AI Agent による DC運用自動化の実装と課題

LINEヤフー株式会社

岡田 嘉

宮城 勝

LINEヤフー

Agenda

01. 我々が運用するインフラ
02. 自動化ツール“kurouto”の導入
03. AIエージェントを作ってみた話
04. AIエージェントとMCPサーバ
05. AIエージェントのディープな話
06. まとめ

自己紹介

岡田 嘉
Yoshimi Okada

ネットワーク本部 ネットワーク1部

経歴

- **ネットワーク装置ベンダーのSE**
- **携帯会社のテレコムエンジニア**
- **2023.01~ 旧ヤフー株式会社 入社**
  ネットワークエンジニアと言い張るが最近は
  Pythonばっかり書いている

JANOGはオンライン勢

趣味

クライミングと子育てと猫

宮城 勝
Miyagi Masaru

ネットワーク本部 ネットワーク1部

経歴

- **2024.04~ LINEヤフー株式会社 新卒入社**
  （現在2年目）

JANOG参加歴

- **JANOG51 @ オンライン**
- **JANOG52 @ 長崎（若者支援）**
- **JANOG53 @ 博多（Org, 若者支援）**
- **JANOG54 @ 奈良（現地）**
- **JANOG55 @ 京都（Org）**

趣味

- **筋トレ, 旅行, カメラ, テニス**

我々が運用するインフラ

我々が運用するインフラ

各社の運用機器台数

	LINE	ヤフー	LINEヤフー
NW機器台数	10,000+	12,000+	22,000+
Clos NW機器台数	5,500+	5,500+	11,000+
サーバ機器台数	70,000+	100,000+	170,000+

我々が運用するインフラ

今回のターゲット

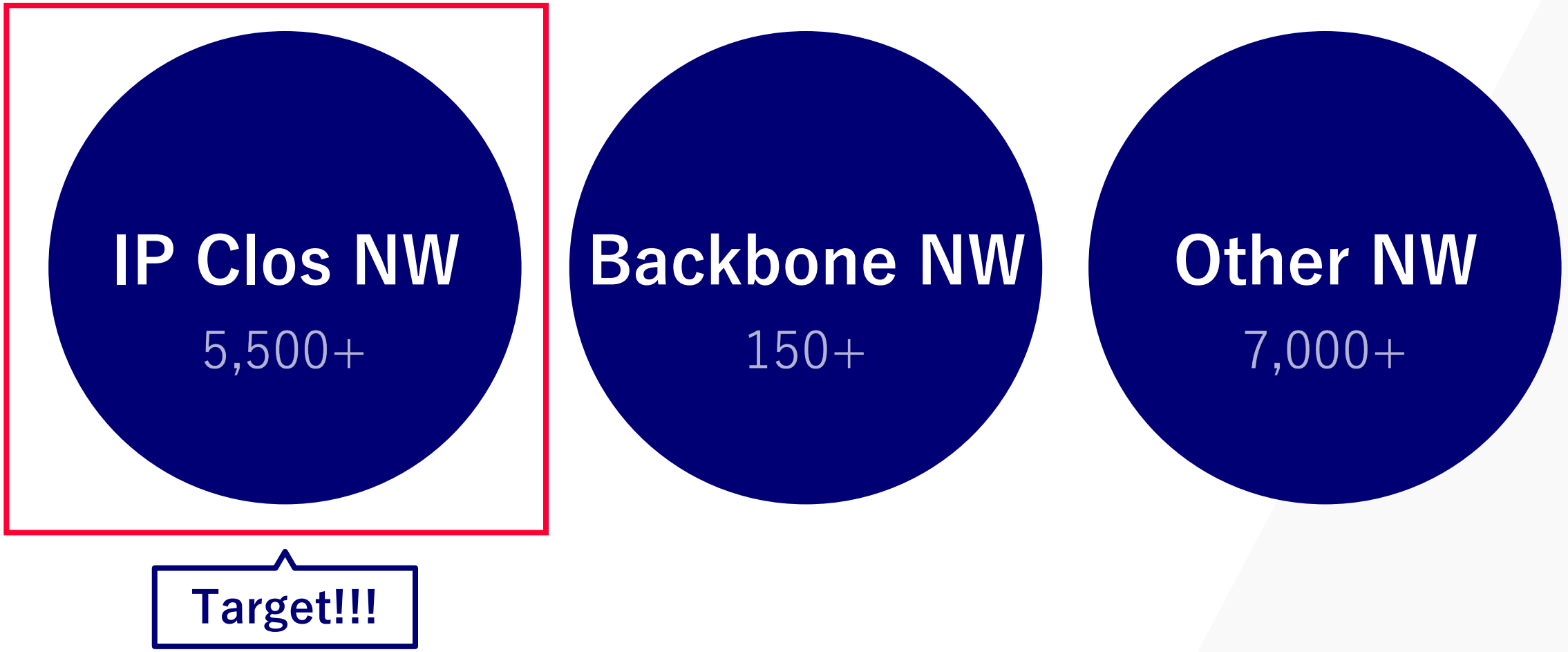
	LINE	ヤフー	LINEヤフー
NW機器台数	10,000+	12,000+	22,000+
Clos NW機器台数	5,500+	5,500+	11,000+
サーバ機器台数	70,000+	100,000+	170,000+

