

# インフラ運用ツール開発組織を 1から組織した話

LINEヤフー株式会社 Site Operation本部  
ネットワーク1部 ネットワークオペレーションチーム

酒井 亨 Sakai Toru

**LINEヤフー**

© LY Corporation



# Agenda

## 1 自己紹介

この人何者?

## 2 NW開発チーム発足のきっかけ

## 3 NW開発チームbefore/after

開発チームの前と、後

## 4 何から始めよう?

立ち上げ期に何を考えて何をしたか

## 5 チームの成長に伴う課題と解決

実体験から効果があったものを紹介

## 6 終わりに

4年間でふりかえって

## 7 質問&議論

# 1 自己紹介

## \* 自己紹介 \*

LINEヤフー株式会社Site Operation本部  
ネットワーク1部 ネットワークオペレーションチーム  
酒井 亨



## \* 経歴 \*

### ● Sler 2005-2014

新卒入社。ネットワークエンジニア。  
営業支援から設計、構築を担当。

### ● Web系サービス 2014-2019

ネットワークを中心に、サーバー管理、アプリケーション開発等幅広く担当。

### ● LINE株式会社(現LINEヤフー株式会社) 2019-

ネットワークインフラ組織のネットワーク以外全般。こぼれ球拾い。  
ネットワークエンジニアとアプリケーション開発者の繋ぎ。業務改善。

## **2 NW開発チーム発足のきっかけ**

# きっかけ

- ネットワークの規模が成長し、自動化の機運が高まった
- NOCが稼働し、誰でも作業ができるような運用の仕組みが必要になった

年	2017	2018	2019	2020	2021	2022	2023
稼働中NW機器	-	-	7,225	8,374	10,857	16,366	13,203
緊急アラート件数	142	208	435	872	2259	1152	834
イベント			CLOS NW 運用開始	NOC稼働 開発チーム発足			

## **3 NW開発チームbefore/after**

# Before: 運用体制

- 体制: 週輪番で社員が運用を全て担当(当番制)
  - 定常業務
  - 問い合わせ対応
  - 障害対応(夜間オンコール対応含む)
- 作業方法
  - 当番が手順書を作成して手動対応(一部、個人で開発したツールを利用して対応)

# After: 運用体制

- 体制
  - 定常業務: NOC
  - 問い合わせ対応: NOC
  - 障害対応: NOC
  - 日中エスカレーション: ネットワークオペレーションチーム
  - 夜間エスカレーション: 当番
- 作業方法
  - ツールを使って対応
  - マニュアル化された手順に従って作業

