

# Janog49 LT

## NW運用自動化に挑んでみた話

2022年 1月 26日 Janog49

NEC ネットエスアイ 小沼高啓

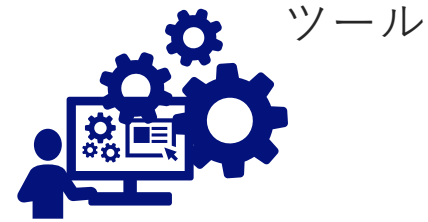
# ネットワークの運用自動化 どう実現しますか？

## ◆ ネットワークの運用自動化の手段: サービス、ツールが多数存在



ツール、サービスは充実…

選択肢が豊富…



ツール



ベンダ様のソリューション



ライブラリ

# ネットワークの運用自動化 どう実現しますか？

◆ ネットワークの運用自動化の手段: サービス、ツールが多数存在



敢えて “NW運用自動化ツール” を開発した話

”



# ネットワーク運用自動化に挑んでみた話

- ◆ ツール開発のモチベーション

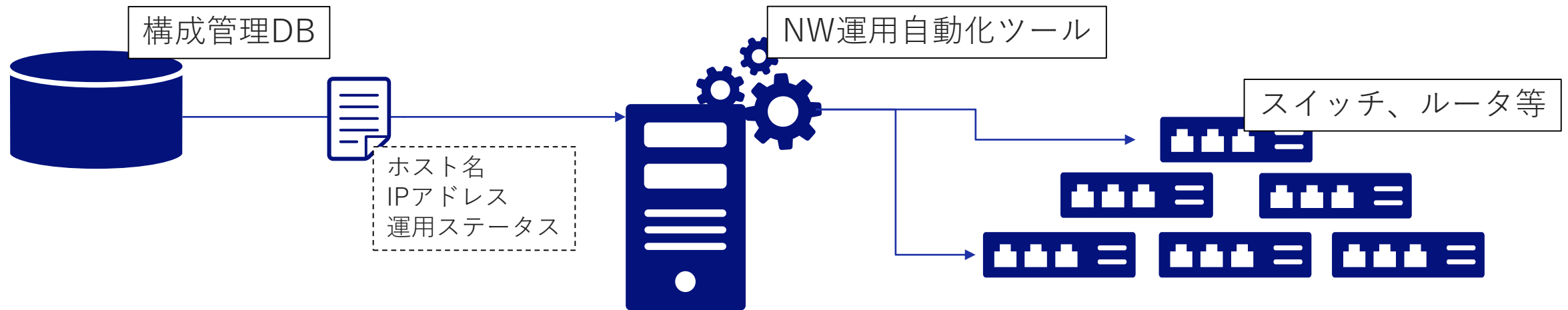
- ◆ 開発して良かった話

ツール開発のモチベーション：“柔軟な処理”