我々が運用するインフラ

ネットワークにもいろいろ種類がありまして

我々が運用するインフラ

今回のターゲットはこちら



我々が運用するインフラ

数字で見る運用台数

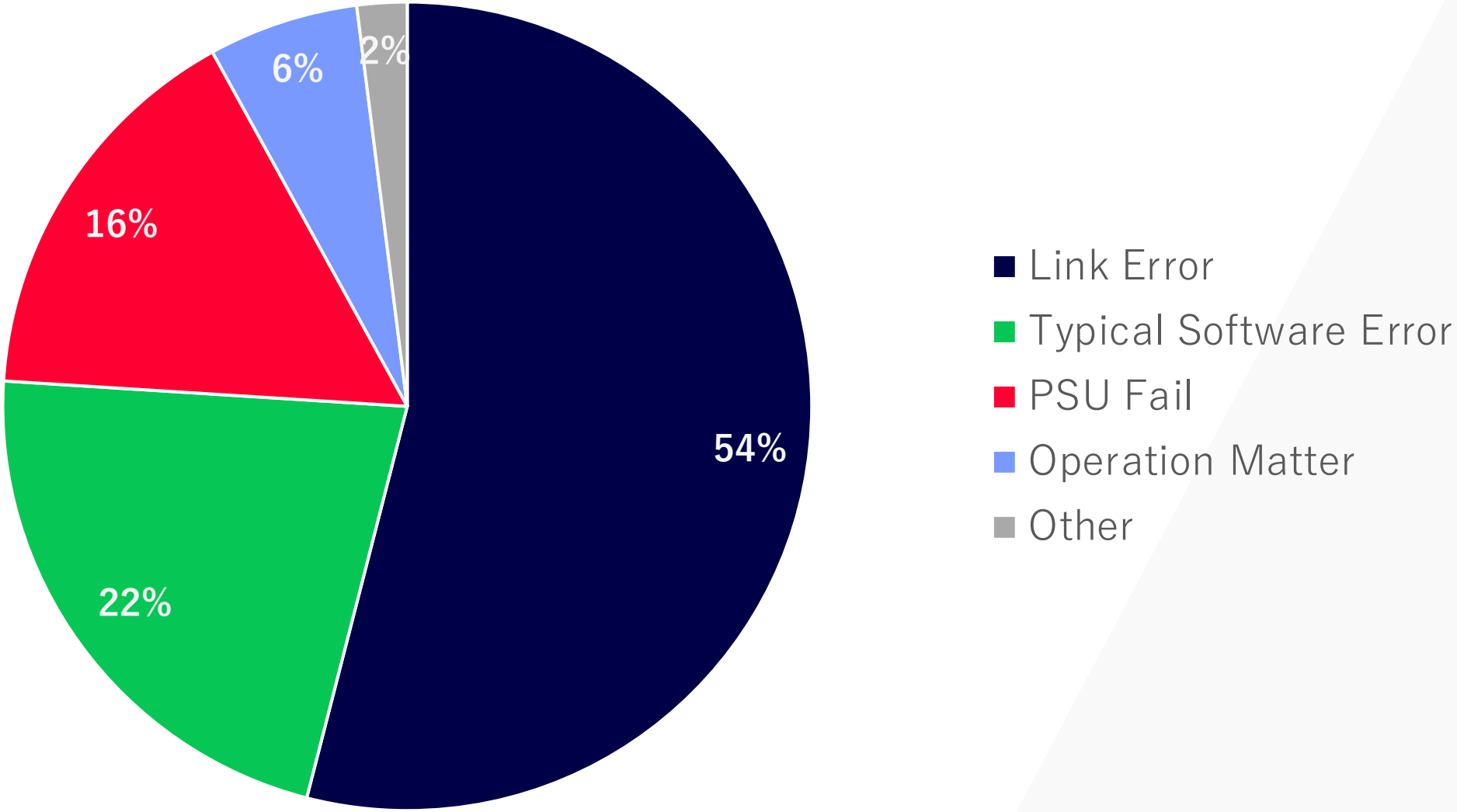
機器台数	運用中トランシーバ
5,500+	28,000+

OS	ベンダ	運用人数*
4	6	7

* アラート全てを主幹部署で対応

我々が運用するインフラ

ログ/アラートフィルタリング後の割合



これだけの台数を片手間で運用することはとても大変

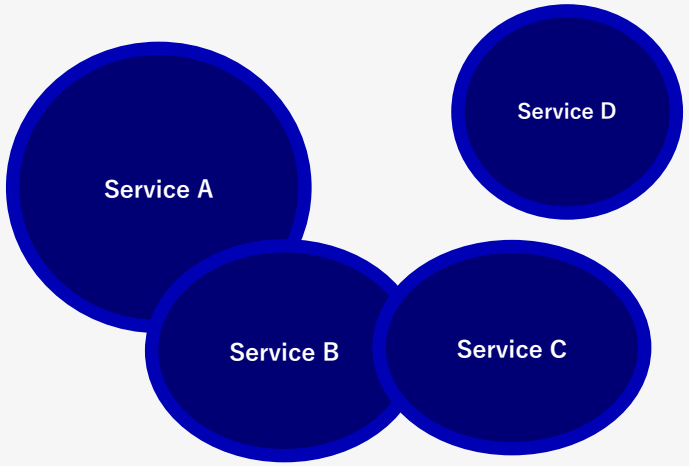
自動化ツール“kurouto”の導入

自動化ツール“kurouto”の導入

自動化開発の課題

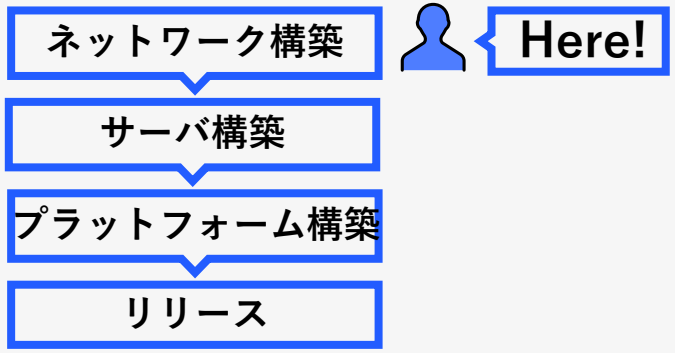
様々な環境が混在

- Clos構成の中でも
  サービスからの要求に
  より様々な環境が存在
- 環境ごとconfig/device/
  構成等..が異なる



構成が暫定的になる事も