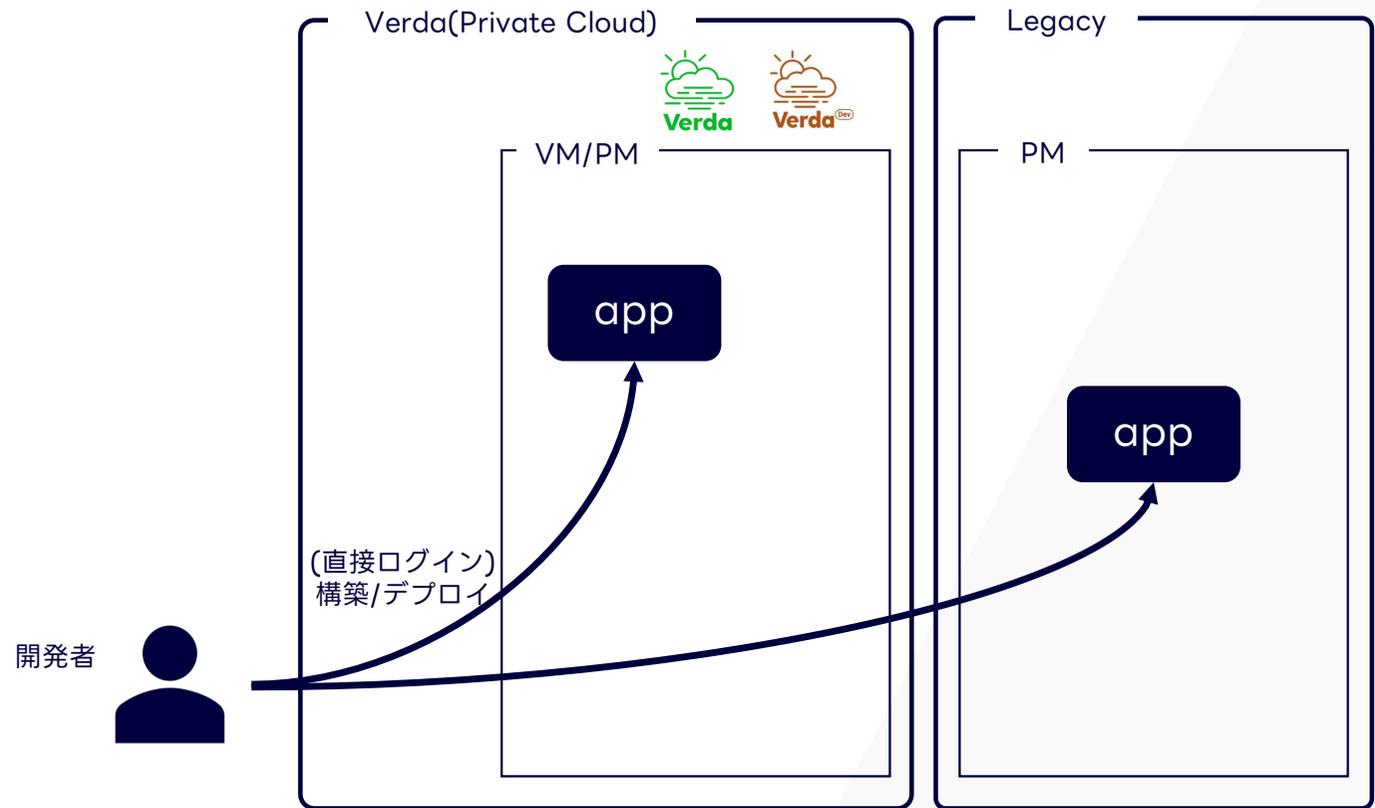
# Before: 開發環境

- Language
  - Frontend: JavaScript
  - Backend: Python
- Framework
  - Frontend: Bootstrap
  - Backend: Flask