# ツール開発のモチベーション：“柔軟な処理”

## ◆ 例1：構成管理データベースとの連携

- ホスト名、IPアドレス、運用ステータス等はデータベースで一元管理

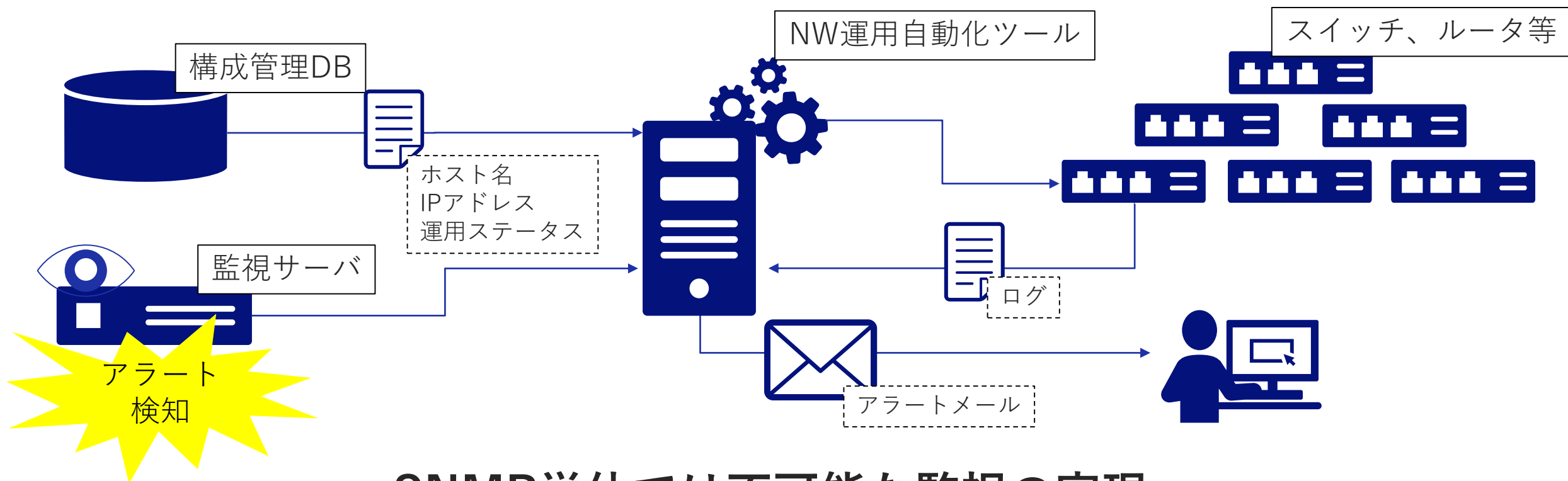


運用者はホスト名のみで任意の機器を指定可能

# ツール開発のモチベーション：“柔軟な処理”

## ◆ 例2：ネットワーク機器からログ取得 + 後処理で色々

■ 監視サーバと連携 → 正常性確認 → アラート発報



### SNMP単体では不可能な監視の実現

# ツール開発のモチベーション：“柔軟な処理”

◆ 構成管理データベースとの連携

