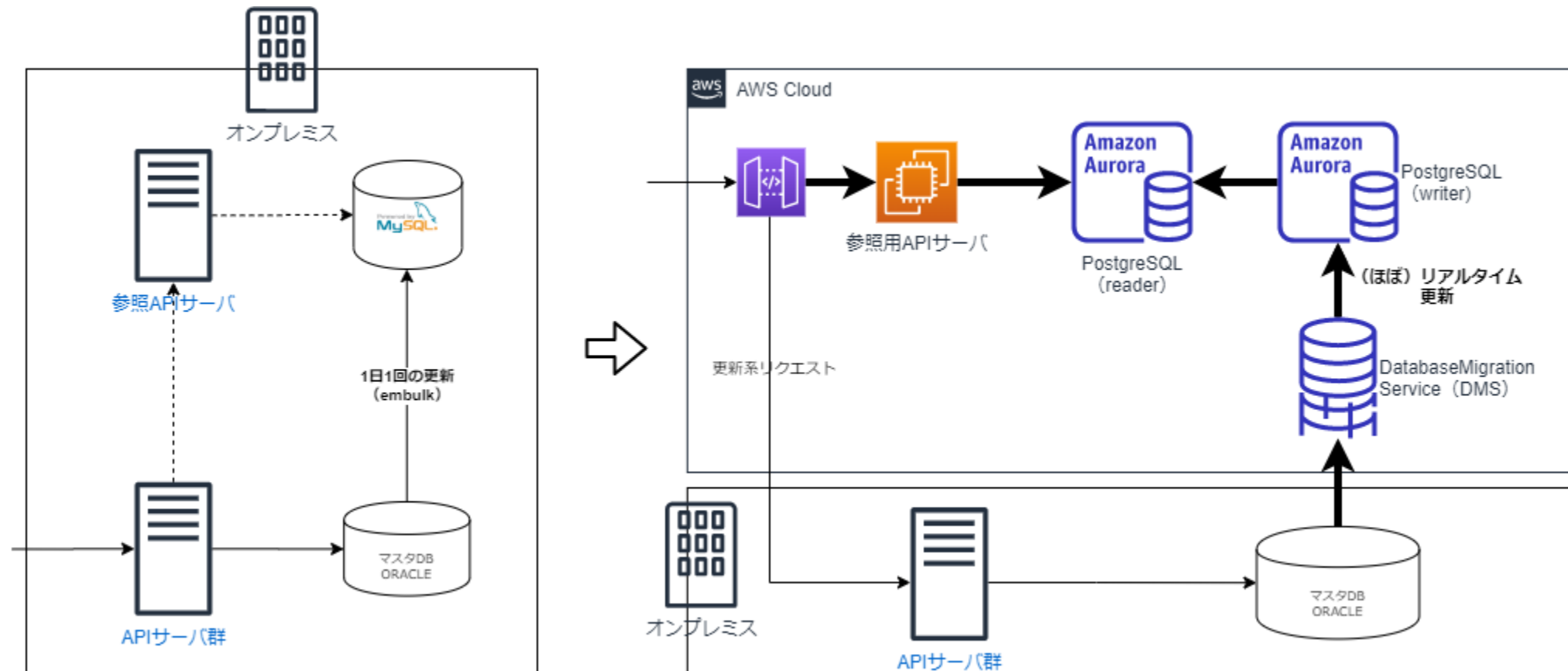
データベース負荷対策の構成

参照系のリクエストに必要なデータを別DBに移設し、AWS上で参照系リクエストを処理する。



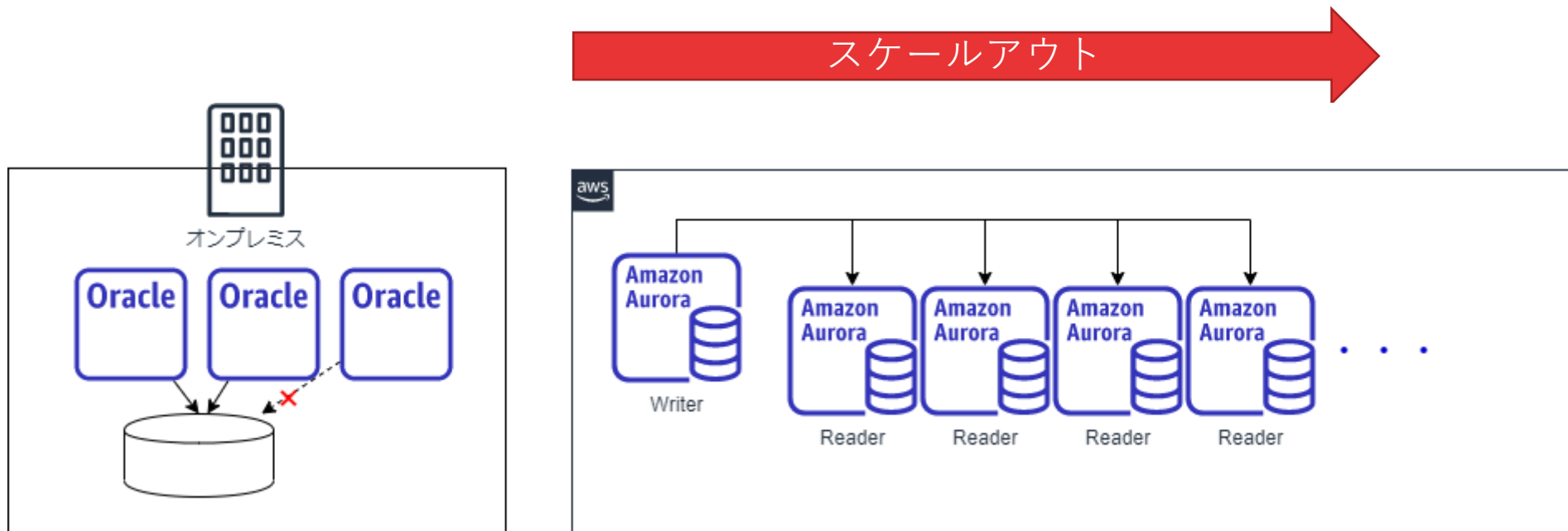
1. AWS Database Migration Service の採用

