



SmartCSのAnsible連携機能と
ShowNetにおけるSmartCS/GSの使われ方

- ・ 名前：佐藤 泰介 (さとう たいすけ)
- ・ 会社：セイコーソリューションズ株式会社 STN営業技術部
- ・ 職種：ネットワークエンジニア
 - SmartCS(コンソールサーバー)、SmartGS(ゲートウェイサーバー)
 - TimeServer(タイムサーバー)、NTPクロック を担当

SmartCSとは

- ・ コンソールポート
- ・ コンソールサーバ
- ・ SmartCSについて



■ コンソールポートの役割

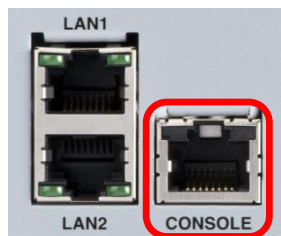
IP通信ではなく、シリアル通信でオペレーションをするためのインターフェース

・ 初期設定

IP設定、ユーザ作成、SSHの有効化 などの初期設定

・ 緊急時のオペレーション

LANポート障害、ネットワーク障害 などの影響で、
装置へIPアクセスできなくなった際の最後のアクセス手段



RJ45



DB9

■ コンソールサーバの役割

コンソールポートを集約し、リモートでアクセスできるようにする装置

・ リモートアクセス

IPアクセスできない状態の装置へ、リモートでアクセス可能

・ オペレーション範囲の拡大

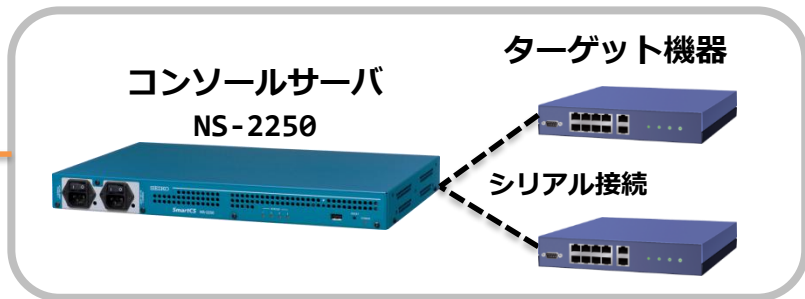
リモートからは実施しづらい作業を、安全に実行可能

ACL/ルーティングなどの設定変更、ファームウェアアップデート など

オペレーションセンター

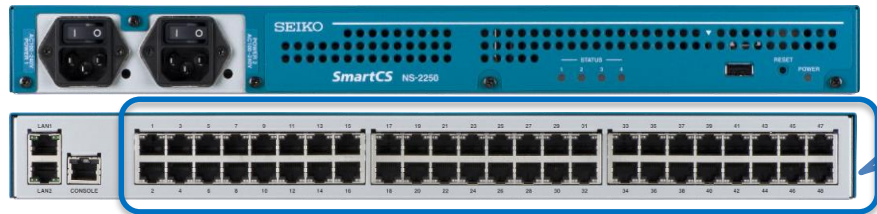


遠隔地のDCなど



SmartCS とは

SmartCS



対向装置のコンソールポートと接続するシリアルポート
1台で最大48のコンソールポートを集約可能

- 通信キャリア様、ISP様など大規模NWを運用しているお客様を中心に、SmartCSシリーズ累計で30,000台以上を出荷し、国内では高いシェアを確立
- INTEROP ShowNet のネットワーク構築においても10年以上の実績
- ネットワーク機器だけでなく、サーバ機器/認証機器などとの接続実績あり
ケーブル構成/ポーレート等を含め、接続実績表として公開中

https://www.seiko-sol.co.jp/products/console-server/console-server_connection/

SmartCSとAnsibleの連携

- ・ 連携メリット
- ・ 連携してできること
- ・ 連携方法



連携メリット

- **IPリーチできないターゲットも運用自動化の対象に**
 - 工場出荷状態でIP未設定の装置
 - 障害発生によりIPリーチできない装置
- **Ansibleモジュールのないターゲットも対象に**
- **リモートから敬遠しがちなオペレーションも自動化の対象に**
 - ネットワーク関連の設定変更 (IP/vlan/ルーティング/ACLなど)
 - 再起動を伴う処理 (ファームウェアアップデートなど)



Ansibleを用いた管理対象、操作内容を拡大！

SmartCS と Ansible の連携

構成と操作範囲

Ansibleホスト

SmartCS用Ansibleモジュール
(SmartCS modules for Ansible)SmartCS
NS-2250シリーズ

ターゲット装置

※NS-2250と
接続可能な装置

シリアル接続

- smartcs_command
- smartcs_config
- smartcs_facts

CLIコマンド機能 :

コンソールアクセス機能 :

