

安定したリリースと運用を支えた Google Cloud活用事例

株式会社ミクシィ デジタルエンターテインメント事業本部

植森 康友

XFLAG

アジェンダ

自己紹介

会社紹介

本番構成

Cloud Deploy活用事例

Google Kubernetes Engine活用事例

Cloud Spanner活用事例

まとめ

自己紹介

株式会社ミクシィ デジタルエンターテインメント事業本部 所属

名前: 植森 康友

HN: wakaba260



前職においてGoogle Kubernetes Engineを利用したゲーム開発の他、BigQueryを利用したログ基盤の構築に従事し、2020年5月に株式会社ミクシィへと転職。

現在はAPI開発とリリースに向けてのインフラ構築を担当している。

Cloud Summit Osaka '18にてCloud Hero受賞。

そのまま同日に行われたGCPUG Kansai Summit Dayにて登壇した。

会社紹介



(C) mixi, Inc.

©XFLAG

会社紹介

mixi GROUP

“コミュニケーション”を事業ドメインとして、モンスターストライクやコトダマン、スタースマッシュなどゲームやアニメを含むデジタルエンターテインメント、公営競技や千葉ジェッツなどを含むスポーツ、家族アルバム みてねやminimo、SNS「mixi」などを含むライフスタイルの3つの領域で事業を推進しています。



デジタルエンターテインメント



スポーツ

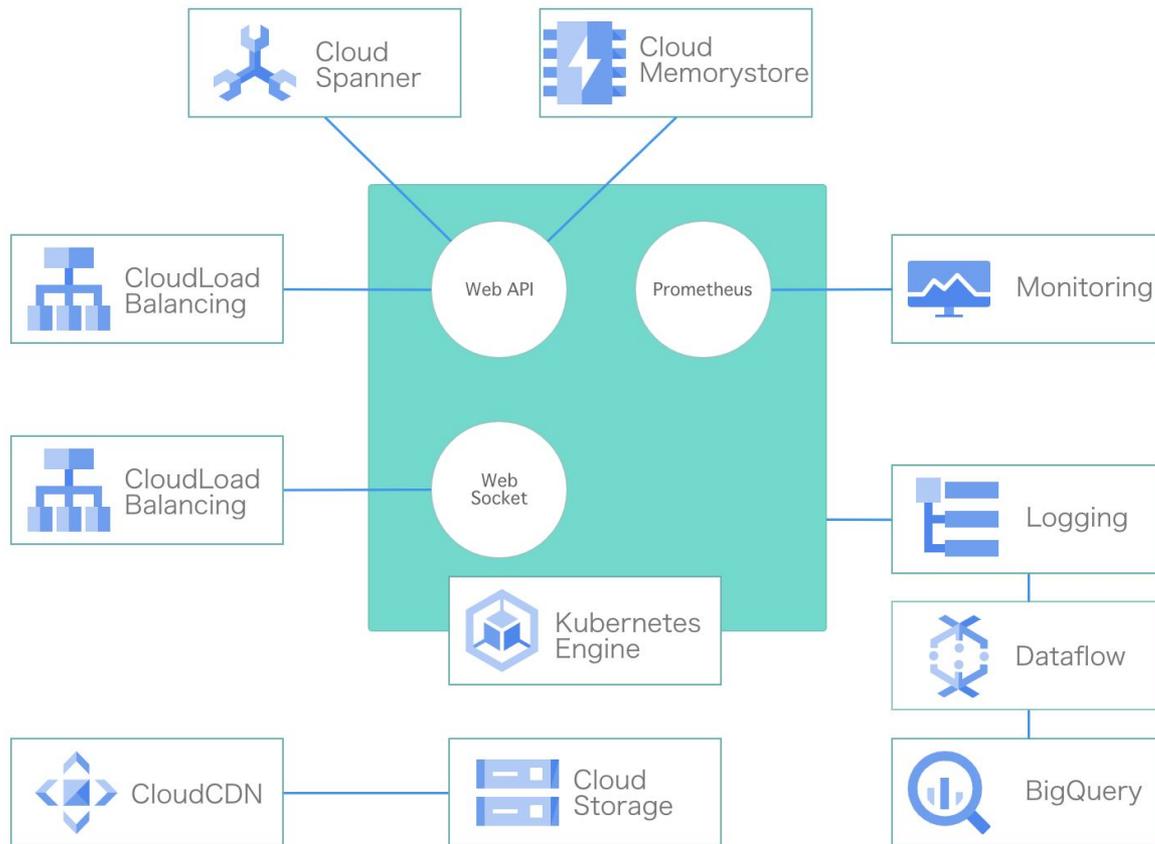


ライフスタイル



本番構成

本番構成



Cloud Deployの活用

Cloud Deployとは？

- Google Cloud Consultingのソリューションの一つで、
Googleのエンジニアにプロジェクトに参加してもらえるサービス
- アーキテクチャなどのレビューやトラブル発生時のサポートを受けることが出来る他、ディスカッションや改善提案をしてもらうことが可能
- 現プロジェクトではリリースの二ヶ月ほど前から利用開始