- NWはDC内構築の一番手
- NWリリースが無いと
  後続の構築が難しく、
  納期問題などから暫定的
  な構成になりがち

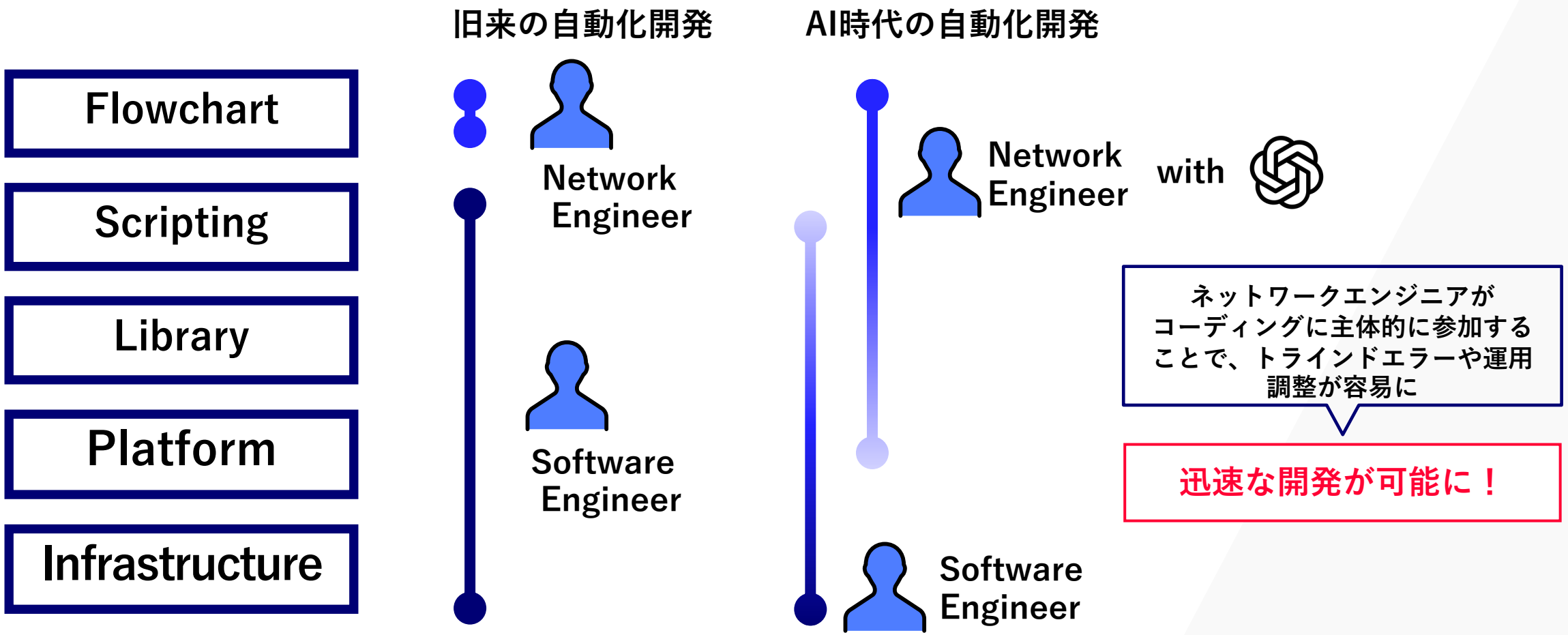


他チームとの連携多め

- DC物理作業：DC team
- 緊急架電： 監視team
- サーバリンクのメンテ：
  サーバーチーム

自動化ツール“kurouto”の導入

AI時代のkurouoto自動化開発



自動化ツール“kurouto”の導入

kuroutoとは

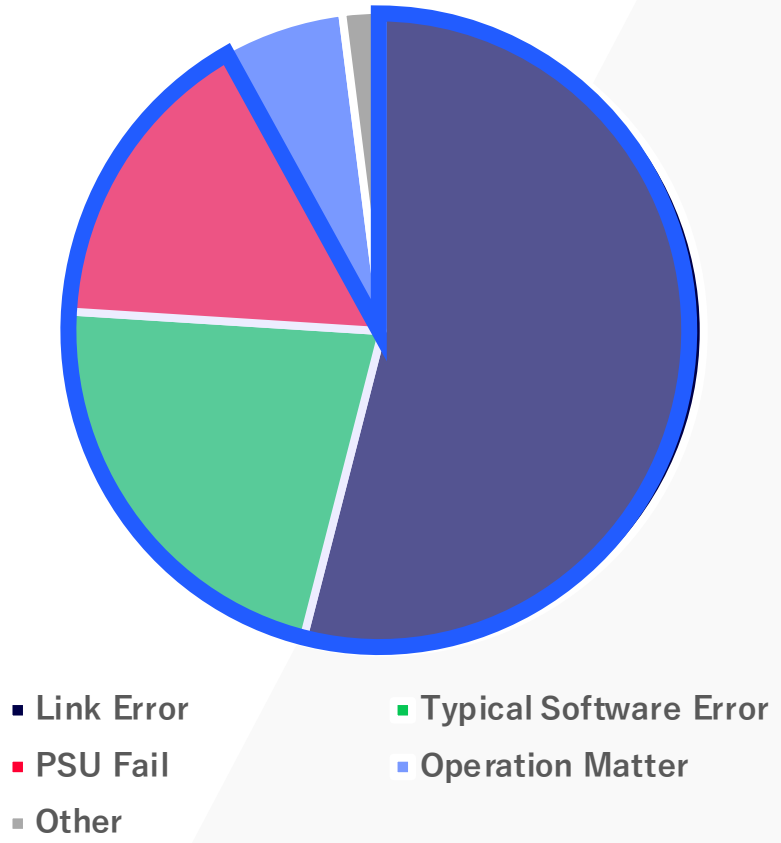
語源

- ClosNWの運用作業負荷低減を目標とした自動化ツール
- Clos + Automation = kurouto

約１年で５つのメジャーなアラートをサポート

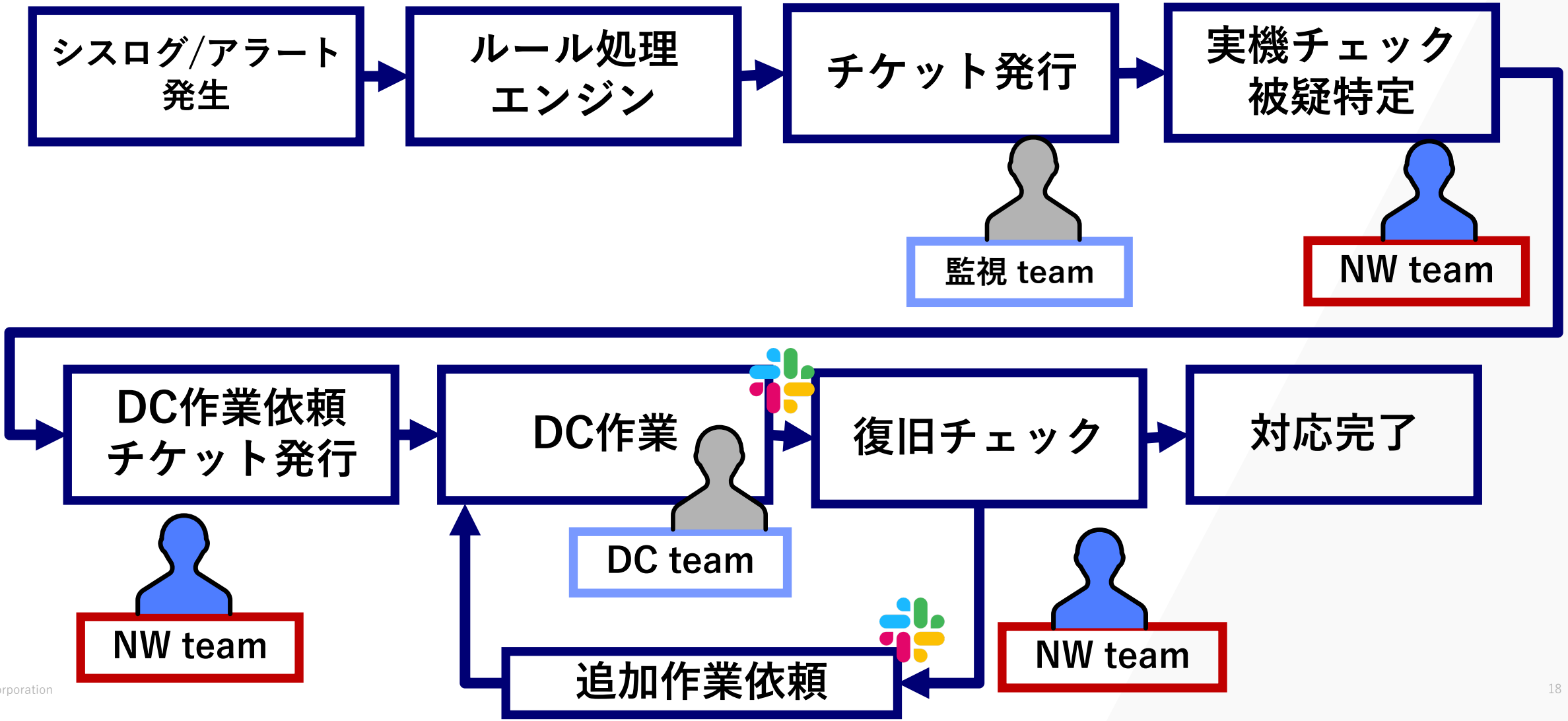
- Interface InErrors
- Link Down
- Software Error
- Disk Error
- PSU Fail