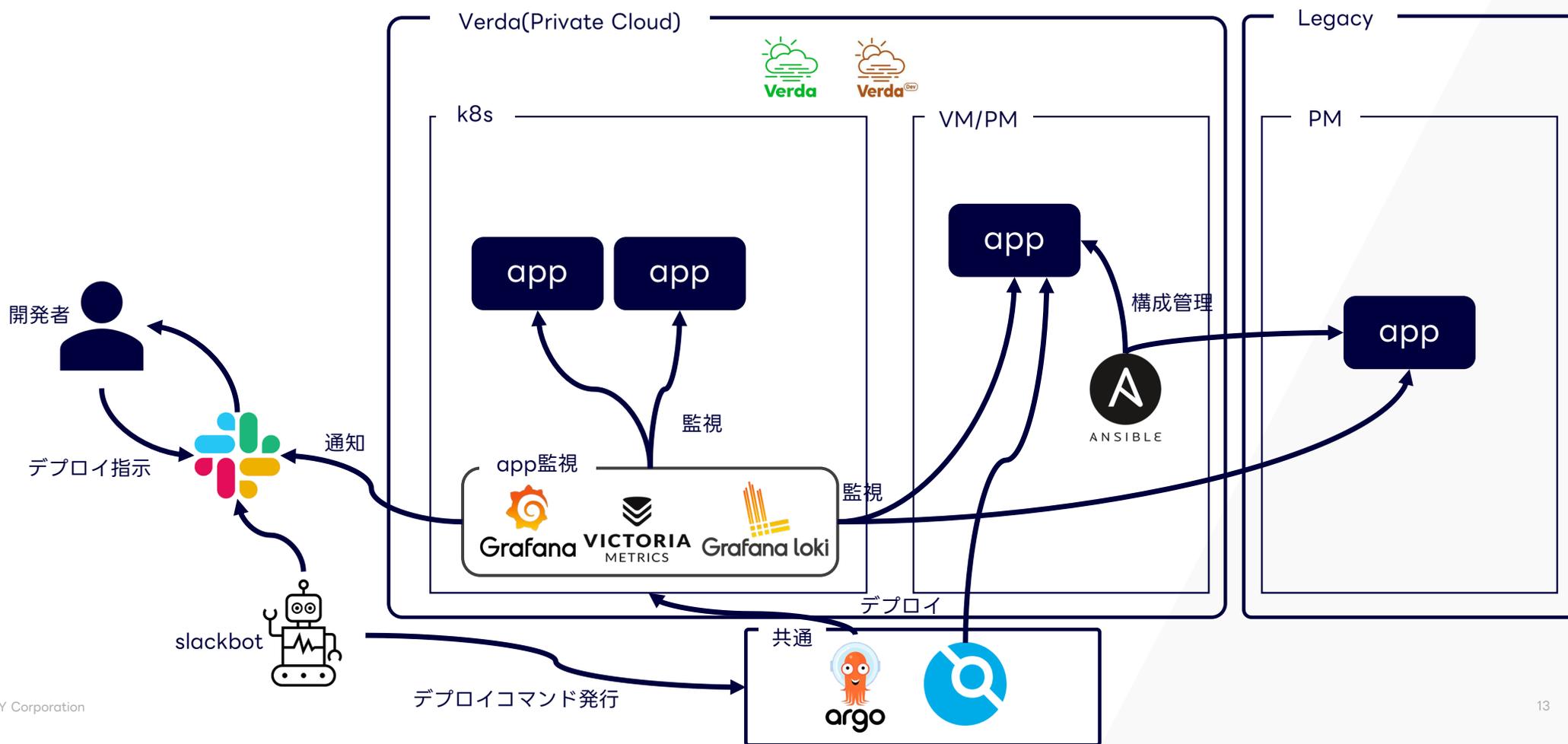
# After: 開発環境

- Language
  - Frontend: JavaScript, TypeScript
  - Backend: Python
- Framework
  - Frontend: React, Next.js
  - Backend: Flask, FastAPI
- Architecture
  - ドメイン駆動設計(DDD)?, レイヤードアーキテクチャ
- Configuration Management
  - Ansible, kustomize/helm(for k8s)
- CI/CD
  - DroneCI, ArgoCD(for k8s)
- IDE
  - VS Code (Dev containers, Copilot(はいいぞ!))

# Before: インフラ開発環境/CICD



# After: インフラ開発環境/CICD



## 4 何から始めよう？

# 成果を出し続ける仕組み作り

可視化と改善のサイクルを回す

## タスク管理(JIRA)

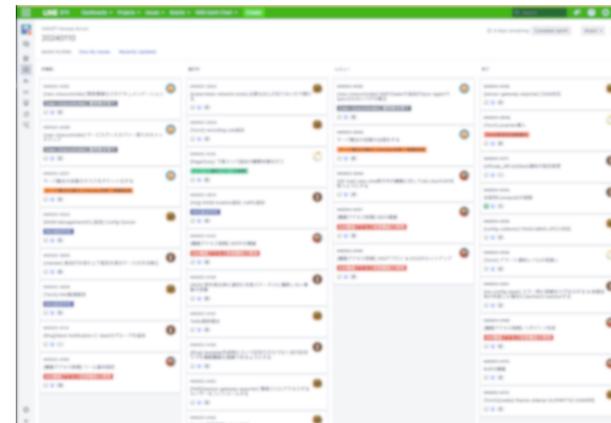
- タスクボード(カンバン)
- タスクの内容をちゃんと書く
  - タスクによって達成したいこと
  - 背景
  - 完了条件
  - 進行中の記録

## スクラム(?)開発

- スプリント: 2週間(水曜日->火曜日)
  - バックログ作成
  - 朝会
  - ふりかえり
  - レビュー(都度開催)

## 参考書籍

- カイゼン・ジャーニー (著)市谷 聡啓、新井 剛



# 成果を出し続ける仕組み作り

## ふりかえり

### グランドルール

- 全力を尽くしたことを疑わない
  - 原因の追求はしても責任の追求はしない

### KPT(?)フレームワーク

- Keep (続けること、うまくいったこと、できたこと)
  - いい仕事をした人への賞賛・感謝
- Problem (うまくいかなかったこと、もやもやしたこと)
  - 他のメンバーにも振って考えをシェアしてもらう
- Try (これからやること)
  - 具体的なアクションに落とし込まれているかを見る
    - がんばる、気をつける、意識する => 黄色信号
  - ルールは守られない => 可能な限り仕組み化を検討する