Cloud Deployのサポート内容

- 週に一度の定例MTG
- Googleのエンジニアによるアーキテクチャレビュー
- リリース時の予測流入に対する内部リソースの相談
- サポートケース対応
- 各種エッジケースの内部実装調査や内部挙動検証

定例MTG

週に一度実施。1～2時間程度。

- アーキテクチャについての説明とQ&A
- リリースまでのスケジュールの共有
- アーキテクチャレビューのフィードバックとディスカッション
- 負荷試験のサポート
- Googleのサービスを利用する上での実装や運用についてのQ&A

アーキテクチャレビュー

- Cloud Spannerのスキーマレビュー
- Google Kubernetes Engine、Cloud Spanner、Google Cloud Load Balancing、ロギングなどを含めた全体の構成
- 各種インフラの利用用途と実装
 - ▶ 単一障害点がないか
 - ▶ Google Cloudの利用方法として間違っている点がないか
 - ▶ より良いアーキテクチャへのアドバイス

リリース時の予測流入に対する内部リソースの相談

- ゲームでは初期流入に備えて多くのリソースが必要
- あらかじめ想定DAUやRPS、QPSなどを定例MTGで共有
- CPU/Memoryなどだけではなく、ネットワークなどの内部リソースを確保するよう調整してもらった

負荷試験サポート

- 負荷試験のシナリオと実施日時について定例MTGで共有
- 実施中のメトリクスをCloud Deploy側でも監視
- APIのレイテンシが高いところなどは協力して調査
 - ▶ Cloud Spannerのパフォーマンスチェック
 - ▶ 実装についてディスカッション

アーキテクチャレビューによる改善例①

Redisの可用性向上のための実装提案

- Redisでキャッシュやルーム管理を行う実装になっていたが、
Cloud SpannerよりMemorystoreの方が可用性が低いという指摘があった
- 可用性を上げるためにどうすればいいかをディスカッション&改善
 - ▶ Cloud Spannerに変更できる箇所は実装を修正
 - ▶ Redisをシャーディングする実装を入れて複数台構成に

アーキテクチャレビューによる改善例②

Cloud Spannerのホットスポット回避のためのID生成案

- Cloud Spanner利用前のID生成の実装が単調増加なINT64の値になっていた
- UUIDの利用を推奨されたが、実装の大きな変更が必要になるため、INT64のまま非単調増加・減少ではないIDの生成方式を検討
- ビットシャッフルなど、いくつかのID生成方法を提案してもらった
 - ▶ 最終的にはUUID方式に切り替えたが、検討の大きな助けになった

Google Kubernetes Engineの活用

利用バージョンの選定

本番ではKubernetes1.17を利用

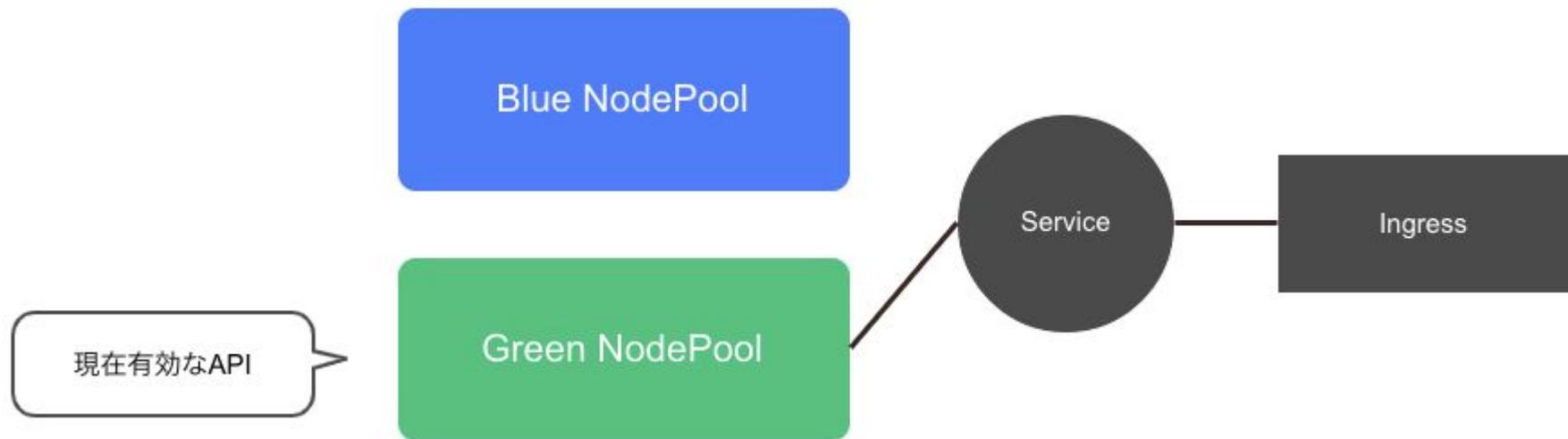
- ログの流量が多いことが懸念されたため、fluent-bitを使えるように
 - ▶ fluentd-plugin-google-cloudと挙動の異なる点があったので注意
- 1.17からデフォルトでNetwork Endpoint Groupが有効になる
 - ▶ Cloud Deployで検証結果をもらい、利用しないことにした