ほぼリアルタイムなデータ同期ができるAWS DMSを採用することで、同期対象のテーブルや参照APIを増やすことができ、DBサーバの負荷低減を実現した。



2. Amazon Aurora PostgreSQL の採用

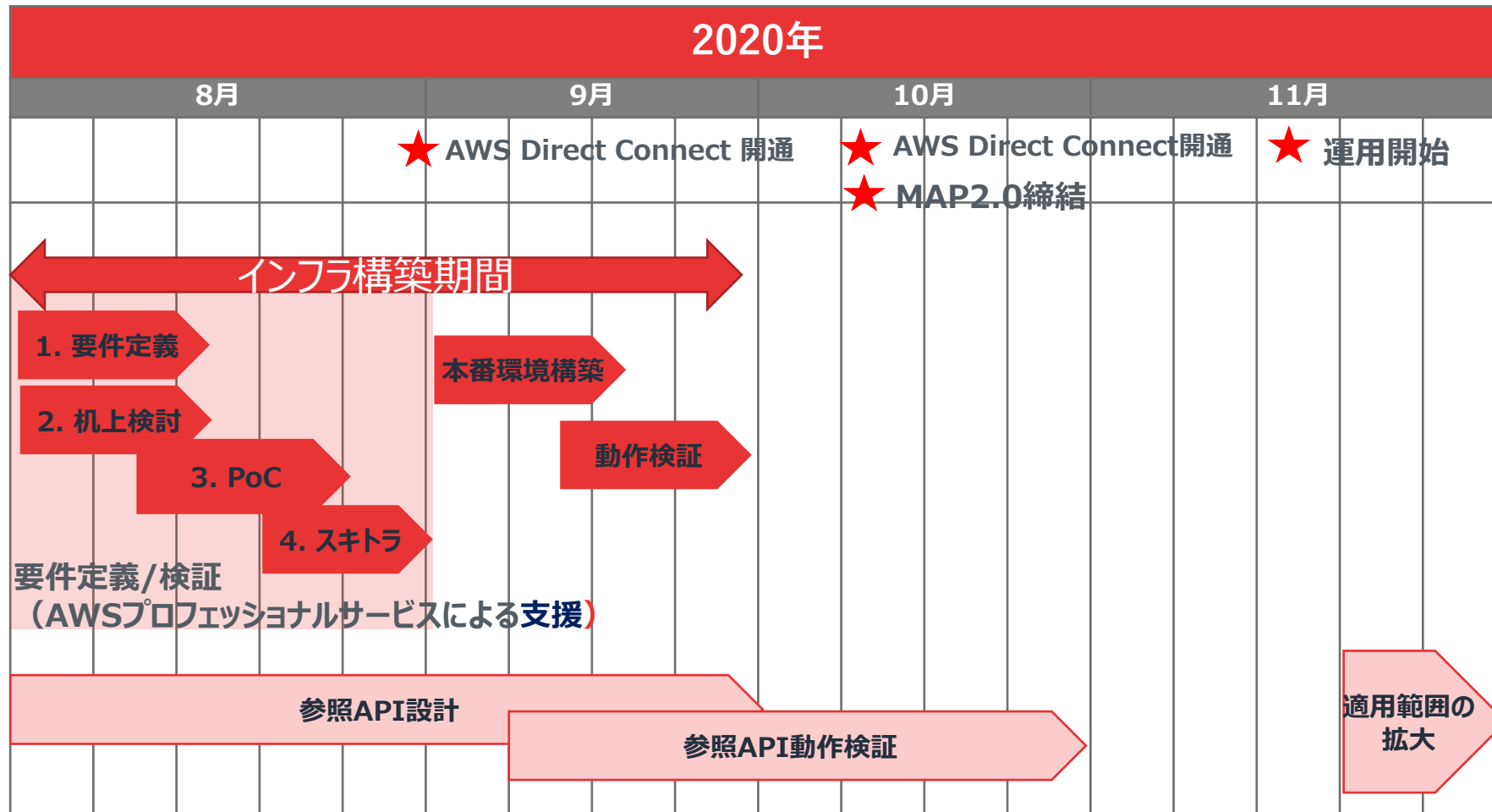
開発コスト、運用コストを考慮して、Oracle との親和性が高く、
スケーリングの管理が容易である Amazon Aurora PostgreSQL を選定した。



