

# BCPの改善へ向けた可用性向上 ～RDS周りを中心に～

株式会社マーズフラッグ  
サービスプラットフォーム部  
佐々木 崇之

# 自己紹介

株式会社マーズフラッグ  
サービスプラットフォーム部

佐々木 崇之  
Takashi Sasaki

- 出身
  - 千葉県船橋市
- 趣味
  - ゲーム
  - コンピュータいじり
  - オーディオ
  - 押し事
- 仕事
  - 2015年入社 SREチームに従事
  - 自社サービスのインフラ構築、運用、改善
  - 社内システムのインフラ構築、運用、改善

# 会社紹介

## 株式会社マーズフラッグ

1998年10月 設立

### 事業内容

サーチテクノロジーをベースとした  
Webサービスの提供

“BEYOND EMOTION

サーチを技術から感動へ”

をスローガンに掲げ、より良い「検索」を提供できるよう切磋琢磨する組織です。

### 主な沿革

- 2006年
  - 見える！サイト内検索  
「MARS FINDER」サービス提供開始
- 2021年
  - 次世代型DXソリューション  
「AIMY Screen」β版サービス提供開始
  - API連携クラウドサービス  
「MARS Platform」サービス提供開始
- 2023年
  - チャット型AI対話サービス  
「MARS EmoTech」α版サービス提供開始
  - サイト内検索「MARS FINDER」  
メジャーバージョンアップ



MARS FINDER

国内シェアNo.1\*

\*株式会社富士キメラ総研「ソフトウェアビジネス新市場 2023年版」  
検索エンジン市場 (SaaS) 2022年度より

# 当社サービス(プラットフォーム)の変遷

- ～2016年「オンプレ期」
  - ハードウェア故障頻度の上昇、これに伴うダウンタイム増。
  - ハードウェアのリプレイスではなく、クラウドへの移行計画を開始。
- 2016年～2019年「AWSへの移行期」
  - 構成は変えず、そのままの形で EC2へ段階的に移行。
  - EC2以外のAWSサービスを利用する機会が増える。
  - よりモダンにAWSの機能を存分に活かすニーズが高まる。
  - モノリックなアプリケーションからマイクロサービス化したいニーズが高まる。
  - 当社サービスおよびプラットフォームのフルリニューアルを計画。
- 2019年～「フルリニューアル期」
  - 段階的にAWSマネージドサービスの利用拡大。
  - Lambda等のサーバレスの利用拡大。
  - 将来的なマルチリージョン対応も視野に入れた、高可用性なプラットフォームを設計。
  - AWS CDKおよびCloudFormationを活用しインフラをcode化。
  - プロダクト単位にAWSアカウントを分離。
  - ベストプラクティスに近づくため、AWS Well-Architectedを活用。

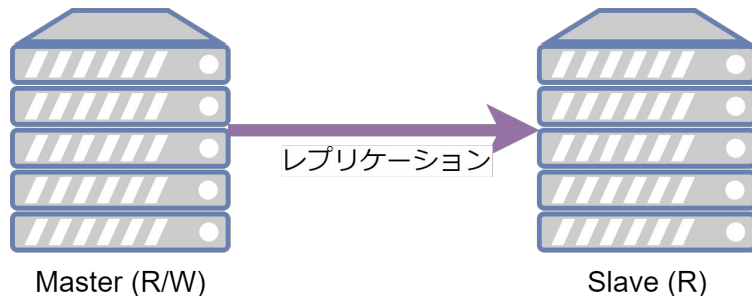
# RDBの可用性向上①「オンプレ期」

構成 ⇒ 災害やDC障害への耐性が無い(幸いにして経験は無い)

- 単一のデータセンタ内において、Master-Slave構成のレプリケーション。
- ネットワークと電源を二重化。
- ストレージを多重化。

運用 ⇒ 手間がかかる、故障要因が多岐に渡る(つらい)

- リカバリ手順は完全に手動。
- 故障時は少なくとも30分程度のダウンタイムが発生する。  
ハードウェア故障の場合はダウンタイムが数時間に及ぶケースも。
- レプリケーションの状態監視が必要。
- 故障の度合いにより、読み取り一貫性が失われる事も。
- ハードウェア故障の場合、現地への移動が発生する。