◆ ネットワーク機器からログ取得 + 後処理で色々



大きなデータを処理しないといけない…

既存のツールだと対応、実装が辛いケースも…



# NW運用自動化ツール

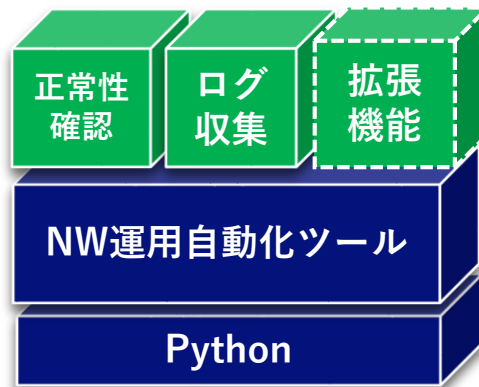
# NW運用自動化ツール



◆ 言語：Python

◆ 思想：

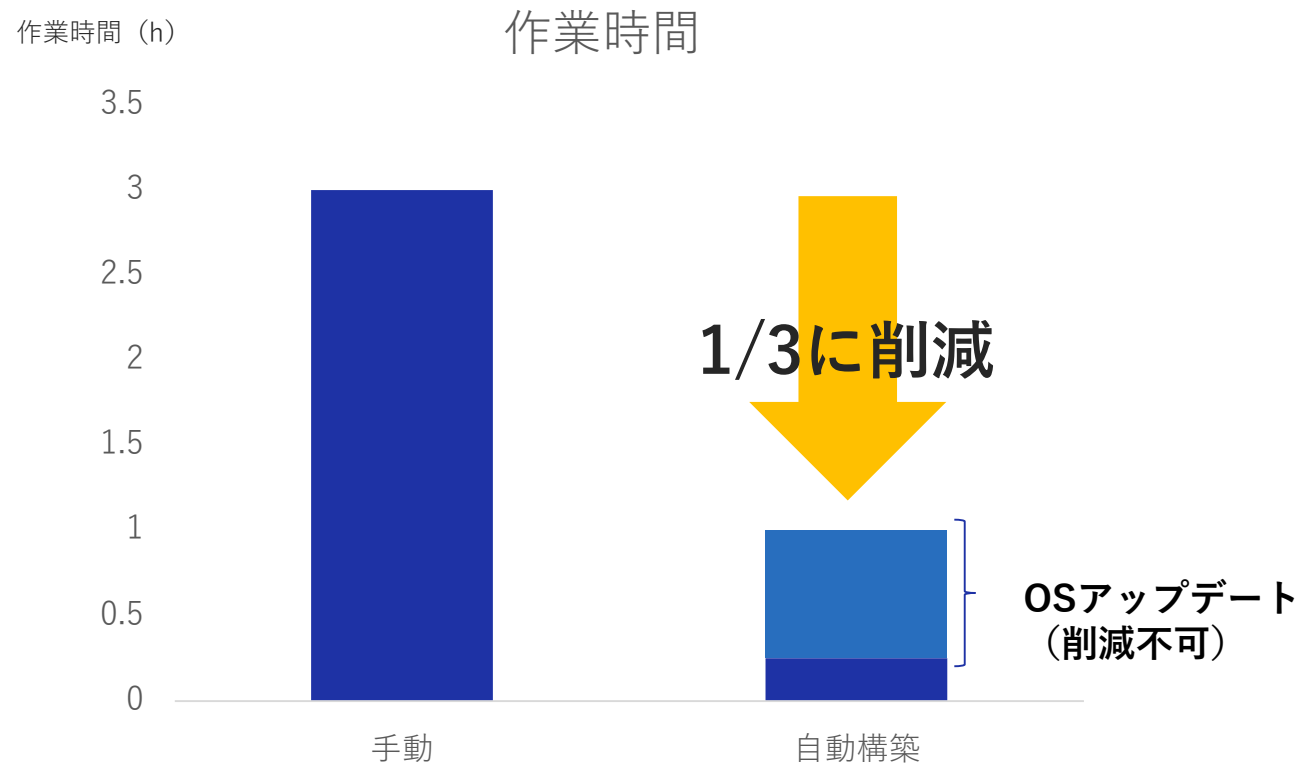
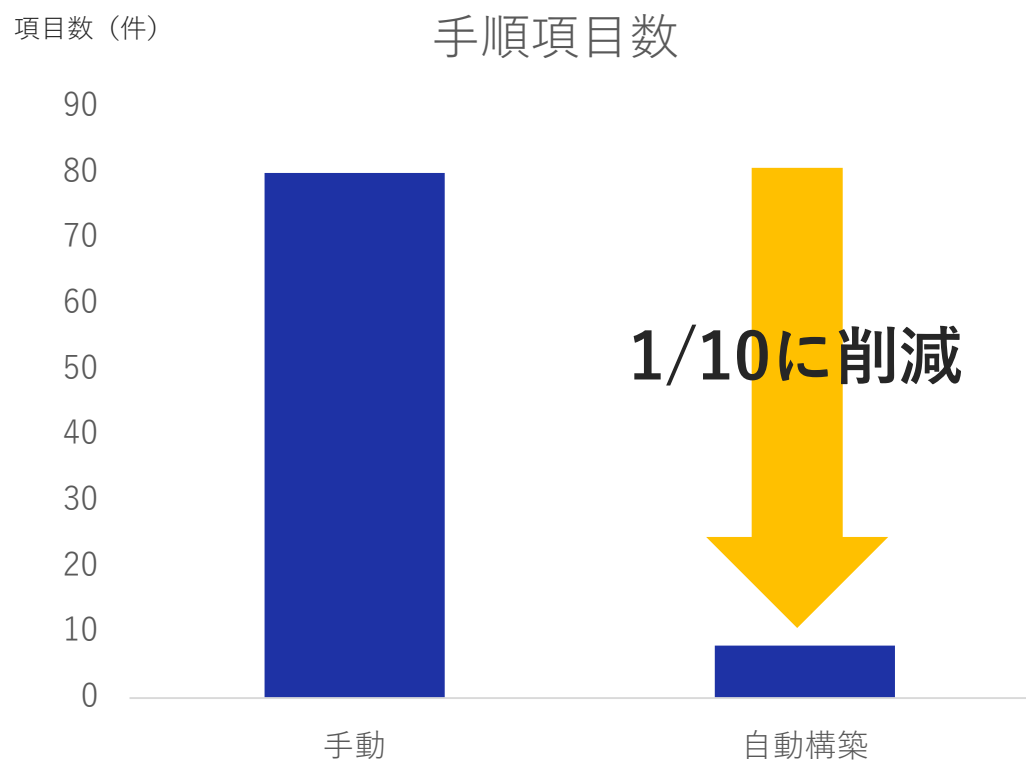
- マルチベンダ対応
- 機能追加時、個別処理以外の多くの汎用処理をツール側が担保
- 汎用/特化した処理をモジュール化、プラグインとして組み合わせ再利用