3. AWSのプログラムの利用 (AWS プロフェッショナルサービス・MAP2.0)

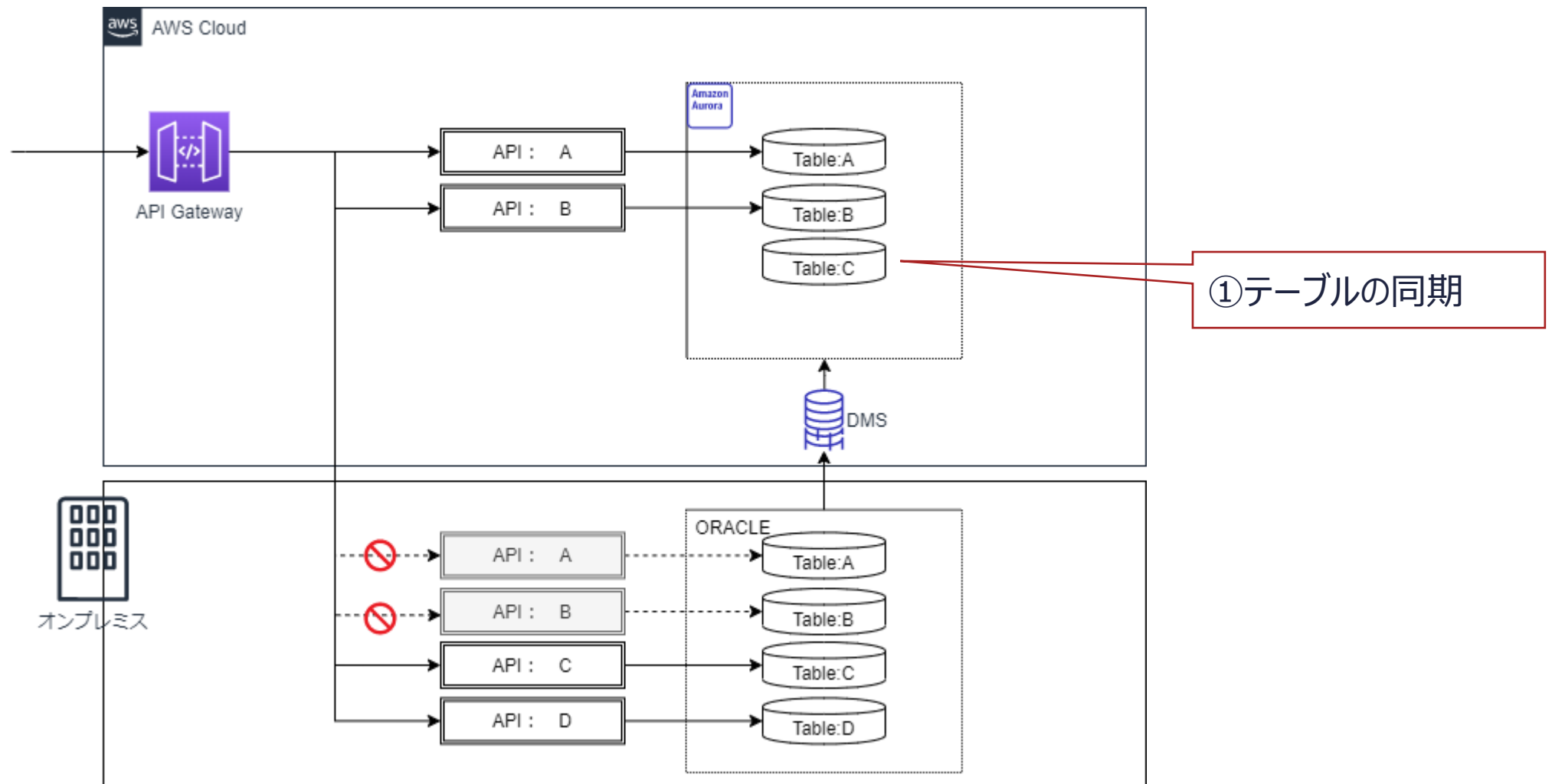
要件定義・DMSの選定から、AWS プロフェッショナルサービスに参画頂き、インフラ構築作業は2か月程度で完了した。

