

基本設計書

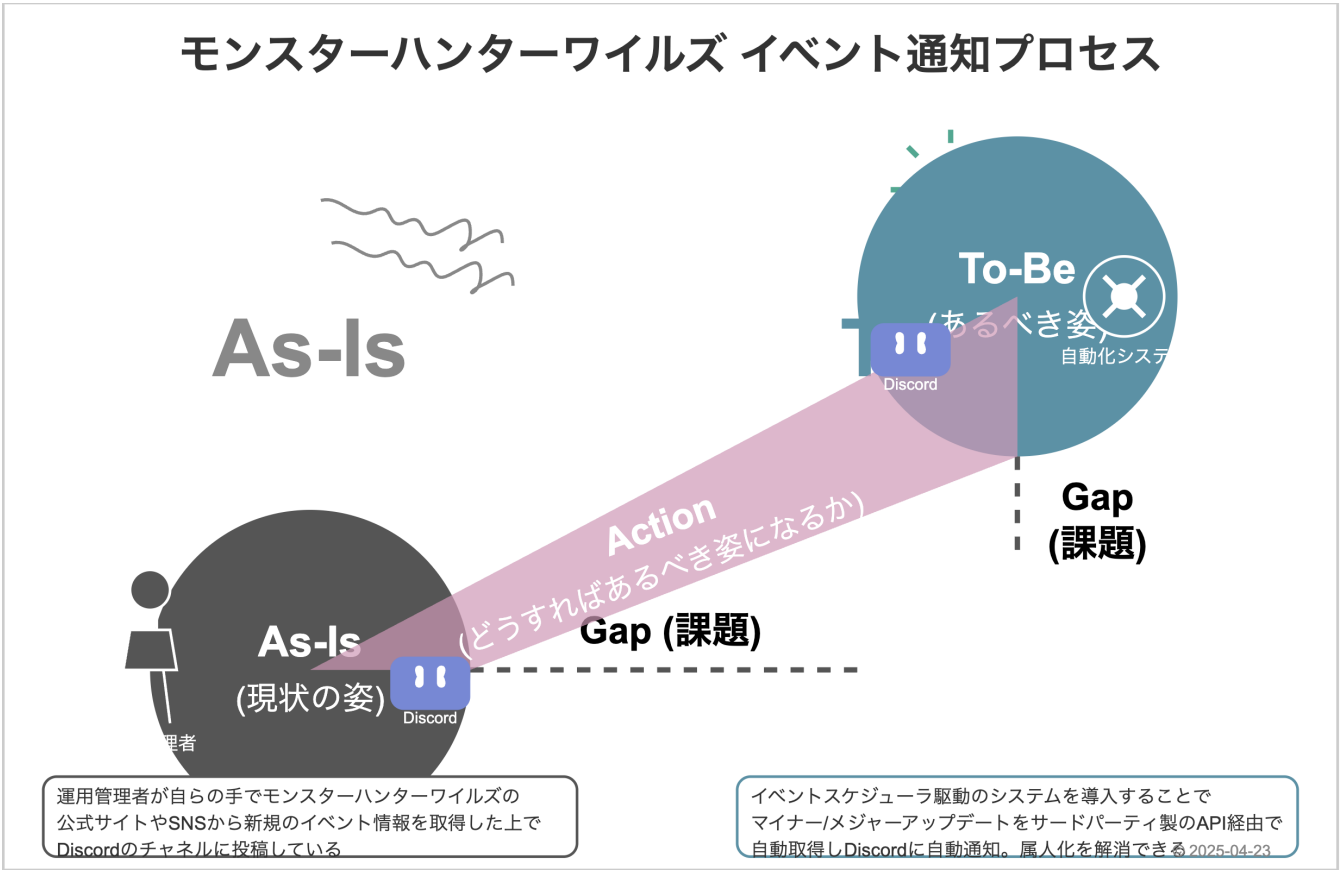
1.現状の課題とあるべき姿

1-1. 現状の課題

- 運用管理者が自らの手でSNSや公式サイトから情報を集約し、Discordのチャンネルに投稿している。

1-2. あるべき姿

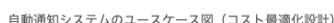
- Discordで通知するアクションをシステム経由で自動的に行える仕組みを提供し、運用管理者の作業工数を削減し属人化を防ぐことができる環境にする