開発して良かった話

# 開発して良かった話

- ◆ 運用業務の効率化
- ◆ SNMP単体では不可能な監視の実現
- ◆ **構築自動化**



# 開発して良かった話： 構築自動化

## ◆ 3コマンドで構築完了

下のようなコマンドを3回実行するだけでOK!

```
sudo -u jncadmin /work/ansible/jncadmin/nw_auto_op/tool/nw_auto_op deploy -p init -t <ホスト名>
```

```
sudo -u jncadmin /work/ansible/jncadmin/nw_auto_op/tool/nw_auto_op deploy -p base -t <ホスト名>
```

```
sudo -u jncadmin /work/ansible/jncadmin/nw_auto_op/tool/nw_auto_op deploy -p nofn -t <ホスト名>
```

# 開発して良かった話： 構築自動化

```
sudo -u jncadmin /work/ansible/jncadmin/nw_auto_op/tool/nw_auto_op deploy -t bhnp13jt004 -p init
```

コマンド1行で構築実行

```
CONFIRMATION *****
```

```
HOSTNAME : bhnp13jt004
```

```
DB STATUS : 10
```

```
PHASE : init
```

```
CONFIG FILES:
```

```
- bhnp13jt004_20211108_100027_43086ec0_junos_init.conf
```

```
- bhnp13jt004_20211108_100027_43086ec0_assignment.yaml
```

使用するコンフィグは事前に確認可能

```
proceed? yes(y)| no(n) : yes
```

```
RESULT *****
```

```
Exit Code: 0
```

```
DEPLOY REPORT *****
```

```
ID: 20211108_183856_b1b194d0
```

```
HOSTNAME: bhnp13jt004
```

```
PHASE: init
```

```
RESULT: OK
```

```
FIRMWARE UPGRADE *****
```

```
STATUS: Executed: 18.4R2.7 -> 18.4R2-S6.3
```

```
RESULT: Successful
```

指定のOSへのバージョンアップにも対応

```
HEALTH CHECK *****
```

```
system_alarms: OK
```

```
chassis_alarms: OK
```

```
led_status: NG
```

構築後の正常性確認も自動で実施

# ご清聴ありがとうございました

## ◆ ネットワーク運用自動化に挑んでみた話 サマリ

### ■ ツール開発のモチベーション

- 既存のツールでは辛いケース
- 柔軟な処理（DB、監視との連携、後処理の追加等）

### ■ 開発して良かった話

- 運用業務の効率化
- SNMP単体では不可能な監視の実現
- 構築自動化

### ■ 苦労話

- 手動手順→自動化への落とし込み
- 標準出力を力技で処理（同じOSでも機種ごとに出力が違うケースが...）
- タイムアウト処理（コマンドの待ち時間等に応じたチューニングが必要...）

\Orchestrating a brighter world

**NEC**