また、DB Freedom Workshop、MAP2.0といった移行支援プログラムも利用した。



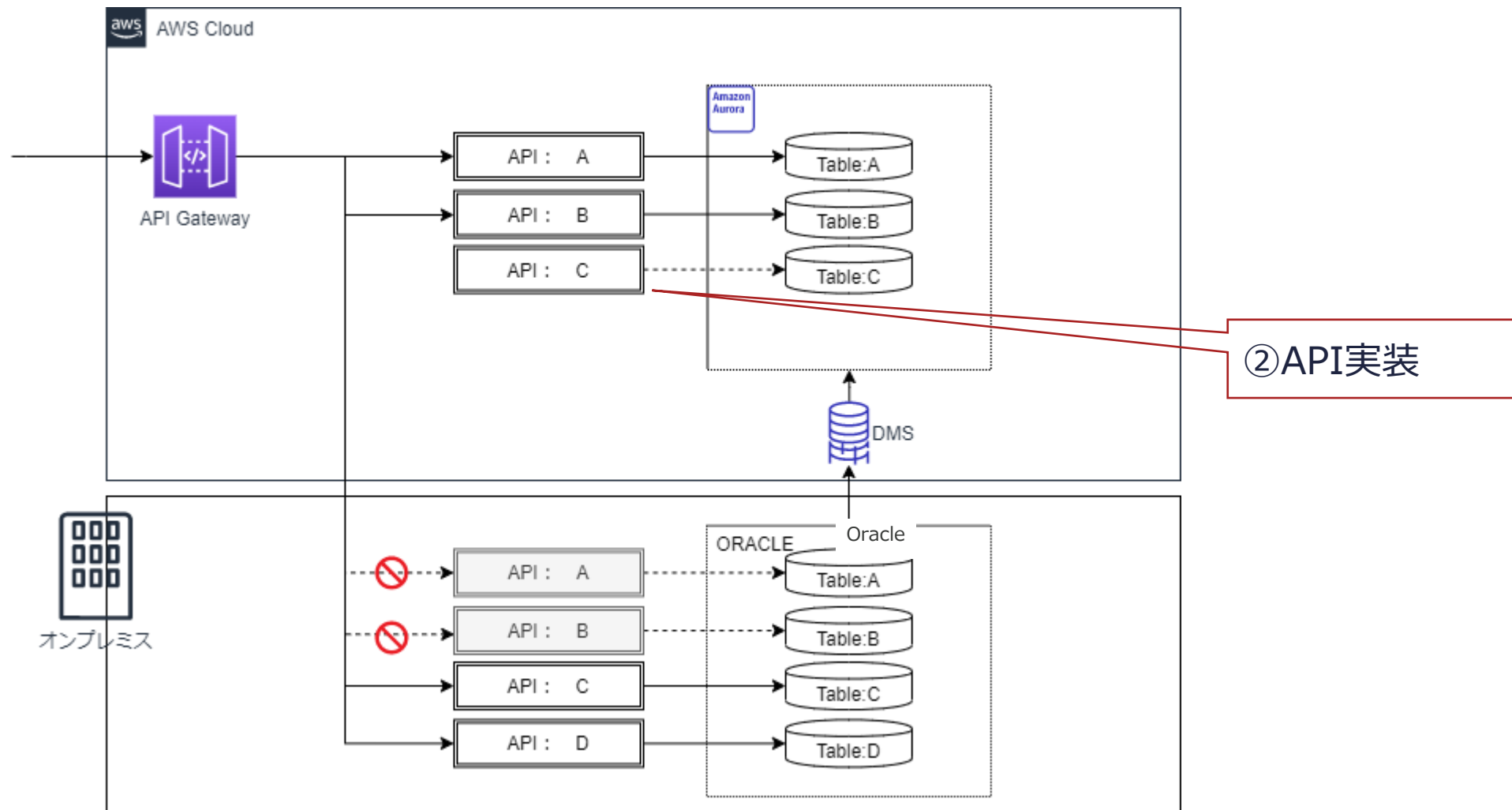
4. Amazon API Gateway の採用

データ同期の対象テーブル、参照APIのリリースを段階的に実施するため、Amazon API Gatewayによる新旧APIの切り替えを行った。



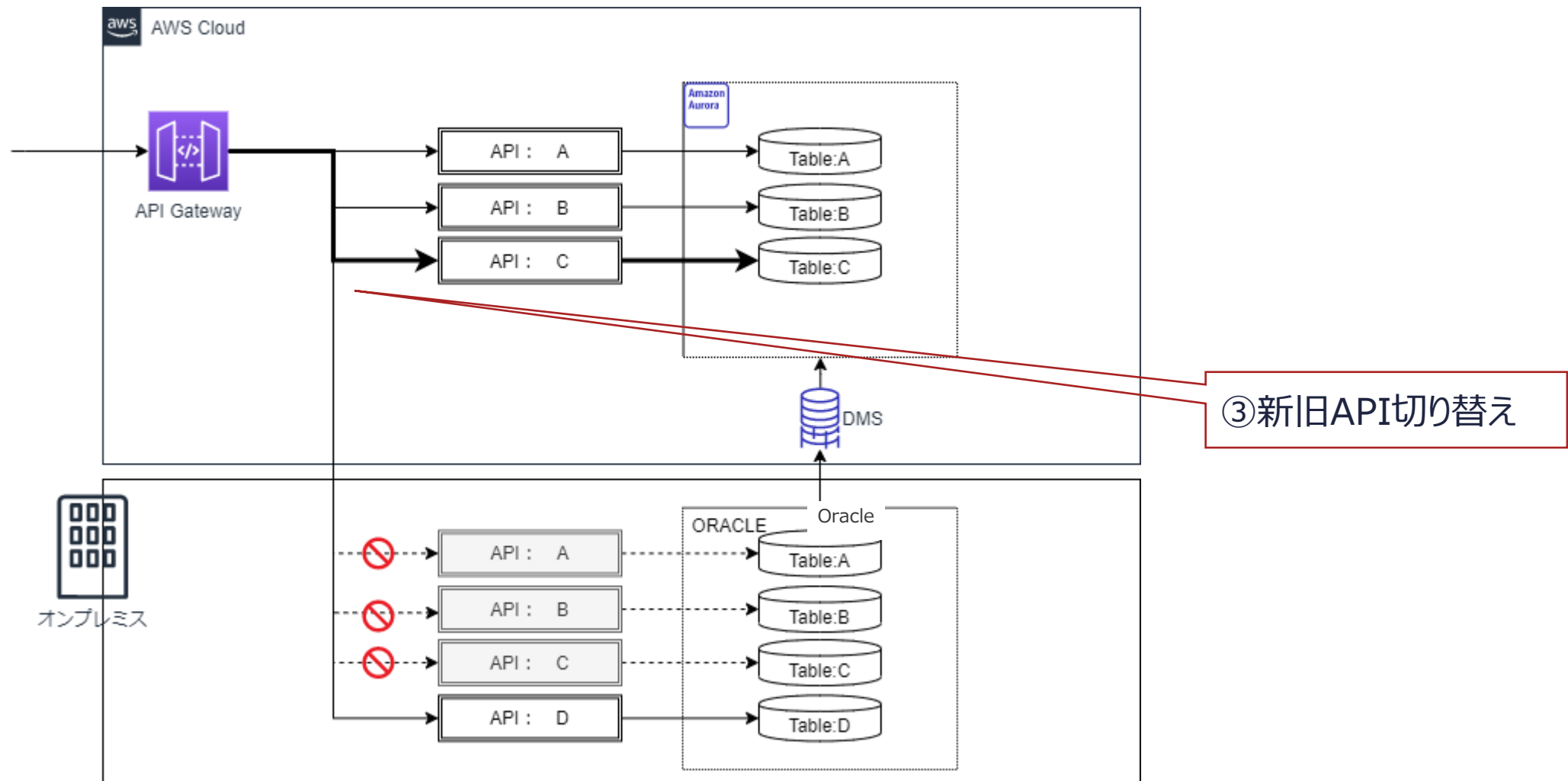
4. Amazon API Gateway の採用

データ同期の対象テーブル、参照APIのリリースを段階的に実施するため、Amazon API Gatewayによる新旧APIの切り替えを行った。



4. Amazon API Gateway の採用

データ同期の対象テーブル、参照APIのリリースを段階的に実施するため、Amazon API Gatewayによる新旧APIの切り替えを行った。



5. アプリケーションのオートスケールの採用

ピークタイムやプロモーションによる急激なリクエスト数の増加に対応するため、アプリケーションはサーバレス（AWS Fargate）を採用し、スケーリングは、CPU使用率ではなく、コネクション数、スループットに応じて実施した。