# 人員採用

どんな人を採用したか

## 派遣社員

- Python開発経験1年以上

## 正社員

- 開発/インフラ/ネットワークのいずれかに強みがあり、他の領域にも興味がある
  - ネットワークのスキル/経験等は必須ではない
- プロジェクトリードの経験
- 何か既存の仕組みを改善した経験
- 既存メンバーとやっていけそうか

# 人員採用

## 具体的なアクション

- ビズリーチ, LinkedIn等でのスカウト/月
  - 検索(200hit)
  - カジュアル面談依頼(20件)
  - カジュアル面談(2~4回)
  - 応募(1~2件)
- 自社採用イベントの登壇/年
  - 組織紹介
  - 業務内容紹介
  - ポジション紹介
  - 開発事例紹介

# 人員採用

NWのドメイン知識がないメンバーがNW開発組織でやっていけるの？

A: やっていただけます

## 1. 分担する

- NWのドメイン知識があるメンバーが設計を担当し、実装を行なってもらう
  - NW機器操作を伴う実装はNWエンジニアのレビュー必須
  - 設計側がボトルネックになる
    - ドメイン駆動設計を極めたら設計を分担できる...?

## 2. NWのドメイン知識が不要な業務を担当してもらう

- 開発組織自体の改善業務
  - コードの品質向上
  - CI/CD
- 共通的なツールの作成
  - 運用支援ツール(slackbot, エスカレーションツール)
  - NW監視基盤構築(監視条件設定以外はOSSの領域)

## 5 チームの成長に伴う課題と改善

# チームの成長に伴う課題

発足初期(2020~)

体制: 4名(社員1名)

## 課題

- レビューの負荷が高い
  - 細かい作法まで見てたらキリがない
  - コードを読むのに時間がかかる
  - 動作は開発者側で保証して欲しい
- テストの実行を忘れる

## 対応

- コーディングルール統一 linter導入 (flake8)
- 型(アノテーション)を書く (mypy)
- 単体テストを書く (pytest)
- CI(自動テスト)導入 (DroneCI)

# チームの成長に伴う課題

インフラつよつよSREがJoin(2021~)

体制: 5名(社員2名)

## 課題

- VMの管理がづらい
  - 設定ファイルが意図せず変更されている
  - VM(ツール)ごとにdeploy手順が微妙に違うのが大変
  - そもそもVMだと完全な構成管理は無理では…?

## 対応

- IaC導入
  - Ansible playbookで管理
  - Applicationのdeploy手順をCD化 (DroneCI)
  - k8s環境への移行

# チームの成長に伴う課題

現在(2022~2024)

体制: 5名(社員4名)

## 課題

- 他人の書いたコードを理解・改修するのがつらい
  - プリミティブ型(int, str, dict, list)は情報量が低い
    - e.g. dict, dict[str, list[dict]]
  - ディレクトリの構造、関数の責務が人によって異なる

## 対応

- データをちゃんと型として扱う(pydantic利用)
  - 実行時の型チェック
  - 型の詰め替えが便利
- バックエンドの設計手法/アーキテクチャとしてドメイン駆動設計(DDD)/レイヤードアーキテクチャを採用
  - 初見のコードでも目的の実装がどこにあるのか理解できる状態を目指す。
  - “ドメイン”の捉え方が人によって異なるため、誰がやっても同じとはならない(DDD難しい…。)

# 6 終わりに

# 終わりに

うまくいった要因をふりかえって

- 開発組織を作ったこと
  - ノウハウを蓄積する場ができた
  - ネットワークエンジニアに限らず、各分野に強みのある人材に採用の幅を広げられた
- ネットワークの運用と開発が同じチームに所属していること
  - 運用は、業務を定型化するので開発(自動化)と相性が良い
  - 運用のKPIを共有できる
- 改善し続ける仕組みを作り維持してきたこと
  - コミュニケーション機会創出
  - 失敗のハードルが下がった
  - 成長実感

# 7 質問&議論

# 質問&議論

## 議論

- みなさんの組織でうまくワークしている施策
- みなさんの組織における課題
- 特定のツールにおいては、開発者が最も詳しい=>その人に開発が集中する問題にどのように対処しているか
- 開発者にネットワークの知識を得てもらった事例

**ご清聴ありがとうございました**