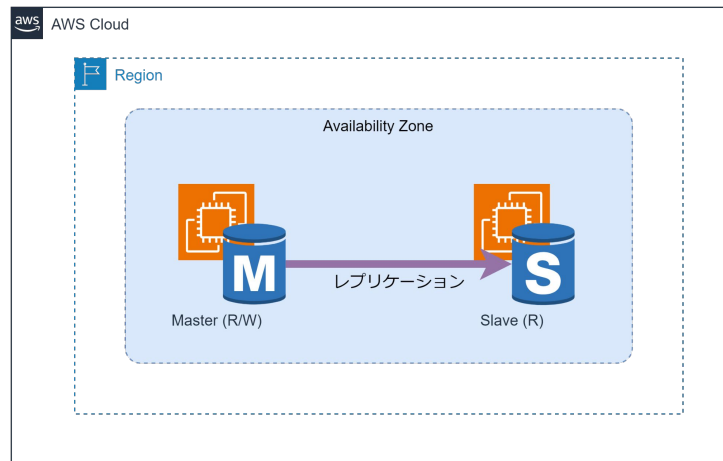
# RDBの可用性向上②「AWSへの移行期」

構成 ⇒ ベストプラクティスにはほど遠く、AZ障害への耐性も無い

- 訳あってオンプレ時の構成をそのままC2に乗せたため、EC2上で動かすMySQLのMaster-Slave構成。
- さらに訳あってEC2インスタンスを単一のAZに配置している。

運用 ⇒ ハードウェア故障から開放されるも、まだ手間を要す

- リカバリ手順は完全に手動。
- 故障時は少なくとも30分程度のダウンタイムが発生する。
- レプリケーションの状態監視が必要。



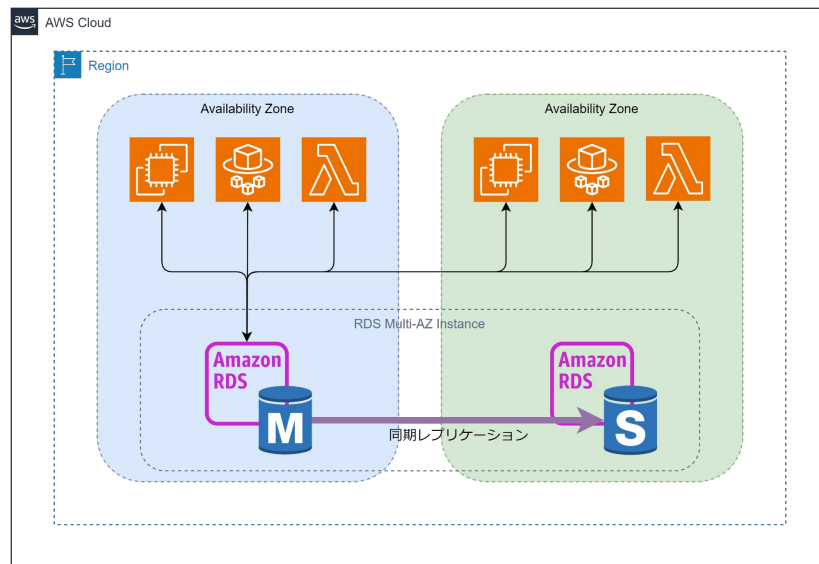
# RDBの可用性向上③「フルリニューアル期」

構成 → AZ障害への耐性が向上。リージョン障害への耐性は無い。

- Amazon RDS マルチAZインスタンス。
- これまでのSlaveはRead可能であったが、ReadReplicaでは無いためStandbyへアクセスは出来ない。
- 異なるAZに配置される。

運用 → 手間はほぼゼロ！

- リカバリは完全に自動で行われる。
- 故障時のフェイルオーバーが60秒程で完了し、ダウンタイムが大幅に減少する。
- レプリケーションの状態監視が不要。



# 導入して良かった点

Amazon RDSマルチAZインスタンスを導入して良かった点を紹介します。

- 可用性が向上した
  - 災害やAZ障害など低頻度なものへの耐性向上のみならず、フェイルオーバーが自動かつ非常に短時間で行われるため、ダウンタイムが大幅に減少した。
  - フェイルオーバーが自動で行われるため、人的ミスやリカバリ時のプレッシャーから解放された。
- 生産性のあるタスクに割ける時間が増加した
  - 運用に係る手間をほぼ自動化できるため、リソースの最適化や改善などに割ける時間が増加した。
- 精神的に楽になった(案外大きいメリット)
  - データロストやサービス停止を心配するストレスから解放された。



# 今後の課題

まだまだフルリニューアルの経過途中であり課題は沢山ありますが、RDSに絞って将来的な展望へ向けた課題を紹介します。

- Amazon RDS マルチAZは異なるリージョンを跨げないため、将来的にマルチリージョン対応となった場合へ向けて構想を固めておきたい。
- スケールアップやスケールアウトが必要と判断する基準の作成および、手順の確立。

ご清聴ありがとうございました