92%のアラートを自動化



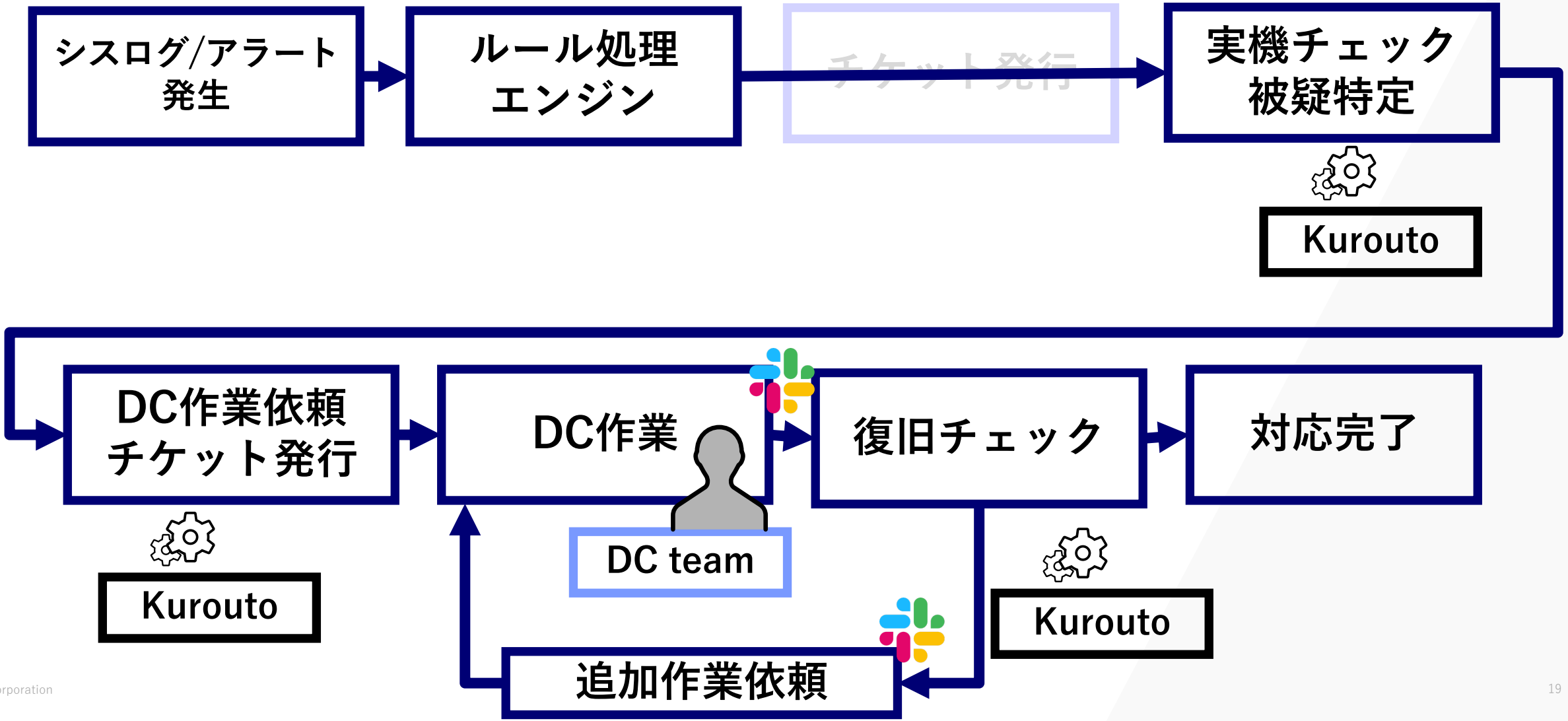
自動化ツール“kurouto”の導入

自動化前の対応フロー



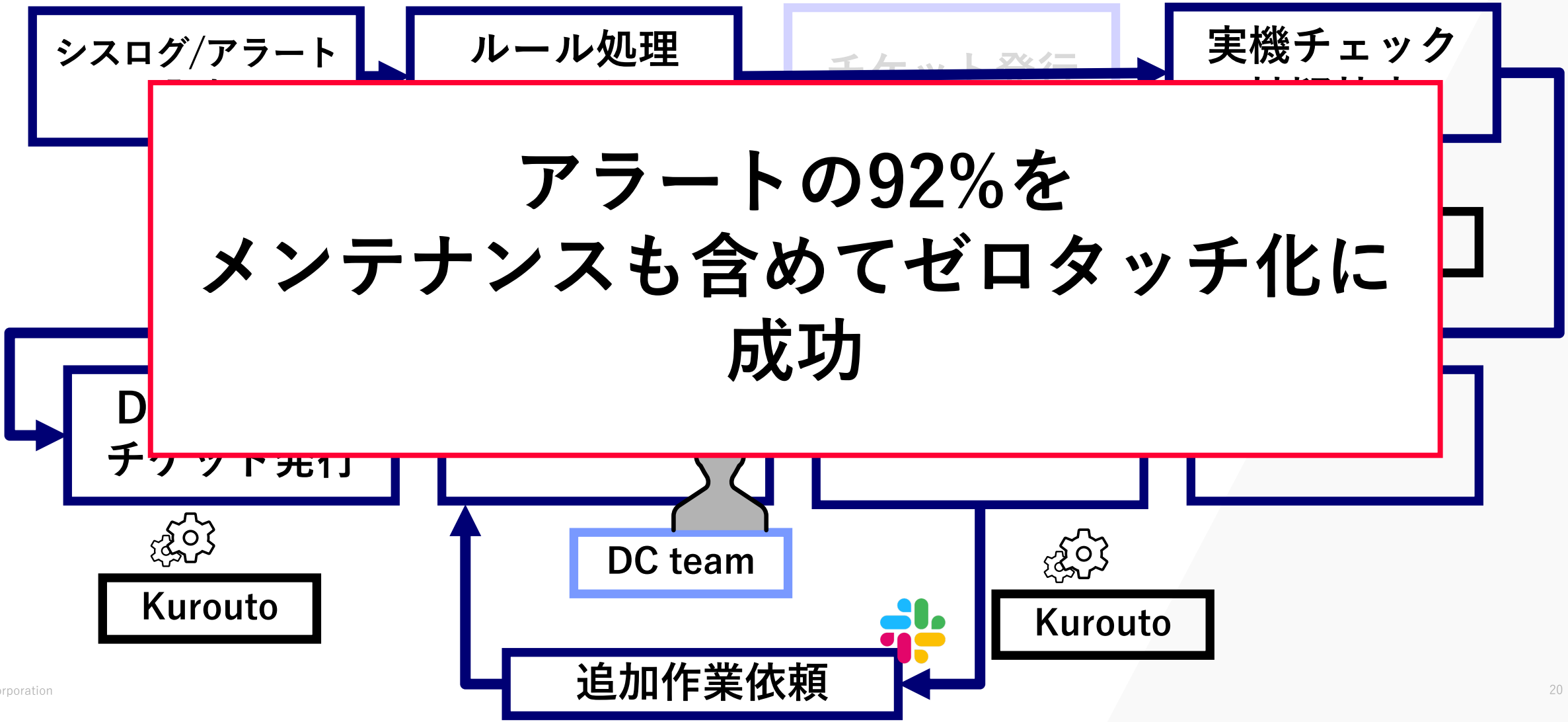
自動化ツール“kurouto”の導入

自動化後の対応フロー



自動化ツール“kurouto”の導入

自動化後の対応フロー



自動化ツール“kurouto”の導入

自動化前のアラート当番の1日

AM 10:00

前日夜分アラートチェック

AM 10:40

AM 11:03　新たなアラート発生！チェック！

AM 11:10　新たなアラート発生！チェック！

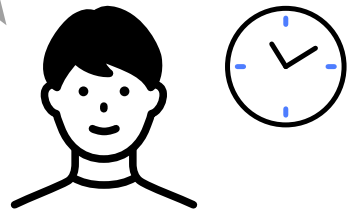
AM 11:30　新たなアラート発生！チェック！

AM 12:10　新たなアラート発生！チェック！

AM 12:30　新たなアラート発生！チェック！

PM 19:00

- 当番制：週一日（日中時間）
- アラート数：約80アラート/一日
- 業務開始時に前日夜分のアラートをチェック
- 通常業務との並列
  - チェックする間にも新たなアラートが発生
  - アラート発生時、通常業務に集中できない
- 筐体故障対応
  - 筐体物理故障時にはDCチームへ対応依頼
  - 交換後、config投入、リンク確認等..
- 対応だけで午前中/一日が終わってしまうことも多々有り



自動化ツール“kurouto”の導入

自動化導入後のアラート当番の1日

AM 10:00　前日夜分アラートチェック

AM 10:15

お仕事！お仕事！

AM 11:30　新たなアラート発生！チェック！

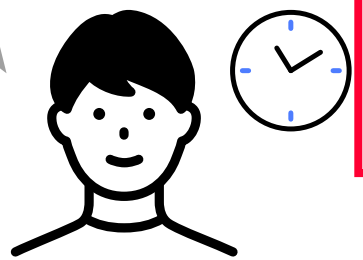
ご飯、お仕事、コーヒータイム！

AM 15:30　新たなアラート発生！チェック！

集中タイム

PM 19:00

- アラート数
  - 多い時：5アラート
  - 少ない時：0アラート
- アラート対応件数の変化
  - 導入前：一日数十件
  - 導入後：前日夜分のアラートチェックと日中にちらほら（ゼロの日も有り）
- アラートチェック/対応からの開放！
- 通常業務に集中できる時間が格段に増加！



とは言え残りの8%…
めんどくさいなあ

残りの8%…めんどくさいなあ

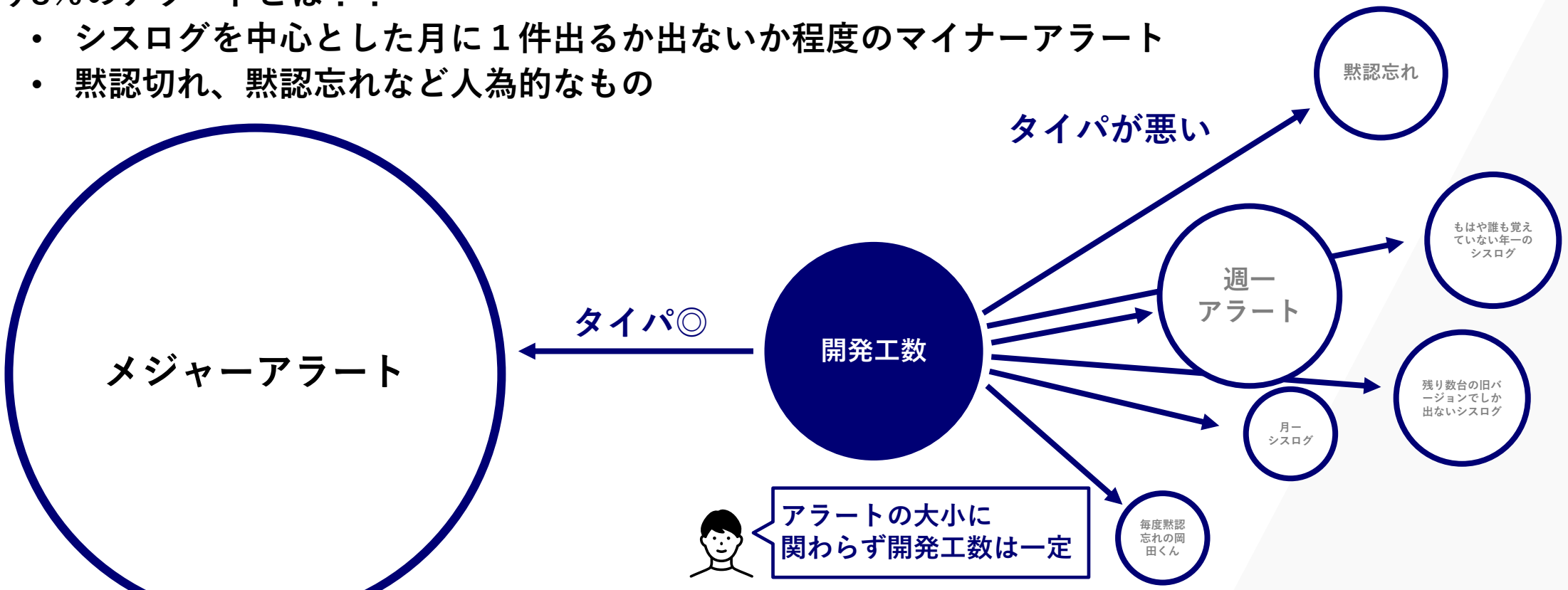
AIエージェントを作ってみた話

AIエージェントを作ってみた話

残り８%はタイパが悪い

残り8%のアラートとは？？

- シスログを中心とした月に１件出るか出ないか程度のマイナーアラート
- 黙認切れ、黙認忘れなど人為的なもの



そうだAIにやらせればいいじゃない！

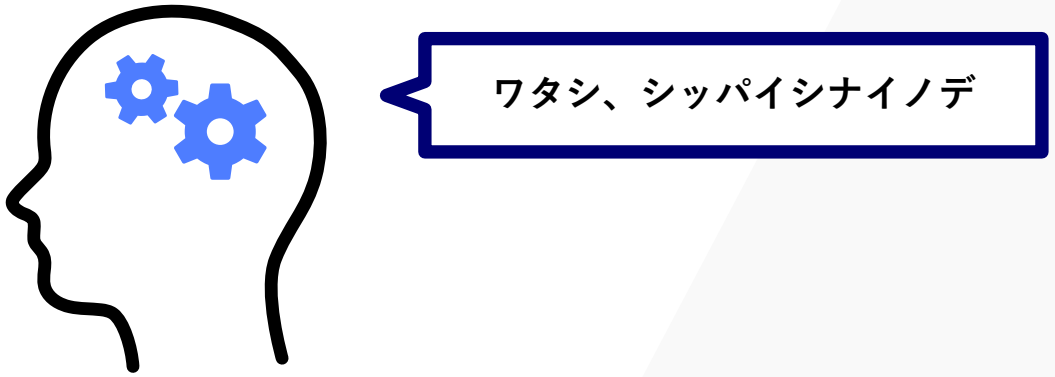
AIエージェントを作ってみた話

私たちが欲しいものは

前提

- AIの判断が正しいかどうか、人間が毎度確認するのはめんどくさい
- AIの誤判断で事故につながる事は絶対にあってはならない

我々が欲しいのは「絶対に間違わない、自立して動くAIエージェント」
そして人間は極力なにもしたくない！！

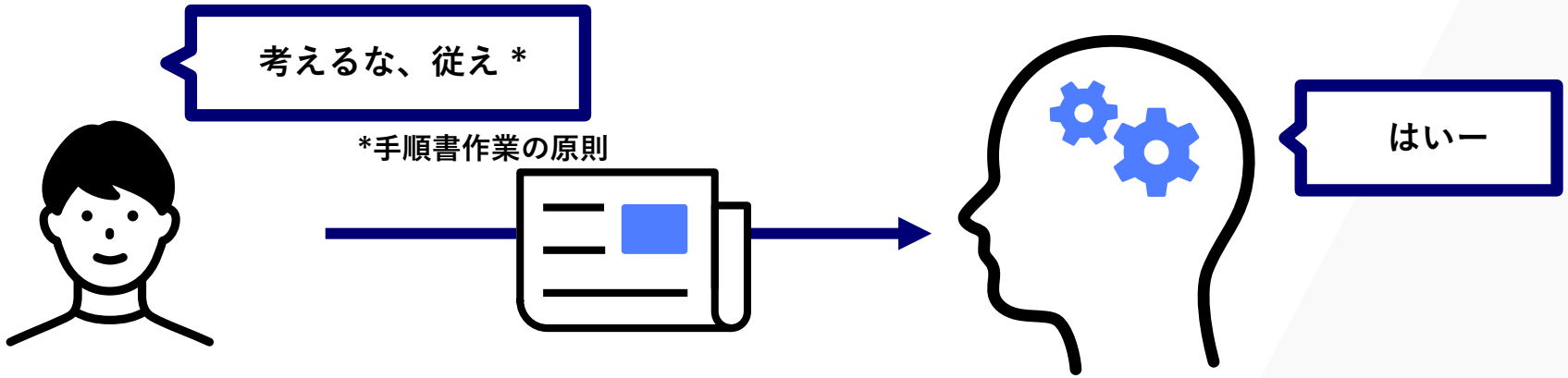


AIエージェントを作ってみた話

僕たちは手順書作りのプロである

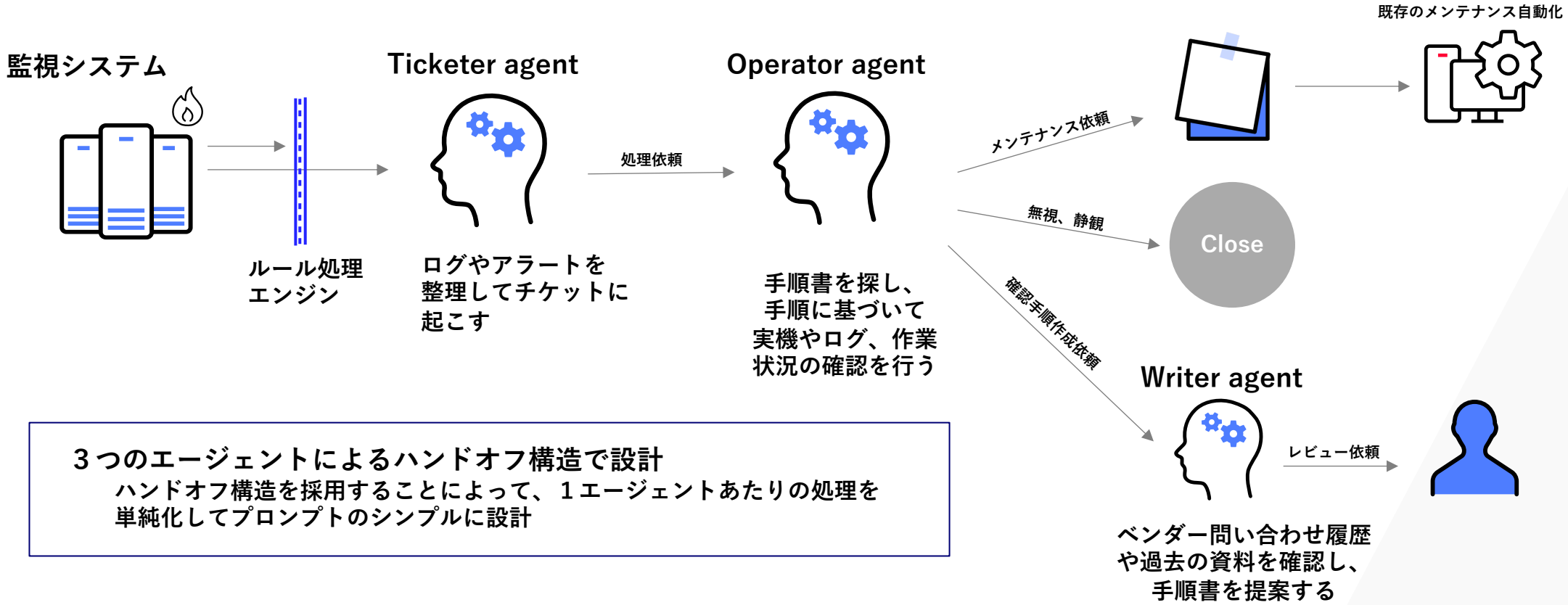
手順書とNWエンジニアの親和性

- NWエンジニアとしてはプログラミングは得意ではないが、手順書作成は得意！
- 手順書はロジックの塊、人間をプログラミングしているようなもの
- AIは自然言語を解釈でき、手順書の作業を任せることは親和性があるのでは？
- 手順書は認知ミスを防ぐノウハウが詰まっていて、AIの動きを拘束できるのではないか



AIエージェントを作ってみた話

３つのエージェントによるハンドオフ構造



３つのエージェントによるハンドオフ構造で設計

ハンドオフ構造を採用することによって、１エージェントあたりの処理を単純化してプロンプトのシンプルに設計

人間は手順書のレビューと修正をすればいい

AIエージェントを作ってみた話

動かしてみた結果

7月某日

19件のログ→7件のチケットを発行
5件のチケットを自動でクローズ
1件のチケットでメンテナンス依頼を実施
1件のログを手順書化しレビュー依頼

AIエージェントとMCPサーバ

AIエージェントとMCPサーバ

AIエージェントとは

AIエージェントって何？

```
agent = Agent(
    name="network agent ticketer",
    instructions=instructions,
    model=OpenAIChatCompletionsModel(
        model="gpt-4.1",
        openai_client=client,
    ),
    mcp_servers=mcp_servers,
)
```