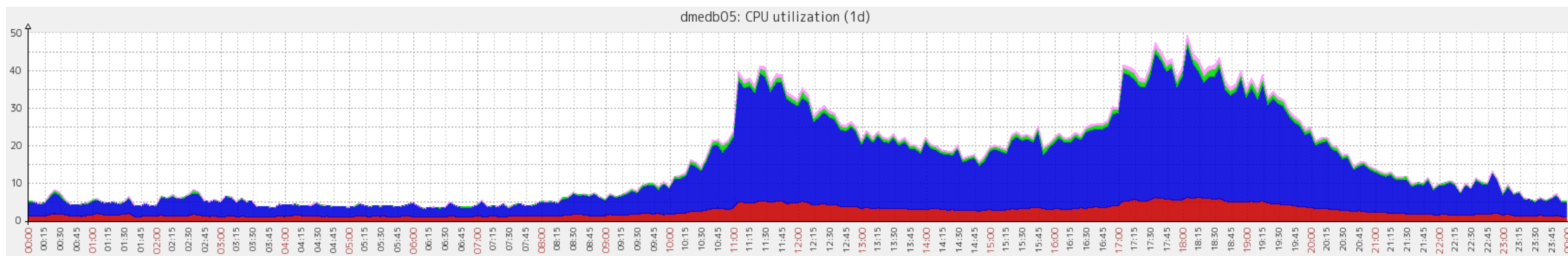


Step Scaling Policy
急激なリクエスト増には、
コネクション数の増加見合いによって
タスク数の増加を行う。

Target Tracking Scaling Policy
緩やかなリクエストの増減に対して、
ALBのターゲット毎のリクエスト数が
一定数以下になるようタスク数の増減を行う。

AWSサービス採用後のデータベースの負荷状況

『ウルトラ半額祭』開催時に前年比約2倍のオーダ数になっても、CPU使用率は低下した。



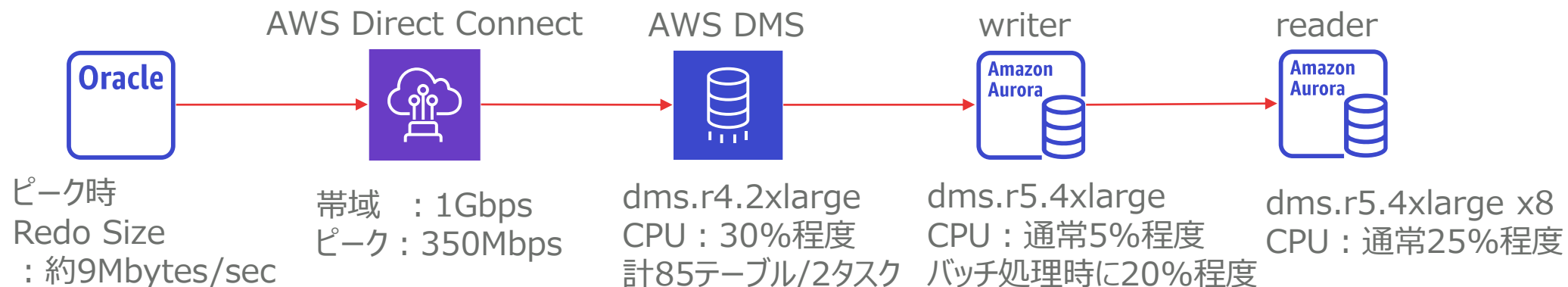
よかった点

- ◎キャパシティが向上し、ピーク時と通常時のCPU使用率の差分が小さくなる。(Oracle)
- ◎ピーク時の運用（スケールアウト）が容易になる。(Amazon Aurora PostgreSQL)

注意点

- △PostgreSQL固有のチューニングが必要。
- △Oracleと同じSQLだと性能が劣化する場合もある。
※オンプレOracleよりもRDSの方がリソースを消費

データベース負荷対策 まとめ



よかった点

- ◎ ほぼリアルタイムのデータ同期を安定的に実現。
レイテンシ：ピーク時でも10秒以下（ターゲット側）
遅延が出ても追いついてくれる。
- ◎ 運用監視も容易。
エラー通知なども他のAWSサービスと同様にできる。
- ◎ 低コスト。
AWS DMSインスタンスサイズに依存するのみ

