

知識ほぼゼロでクラウド部配属、しかも運用部隊 ～ 自動化とAIで切り拓く、テレコムクラウド運用の現場 ～

若松 奏汰

楽天モバイル株式会社

クラウドプラットフォーム部 /

アプリケーションプラットフォームSREチーム

自己紹介

若松 奏汰

Sota Wakamatsu

所属	楽天モバイル株式会社 クラウドプラットフォーム部（入社二年目）
経験	クラウド運用 1年 / プログラミング 5年
入社前	大学でのプログラミング経験 ・ クラウドは未経験
主な業務	障害・トラブルチケット対応 / プラットフォーム監視・維持 / 運用自動化スクリプト開発

本セッションの概要

THESIS

新人クラウドオペレーターが、**AI と スクリプト**でどう運用業務を効率化したか。

01

クラウド運用業務の流れについて

02

AIでタスクの速度を上げた話

03

AI を使う上で学んだ教訓

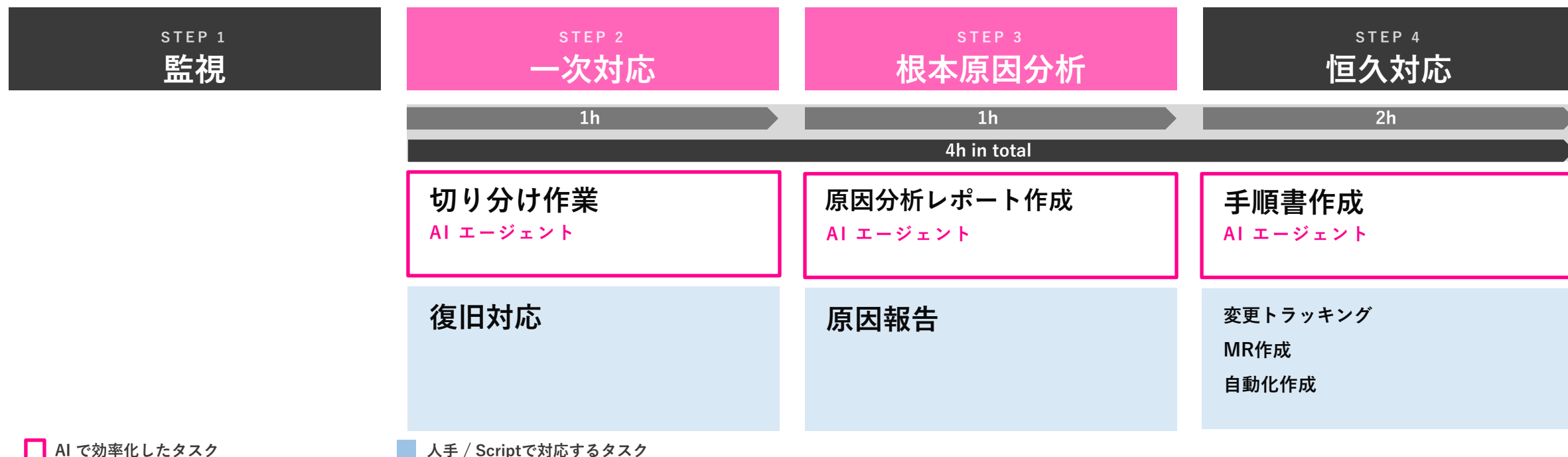
04

まとめ

本セッションのターゲット： 業務へのAIを活用を考えている若手エンジニア

クラウド運用業務 — どこを AI × Script で効率化したか

業務フロー



"切り分け → RCA → 手順書作成" の 3タスクをエージェント化。

運用業務でのAI活用 — 自作エージェント紹介

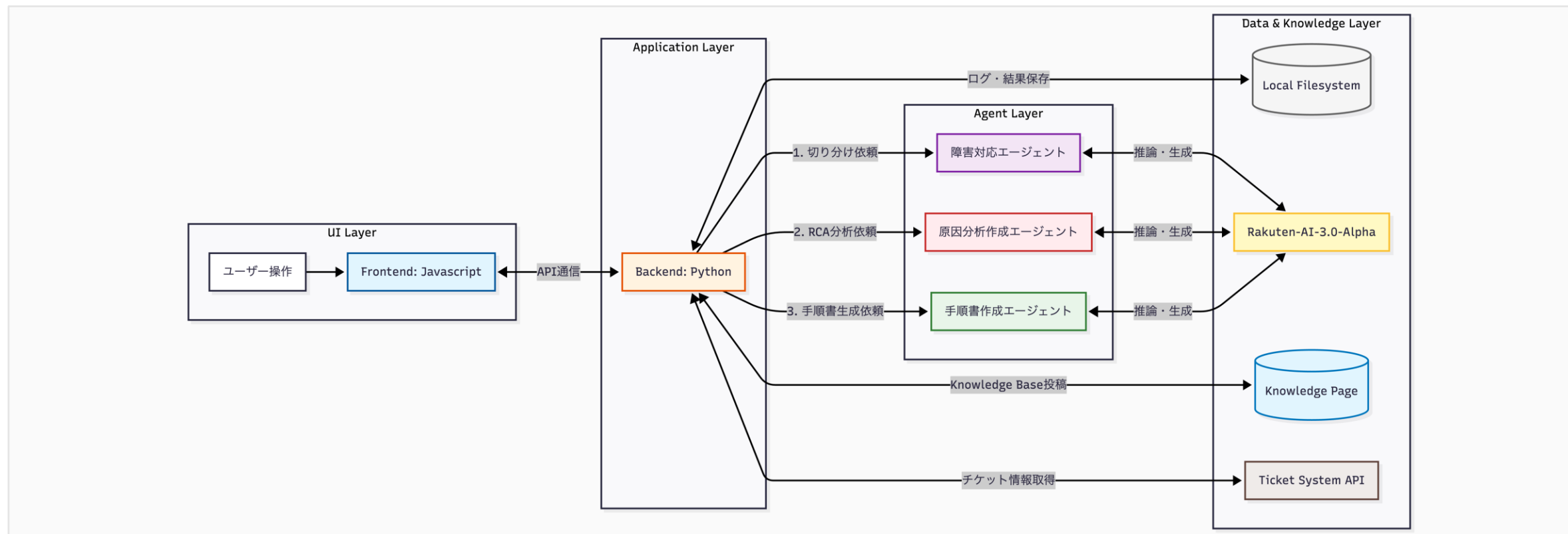
構築した3つのエージェント

① 障害対応サポート エージェント	② 原因分析(RCA)レポート作成エージェント	③ 手順書作成エージェント
ROLE 障害発生時の切り分け作業をリアルタイム支援 ▶ INPUT ・ 障害アラート内容 ・ 現在のクラスター状態 ▶ OUTPUT 原因仮説 + 確認コマンド	ROLE 原因分析(RCA)レポートの自動ドラフト作成 ▶ INPUT ・ 作業ログ ・ 原因分析メモ (ざっくりOK) + コマンド ▶ OUTPUT 所定フォーマットの 原因分析(RCA)レポート	ROLE 手順書を HTML 形式で自動生成 ▶ INPUT ・ コマンドと実行結果 ・ 作業の説明 ▶ OUTPUT HTML形式の手順書 (Confluence掲載)
PERSONA 熟練クラウドオペレーター	PERSONA レポート作成のプロ	PERSONA テクニカルライター

用途ごとにペルソナを定義し、出力フォーマットを固定する。

エージェント構成 — どう動いているか

ARCHITECTURE / 4層構成



レイヤー 1

UI

Frontend (JavaScript)

レイヤー 2

Application

Backend (Python) — タスク振り分け

レイヤー 3

Agent

RCA作成 / 手順書作成 エージェント

レイヤー 4

Data & Knowledge

Local FS · Rakuten-AI-3.0-Alpha
Confluence Page

操作デモ



ダッシュボード



チケット一覧



履歴



レポート



設定



チケット

XXXXXX-XXXX-XXXX

New

[Sample Ticket] Node NotReady Issue in test cluster

🕒 期限

2日 23時間 55分
(残り)

🚩 優先度

P4 - Minor



影響度
Minor



影響を受けているユーザー
なし



緊急度
Low



作成日時
2026/06/02 11:20

詳細

添付ファイル

履歴

コメント

チケット説明

Kubernetesクラスタ内のワーカーノード（worker-node-03）がNotReadyステータスに遷移しました。
該当ノード上のPodがEvictされ、サービスの一部でレイテンシが発生しています。
ノードの復旧および原因調査が必要です。