Blue / Greenによるオンラインアップデート

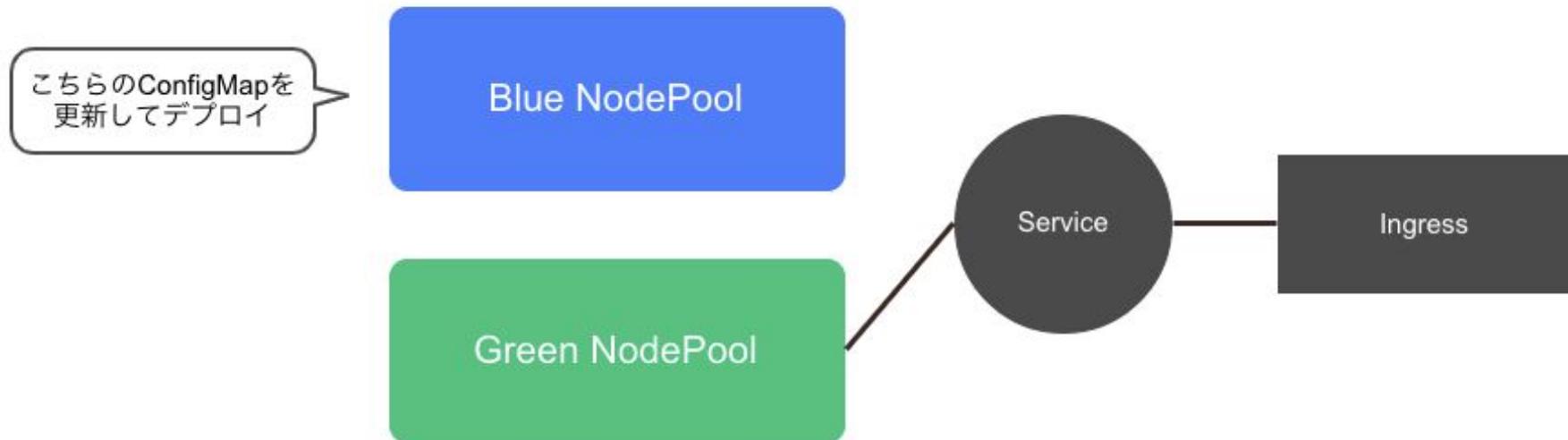
マスターデータのアップデート方式

- マスターデータはPod起動時にCSVをCloud Storageからダウンロード
- ダウンロード先のパスはConfigMapで設定しており、
データ更新時はConfigMapを変更してKubernetesへapplyする
- この際にRollingUpdateを使うと新旧のバージョンが混在する
 - ▶ この問題への対策として、Blue/Greenによるアップデートを実施