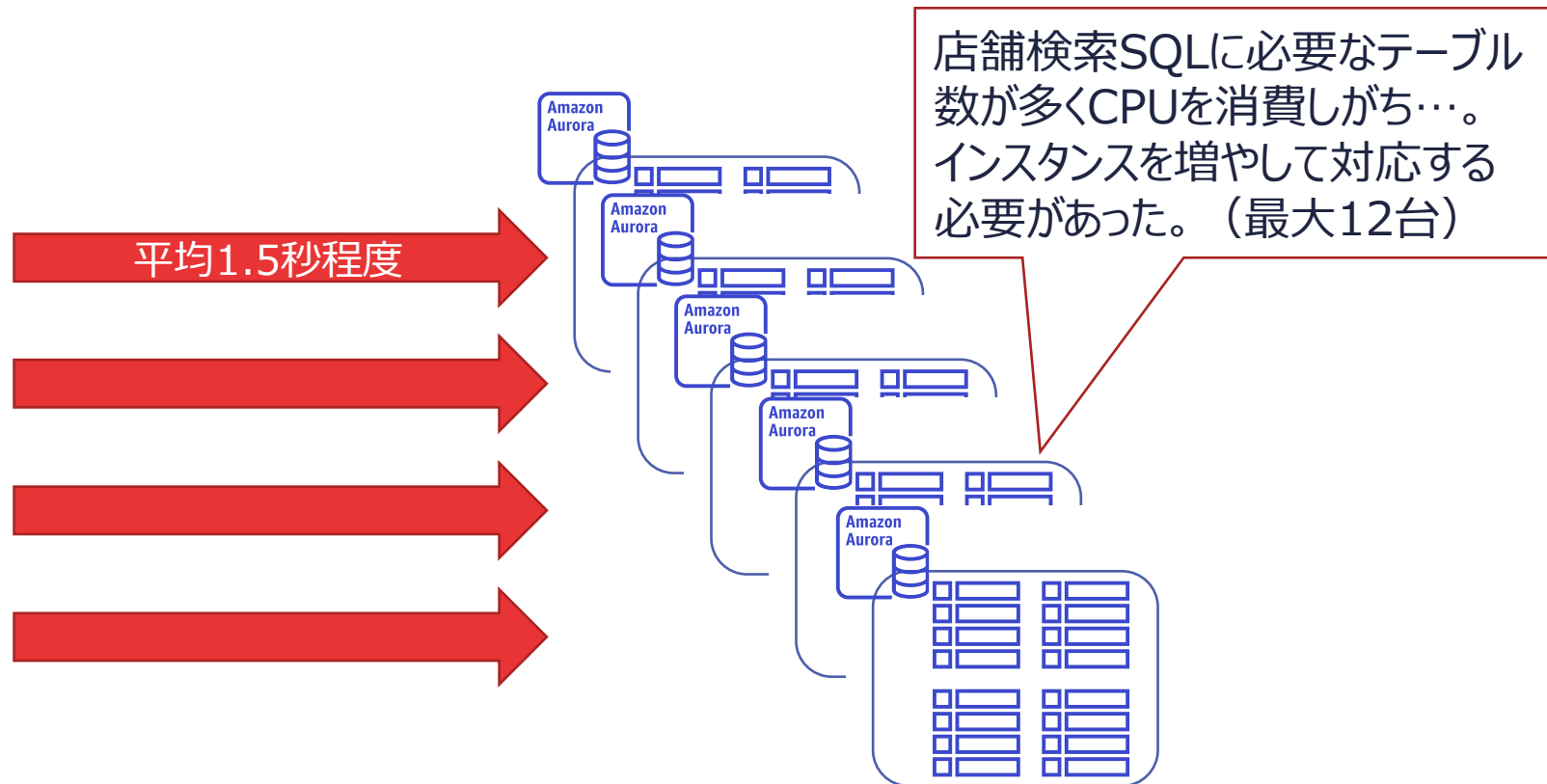
注意点

- △ 不向きなテーブルもある。
 - ・ 大きいテーブル、更新頻度が高い。
 - ・ 大量更新（delete/insert）でcommitまで時間がかかるテーブル
 - タスク内の他のテーブルも遅延する。
 - 一部のバッチ処理はAWS側でも実装。
- △ NW転送量は、Redoサイズ x タスク数になるのでNW帯域の監視を忘れずに！

- 01 出前館について
- 02 ユーザー数の急増とキャパシティ限界
- 03 AWS本格利用によるスケーラビリティの改善
- 04 レスポンスタイム向上への取り組み**
- 05 現在の取り組みと今後の計画