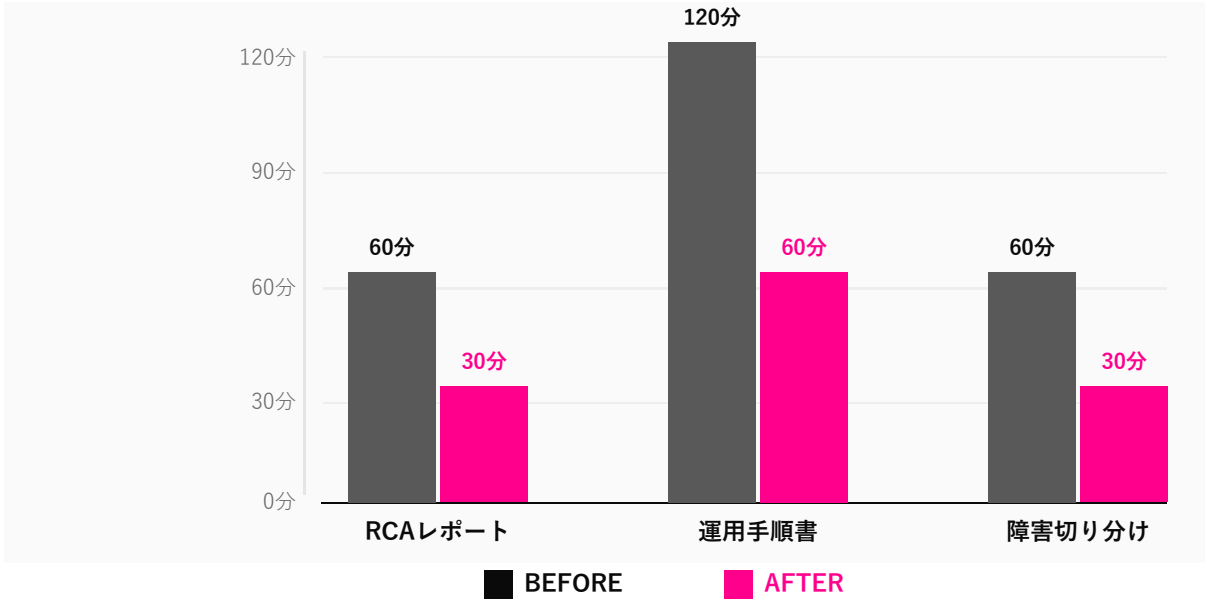
こちらは、私たちが普段使用している、
障害チケットを再現したダミー画面です

AI導入による削減効果

BEFORE → AFTER / 作業時間

タスク	BEFORE	AFTER	SAVED
原因分析レポート作成	60分	30分	▼ 30分
運用手順書作成	120分	60分	▼ 60分
障害切り分け作業	60分	30分	▼ 30分

各タスクの所要時間推移（最大値ベース）



月次削減効果

$(30分 + 60分 + 30分) \times 20営業日$
 $= 2,400分 / 月$

40時間 / 月

= 約 50%もの作業時間の削減

AI × Script を使ってみて

+ 良かった点

01

圧倒的な高速化

プロンプト作成・状況説明の手間が不要に。実作業までのリードタイムを劇的に短縮。

02

スキルの平準化

新人でもほかのメンバーと同等のタスクを遂行可能に。

03

高速な学習サイクル

業務と学習が一通貫で回せる。

- 改善点

01

AI 回答の信頼性

提案された解決策が必ずしも正解とは限らず、最終的な実行判断はエンジニアに委ねられる。

02

社内情報の『AI最適化』に伴う準備負荷

汎用 AI では独自仕様を理解できない。環境情報のインプットに膨大な準備時間が必要。

03

インプットの質が、アウトプットの質を決める

AI 回答精度はインプットするログ・指示の質に完全に依存。

AI は "万能の答え" ではなく、"優秀な共同作業者"。

結論 — 理想 と 現実

理想

AI が全工程を担い、運用が回る世界

検知 → 分析 → 対応 → 報告までを AI が担当、人手をほとんど借りずに運用が回る。

現実

100% 自動化は、まだ難しい

コマンド実行・Config編集はリスクが大きく、人手の介在が不可欠。熟練者の "感覚" も再現困難。

最適解

AI

推論・分析・文章生成

柔軟な思考が必要な領域を担う。
切り分け / RCA / 手順書 など。

SCRIPT

再現性可能な領域

確実性が必要な処理を担う。
定型作業・通知 など。

HUMAN

最終的な実行判断

責任を持って意思決定する。
三者の強み弱みを理解した棲み分けが鍵。

AI × Script × 人間 —— この三者の棲み分けが、運用におけるAI活用の最適解。

今後の展望 — Human-in-the-loop の高度化

VISION

「AI が自律的に提案し、人間がそれを管理・監督する」—— **Human-in-the-loop** の高度化。

NOW

人手で起動・人手でレビュー

異常検知 → 切り分け → RCA → 手順書生成まで、
各工程で人間が起動・確認・修正を行う。



NEXT

"半自動化" + "承認だけ" の人間操作

異常検知をトリガーとして、RCA・手順書生成までの一連のプロセスを半自動化、運用タスクのさらなる効率化・高速化を目指す。

人間の知見を取り込み続ける学習パイプライン

① AI が手順書を生成



② 人間がレビュー



③ フィードバック投入



④ AI 精度向上

ご清聴、ありがとうございました。