思考の手段としては普通のチャットモデル

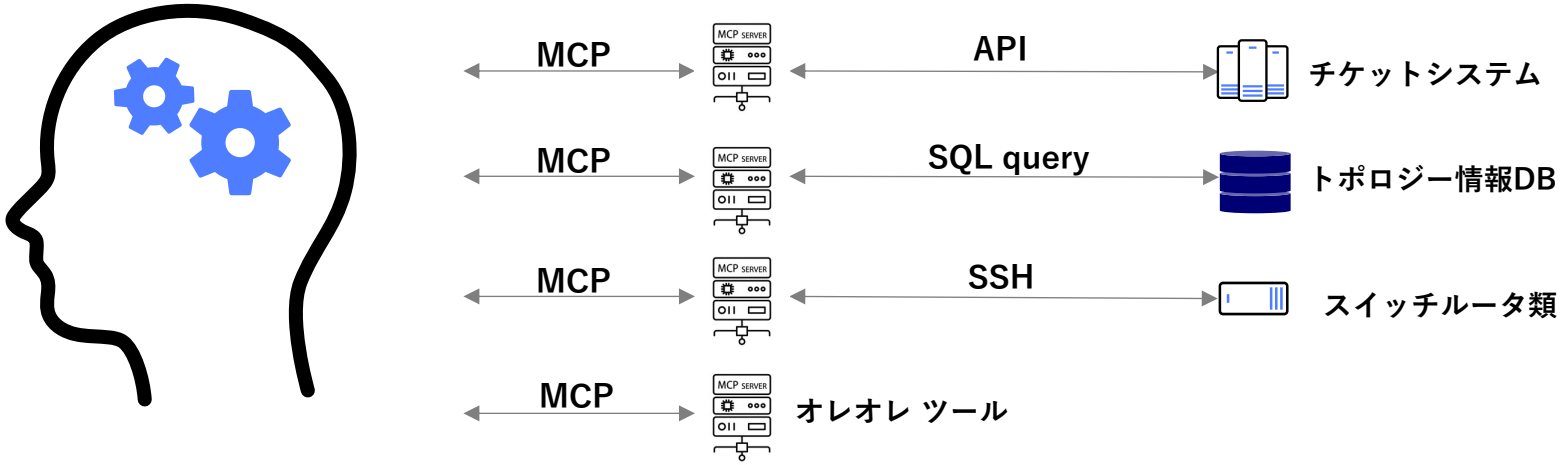
AIエージェントとは
　自問自答とアクションを繰り返すチャットAI


```
{
  "role": "user",
  "content": "jira_ticket_id: *******\nを処理してください\n"
},
{
  "role": "assistant",
  "tool_calls": [
    {
      "id": "call_SBHmKxicnHvsjyzaVM8Vr6rS",
      "type": "function",
      "function": {
        "name": "get_jira_logalert_ticket_by_id",
        "arguments": "{\"jira_ticket_id\":\"*******\"}"
      }
    }
  ]
},
{
  "role": "tool",
  "tool_call_id": "call_SBHmKxicnHvsjyzaVM8Vr6rS",
  "content": "{\\\"task_details\\\": *******}"
},
{
  "role": "assistant",
  "tool_calls": [
    {
      "id": "call_stkpOIw86Y2UR3brJWZOlsOU",
      "type": "function",
      "function": {
        "name": "device_name_formalyzer",
        "arguments": "{\"provide_name\":\"*******\"}"
      }
    }
  ]
},
{
  "role": "tool",
  "tool_call_id": "call_stkpOIw86Y2UR3brJWZOlsOU",
  "content": "{\"type\":\"text\",\"text\":\"{\\n  \\\"result\\\": \\\"success\\\",\\n  \\\"message\\\": \\\"Formalized device name: *******\\\",\\n }"
},
{
  "role": "assistant",
  "tool_calls": [
    {
      "id": "call_quHbM5hMwrqNCSmR3KFUdpaN",
      "type": "function",
      "function": {
        "name": "search_analystic_histry",
        "arguments": "{\"mySQL_regex_string\":\"Failed to read the snmp agent state with error Connection to management daemon lost\"}"
      }
    }
  ]
},
```

AIエージェントの構造

MCPサーバーって何？

MCP（Model Context Protocol）サーバー
　MCP＝AIエージェントが外部のサーバーとの通信に使えるプロトコル
　このプロトコルでAIエージェントにAPIやDB、装置のリモート接続やツール類を開示してあげるサーバーと接続することで、
　AI agentが実環境と接続される。



MCPサーバーと接続することでAIエージェントは現実に作用できる
“エージェント”としての振る舞いができるようになる。

```
{
  "role": "assistant",
  "tool_calls": [
    {
      "id": "call_SBHmKxicnHvsjyzaVM8Vr6rS",
      "type": "function",
      "function": {
        "name": "get_jira_logalert_ticket_by_id",
        "arguments": "{\"jira_ticket_id\":\"*******\"}"
      }
    }
  ]
},
{
  "role": "tool",
  "tool_call_id": "call_SBHmKxicnHvsjyzaVM8Vr6rS",
  "content": "{\\\"task_details\\\": *******}"
},
```