店舗検索が遅くて重い

DBサーバの負荷は下がったものの、従来から課題であった店舗検索等のレスポンスタイムは改善されておらず、また参照DBサーバの負荷も高い状態でした。



Amazon OpenSearch Serviceの採用

全文検索の機能強化とLINEデリマでの利用実績、スケールアウトの容易さから Amazon OpenSearchを採用した。



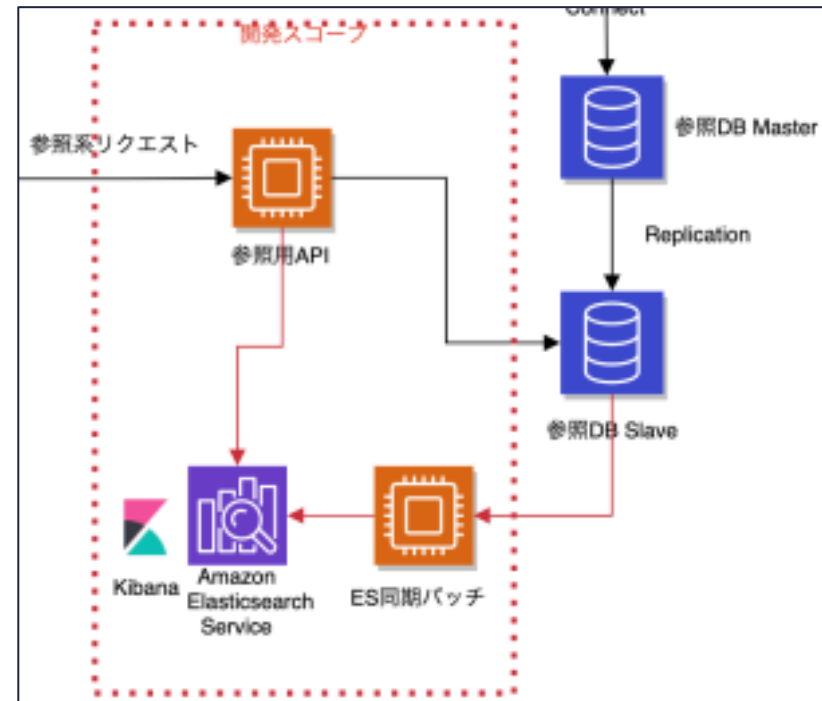
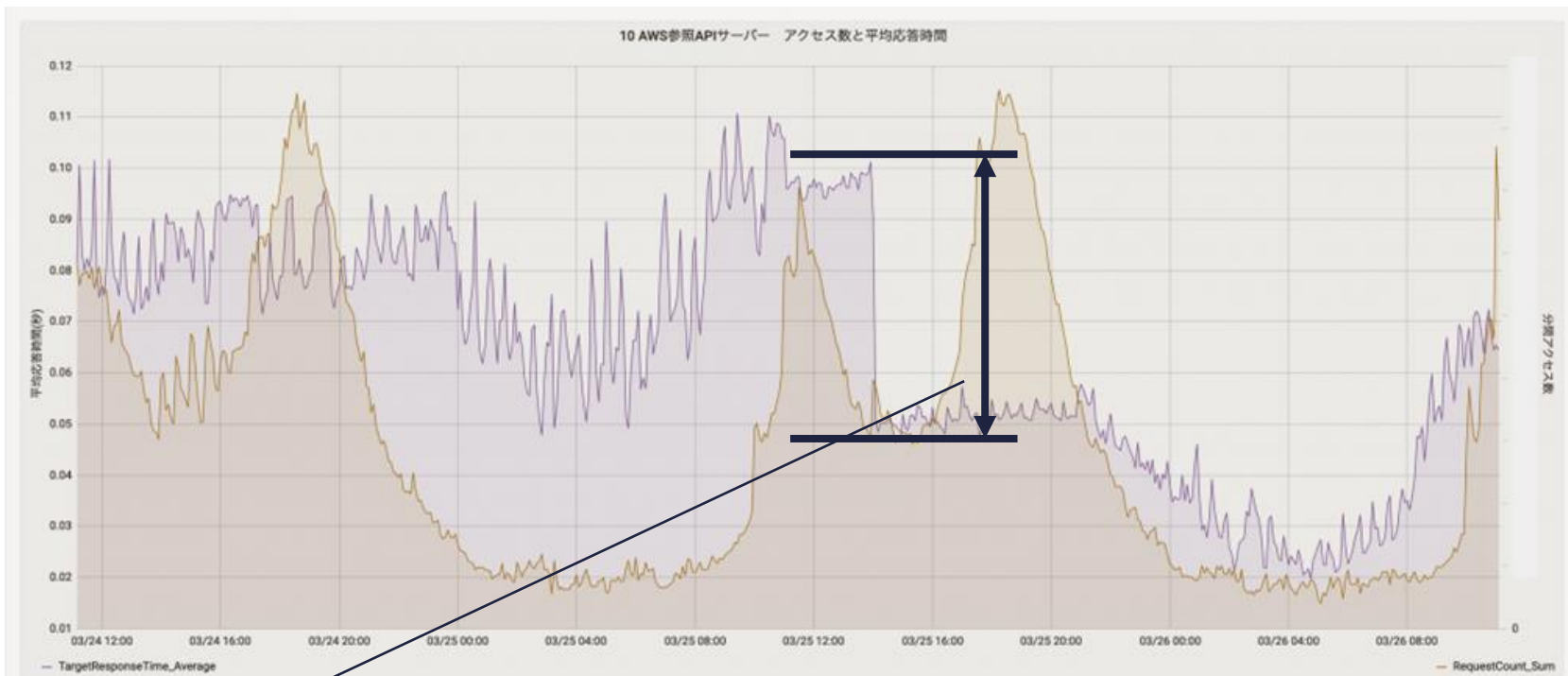
全文検索用のRDBを用意していたが、レスポンスが遅すぎてほとんど使われていなかった…。