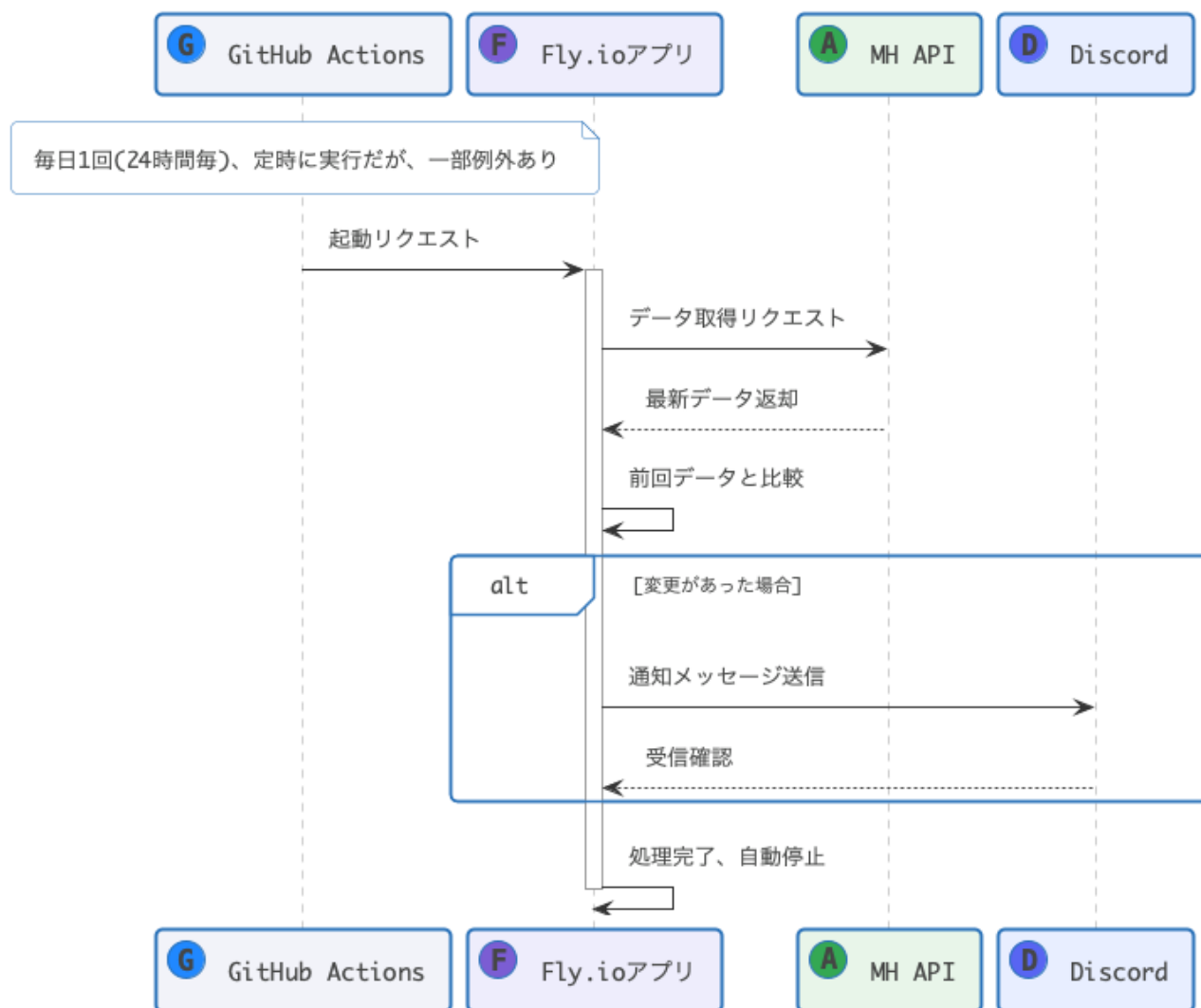
2.ユースケース

2-1. ユースケースとは?

- システムを利用する人が、システムを使ってある目的を達成するために行う一連の行動や操作をシナリオとして記述したもの



- Github Action
 - Discordアプリケーションを起動させるためのイベントスケジューラの役割を担います。
- Fly.io アプリ
 - Discordアプリケーション(情報が混在しないように簡略化してます。)
 - 詳しくの説明しますと、Fly.ioはdiscordアプリケーションを稼働させるためのアプリ基盤(PaaS)というものになります。
- MH API
 - インターネットで公開されている非公式のAPIです。APIはApplication Programming Interfaceを正式名称とし、役割としてはアプリケーションの一部の機能を外出しとすることで、外部のアプリケーションでも流用を可能とする働きがあるアプリケーション全般を示します。
- Discord
 - 皆様となじみのあるアプリなので割愛します。



モンスターハンターワイルズイベント通知システム シーケンス図

5. 運用コストについて

月間予測実行回数

前提として、モンスターハンターは水曜日～木曜日にかけてがアップデートのコアタイムとなることが想定されるので、この2日間に関してを「目標日前日」と定義し、上記に該当しない日を「通常日」と定める

- 通常日
 - $28(\text{日}) : 1(\text{実行}) \times 28(\text{日}) = 28(\text{回})$
- 目標日前日
 - $2(\text{日}) : 48(\text{実行}) \times 2(\text{日}) = 96(\text{回})$
 - 48回の実行になる理由
 - 水曜日～木曜日はコアタイムとなるため、0.5hごとにアプリケーションを実行する。そのため1hあたり2回の実行回数となる。ゆえに1日あたりに拡張した場合、1日は24hであるため、 $24 \times 2 = 48(\text{回})$ となるため

サービス利用量

上記の運用時間をベースにそれぞれのサービスで発生するコストを見積もる

Fly.io:

- 実行時間: $\text{約}124\text{回} \times 10\text{秒/回} = \text{約}20\text{分/月}$
- 無料枠: 500時間/月 結論: 無料枠内（使用率約0.07%）

GitHub Actions:

実行時間: $\text{約}124\text{回} \times 1\text{分/回} = \text{約}124\text{分/月}$ 無料枠: 2,000分/月 結論: 無料枠内（使用率約6.2%）

Discord

- webhookによるチャネル連携だけなので無償で利用可能

総コスト

月額: ¥0（完全無料） 年額: ¥0（完全無料）