AIエージェントの構造

オレオレMCPサーバーを以下にカジュアルに量産できるか

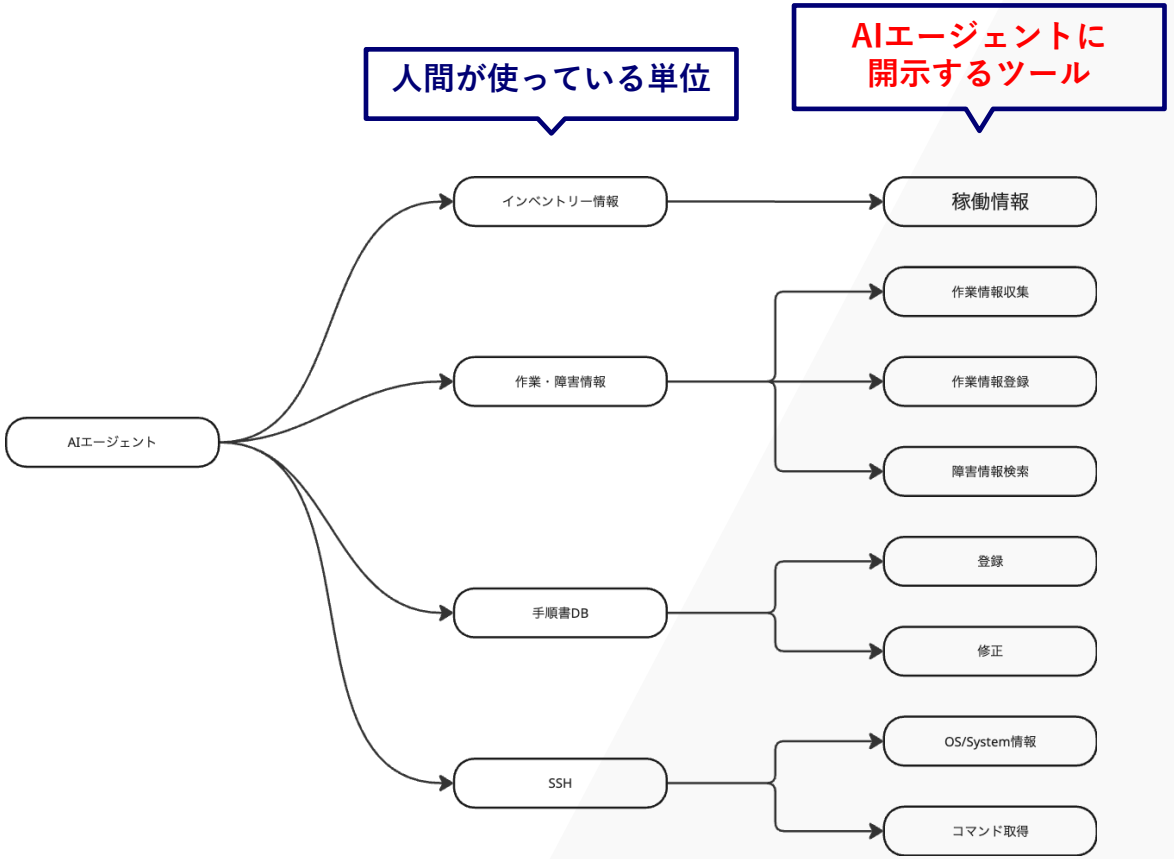
普段人がおこなってることをAIにやらせたい！

人が使っているツールをAIに全て渡さないと、、、

ざっと30-40のツールが必要そう、、

Fast MCPと既存の自動化のライブラリを流用MCPを
軽量に量産できる体制を作り、2-3個/日で量産

最終的に38ツール開発、6ツール開発中



AIエージェントの構造

ちょっと寄り道汎用MCPサーバーとの付き合い方

汎用MCPサーバーとは…
サーバー上にインストールするだけで稼働する各種MCPサーバー

ベンダーやOSSとして多数公開されています。

実際につかってみると。。。
- セキュリティが未実装
- 少し込み入った要求がくるとお手上げ
- 汎用すぎてプロンプトで指示することが多い、、、

といった問題があり、現状ではオレオレMCPサーバーを作った方が手っ取り早い

AIエージェントの構造

AIエージェントを作ることはMCPサーバーを作ること

実際の開発規模は…

蓋をあけてみるとAIエージェントの開発は、

MCPサーバーの開発でした

AIエージェントのディープな話

AIエージェントのディープな話

手順書でAIの動きはどれくらい変わるか

手順書処理AIエージェント 人がレビューした手順をもとに対応	過去履歴対応AIエージェント 過去の問い合わせ履歴をもとに対応	LLMネイティブエージェント LLMとしての知識のみで対応
想定通りに対応している：196件	想定通りに対応している：102件	想定通りに対応している：14件
対応しきれていない：4件 ＋Token overによるスタック3件 ＋コマンドが許可されていない1件	対応しきれていない：98件 ＋Token overによるスタック3件 ＋コマンドが許可されていない1件 ＋ベンダー履歴なし32件 ＋サイドエフェクトの取違い62件	対応しきれていない：186件 ＋Token overによるスタック1件 ＋ちょっと何言ってるの他。。。
致命的な判断ミス：0件	致命的な判断ミス：4件	致命的な判断ミス：30件超

さらにアップデート予定

AIエージェントのディープな話

実際の手順書は割とバイブな感じ

AIエージェントが自動作成した手順

タイトル:
＊＊＊＊の確認手順
チェック手順:
1. ログが単発かどうかを確認する。
   - 1回のみの発生で継続的に出力されていないことを確認
2. 直近で同様のエラーが連続発生していないことを過去ログで確認する。
   - 複数回発生している場合は切り分けまたはベンダー問い合わせ
3. サービスへの影響が出ていないかを確認する。
   - リンクダウンやエラー発生などの付随事象を確認
判断基準:
- 単発で本ログのみ、サービス影響もない場合：継続監視（特に対応不要）
- 頻発または他の異常（リンクダウン等）と併発の場合：ベンダーへエスカレーションまたは詳細調査
終了条件:
- 単発・影響なしで問題なければ「引き続き監視」としてクローズ
- 異常や頻発の場合は「エスカレーション（ベンダ問合せ）」

悪くはないんだけど、、

人間が修正した手順

タイトル:
＊＊＊＊の確認手順
チェック手順:
1.他の異常（リンクダウン等）と併発の場合
　緊急連絡を行う

2.同様のログが過去１週間で単発かどうかを確認する。
　- 複数回ある場合
　　＋ハードリセットの履歴が1月以内にあればRMAを依頼。
　　＋なければハードリセットの依頼を実施

3.正常性チェックツールを使い正常性確認を行う

修正は手間だけど、バイブな感じで
やってくれるのでとても楽

AIエージェントのディープな話

でもお高いんでしょー

あんまり詳細には言えませんが

7月某日

処理数：7チケット40ログの処理を実施

利用モデル：gpt-4.1（性能いいけどとてもお高い！！！）

$1行きませんでした！

GPT-4.1

複雑なタスクに最適なスマートモデル

料金

入力：
$2.00 / 100万トークン

キャッシュされた入力：
$0.50 / 100万トークン

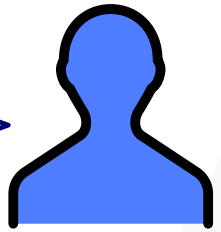
出力：
$8.00 / 100万トークン

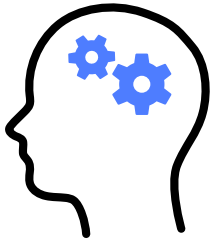
TokenはAIが認識する分割単位

AIエージェントのディープな話

でも、、、ちょっと信用ならないなー

ログを分類しjiraチケットを発行してください
その後、そのjiraチケットを手順書を元に処理し、
クローズしてください





確認したところ、これらのログはいずれも静観して
クローズに相当するものでした。
よって発行が必要なチケットはありませんでした。

チケット発行。。。

開発初期ではこのように指示にきちんとしたがってくれない問題に直面しました。
この時は、手順にステップ番号を振る。プロンプトを調整する×１０、フローチャートにして渡す。英語にしてみるなどいろいろ試しましたが、改善しませんでした。。
1エージェント ＝ 1役に限定し、プロンプトをコンパクトにすると起きなくなった。
後日既存のLLMはプロンプトが長すぎると精度が格段に悪くなるらしいとの情報を知る。。

AI agent特有のノウハウがあり、それをフォローできていれば精度はいい
でもキャッチアップが大変。。。

まとめ

まとめ

- AIエージェントによるアラート処理はできそう！JANOG明けから本格運用始めます。
- ネットワーク界隈の手順書文化との親和性が高い
- 自動化を自然言語（手順書）で設計、監査できると、ネットワークエンジニアのリソースが活用できる
- AIエージェントはわりと優秀めな新卒一年目ぐらいの能力はある
- AIエージェント特有のノウハウのキャッチアップが大変、、、
- 最後の難所はベンダー問い合わせ。ベンダーさんカスタマーサポートMCP一緒に検証しませんか？

お話きかせてください！

- AI活用すすんでますか？どうやって進めてますか？
- こんな感じのAIエージェントみなさんの会社でも使えますか？
- MCPサーバーとどうつきあってますか？
- AIによる自動化で困ってること、困難なことおしえてください！
- AI時代のキャリアプラン、、どうしましょう。