LINEデリマ（2020年12月 出前館に統合）で、出前館と同じデータを用いた利用実績があり、LINEエンジニアの知見も得られた。

Amazon OpenSearch Service 導入の効果

検索のニーズに合わせたスキーマ設計を行い、検索処理のレスポンスタイムを改善した。



店舗検索の平均レスポンスタイムは平均1.5秒から0.4秒程度まで改善しており、Amazon OpenSearchを適用するAPIを増やしている。

01 出前館について

02 ユーザー数の急増とキャパシティ限界

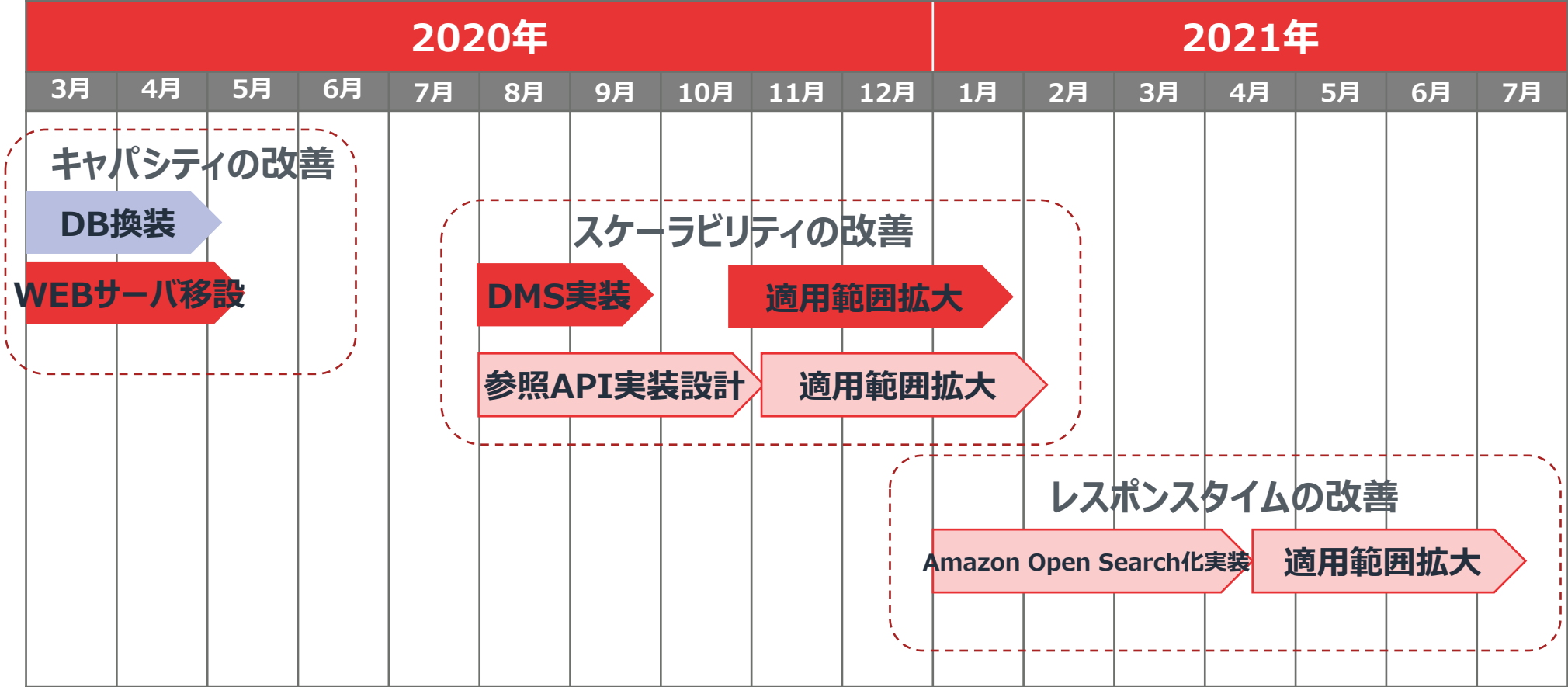
03 AWS本格利用によるスケーラビリティの改善

04 レスポンスタイム向上への取り組み

05 現在の取り組みと今後の計画

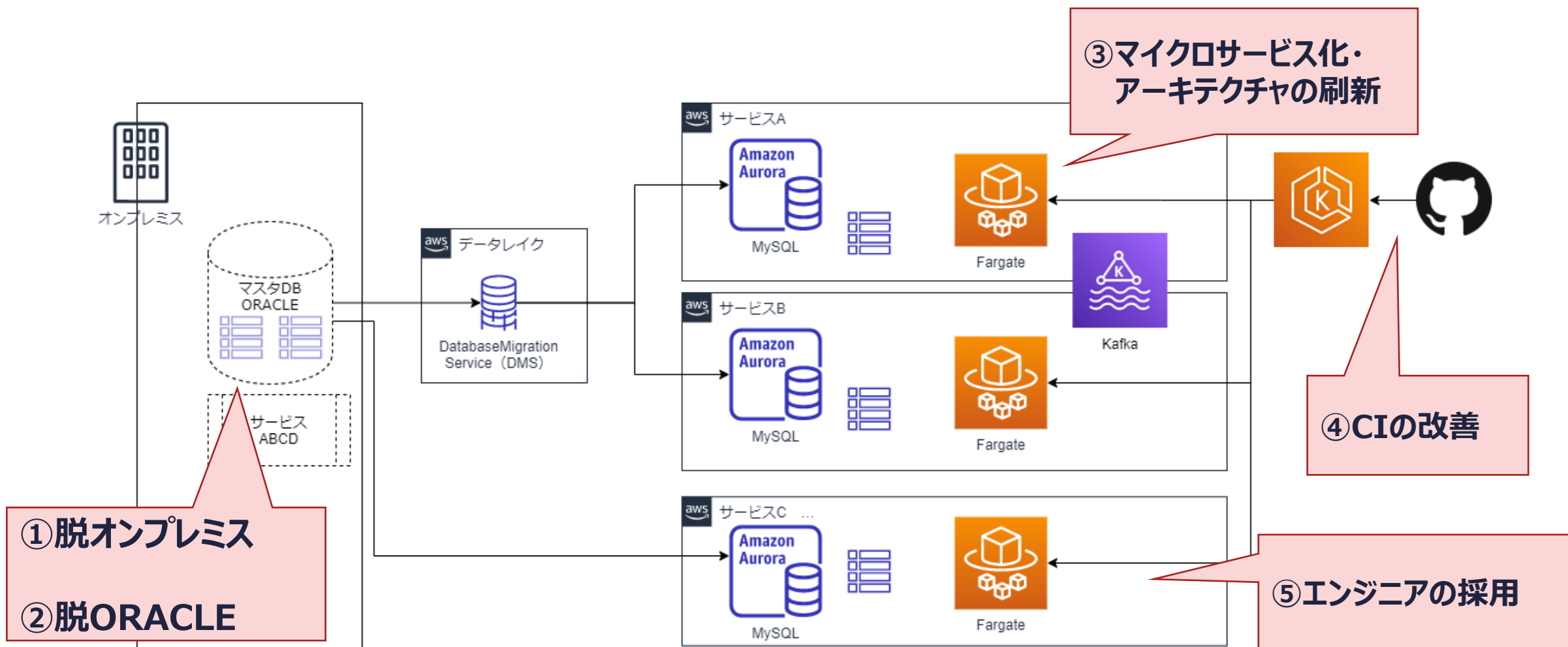
これまでのまとめ

AWSを用いてスケーラビリティの確保、レスポンスタイムの改善に概ね1年を費やし、
拡大するサービスを支えることができました。



現在の取り組みと今後の計画

まだまだ課題はありますが、スケーラビリティを確保しつつ、開発スピードを上げるためにAWSの各種サービスを積極的に活用していきたいと考えています。



最後に – エンジニアを積極採用中です！



<https://recruit.demae-can.com/engineer-recruitment/>

Thank you!

岡田 泰弘

株式会社出前館

プロダクト開発本部 開発管理部