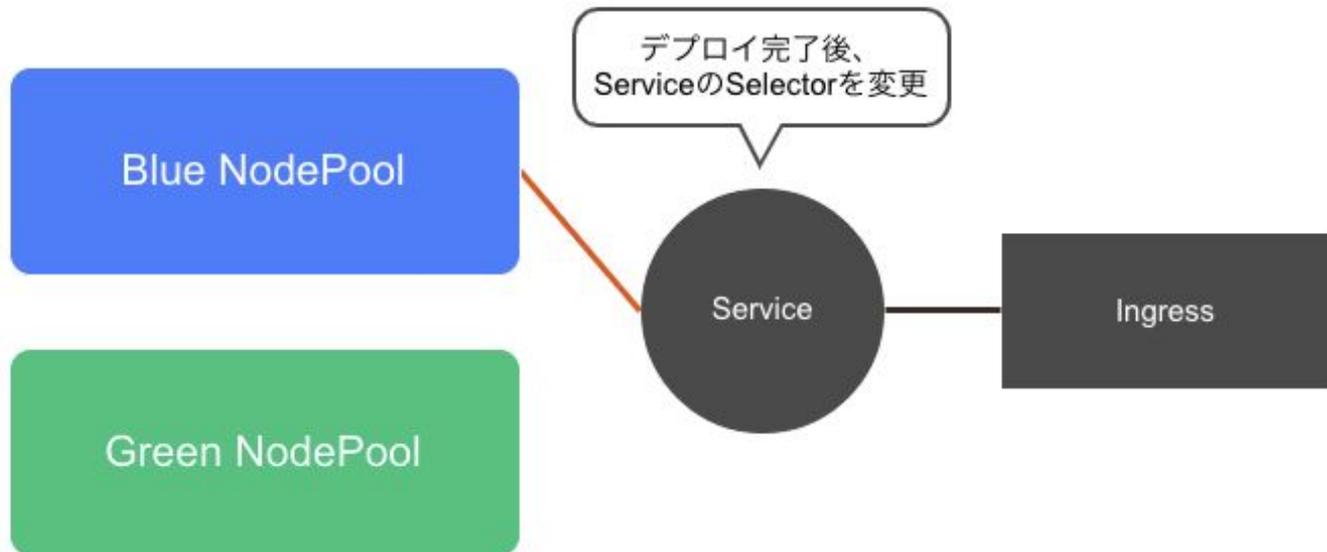
Blue / Greenによるオンラインアップデート



Blue / Greenによるオンラインアップデート



Blue / Greenによるオンラインアップデート



Blue / Greenによるオンラインアップデート

オンラインアップデートのためのBlue/Greenデプロイ

- Blue/GreenのAPIを用意し、現在参照しているものとは逆のものを更新
- デプロイ終了後、ServiceのSelectorを変更して参照先を切り替え
- Network Endpoint Group(NEG)下では、Serviceの更新に一貫性と即時性がなくなるので無効にした

Cloud Spannerの活用

Cloud Spannerの利用

データベースにはCloud Spannerを利用

- スケールイン・スケールアウト性能を評価
- サーバサイド言語がElixirだったためライブラリを内製
 - ▶ gRPCによる通信は行わず、REST API経由での利用
 - ▶ gRPCに比べて大きなレイテンシの悪化などはなかった

Cloud Spannerのテーブル設計

ゲームではユーザに対しINTERLEAVEする設計がしやすく相性が良い

- ユーザに紐づくテーブルは全てUserAccountに対しINTERLEAVE
- Cloud Deployでのスキーマレビュー
 - ▶ INTERLEAVE漏れ、INDEX設定漏れなどの指摘があった
 - ▶ PRIMARY KEYがホットスポットを回避できるかどうかのチェック

Cloud Spannerのトラブルシューティング

負荷試験中にサーバ実装やライブラリ側の問題でいくつかトラブルがあった

● レイテンシが高いAPIの問題の切り分け

- ▶ Cloud Traceに対しCloud Spannerへのリクエストをspanとして送信

● Session Leak問題

- ▶ session_countのメトリクスをCloud Monitoringで計測
- ▶ PrometheusでSession数を計測しCloud Monitoringへ送信

オープン前の暖機運転

ゲームのように初期流入の多いユースケースでは暖機運転が必要

- CPU使用率が下がると時間経過でSplitがマージされてしまうため
- 初期流入の負荷に耐えるためにあらかじめ暖機運転でSplitを作成しておく
 - ▶ 負荷試験のシナリオをリリース準備時に実施後、データを削除した

Cloud Spannerのデータエクスポート

KPIの解析用途でBigQueryでCloud Spannerのデータが必要だったが、Dataflowを利用したCloud SpannerのエクスポートはCPU負荷が高いため対策を行う必要があった

- エクスポート用のインスタンスを別で用意
 - ▶ 一度バックアップを取り、エクスポート用インスタンスへリストア
- エクスポート用インスタンスからDataflowでエクスポート
 - ▶ エクスポートしたデータはBigQueryへ入れて解析で利用

まとめ

まとめ

- Google Kubernetes Engineを利用することで、ゲーム運用において重要性の高い柔軟な運用やスケーラビリティを実現出来た
- Cloud Spannerは設計や運用がゲーム開発と相性がよく、スケーラビリティを担保しつつ今までの経験を活かすことが出来る
- なにより、Cloud Deployを利用することで安心してリリースを迎えることができた

Thank you for Google Cloud !

XFLAG