- smartcs_tty_command
- ベンダーモジュールとの連携

本資料の
メインテーマ

連携方法

独自モジュール or ベンダモジュール
2パターン用意しています

	smartcs_tty_command	ベンダーモジュールと連携
特徴	<ul style="list-style-type: none"> SmartCS用の独自モジュール TeraTermマクロのように、CLIの入出力を定義してPlaybookを作成 	<ul style="list-style-type: none"> ベンダーモジュールをSmartCS経由(コンソール経由)で実行可能
利点	<ul style="list-style-type: none"> モジュールがないターゲットの管理 既存モジュールでは実現が難しい作業(reboot/verup など)に対応する場合 	<ul style="list-style-type: none"> Playbookの再利用が可能 ※vars変数のみで切替
注意	<ul style="list-style-type: none"> 冪等性の担保がない Playbook作成に入出力情報が必要 	<ul style="list-style-type: none"> network_cli をサポートしているモジュールのみ連携可能

※どちらの場合も共通して処理時間が長くなる為、タイムアウト値の調整が必要です

SmartCS × Ansible連携のユースケース

- ・ コンソールから実施するオペレーション
- ・ Ansible連携のユースケース

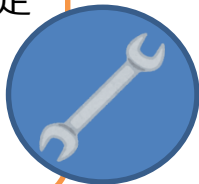


コンソール経由で実現できるオペレーション

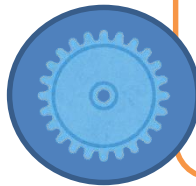
Ansibleリーチまでの初期設定

- ・ IP設定、SSH有効化
ユーザ作成 など
- ・ ZTPを簡易に実現

キッティング



再起動を 伴う処理



メンテナンス

- ・ reboot / reload
- ・ OS / ファームウェア更新

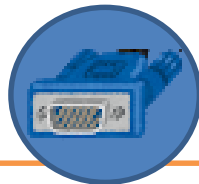
運用



遠隔から敬遠しがちな作業

- ・ 経路設定の変更
- ・ ACL設定

コンソール特有



初期化作業

- ・ 検証時

監視 & 確認



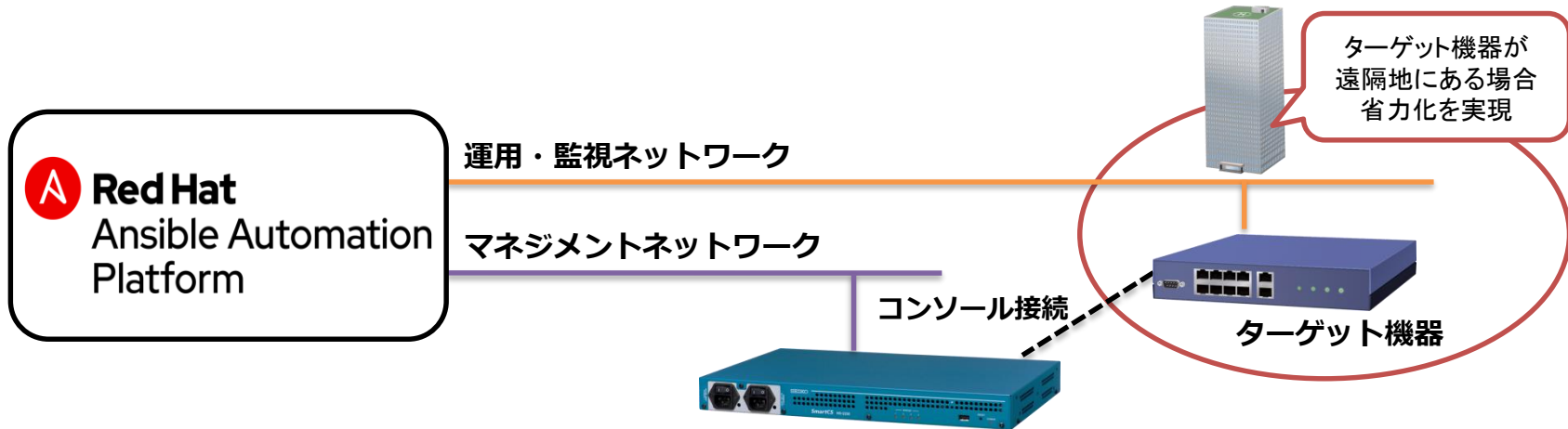
- ・ IPリーチできない時の
情報収集(Console経由)
- ・ 運用系に接続していない
スタンバイ機器の確認

時間なければい
くつかは割愛

ケース① 初期構築（設置時・交換時）

NW機器の設置時や交換時に、**コンソール経由でIP設定をはじめとした初期構築を行います。**
IP設定後 疎通確認までを行う事で、設置・交換後の設定投入の自動化を実現します。

構成



ワークフロー

— On Success
— On Fail

初期構築の設定投入
(SmartCS用 モジュール)

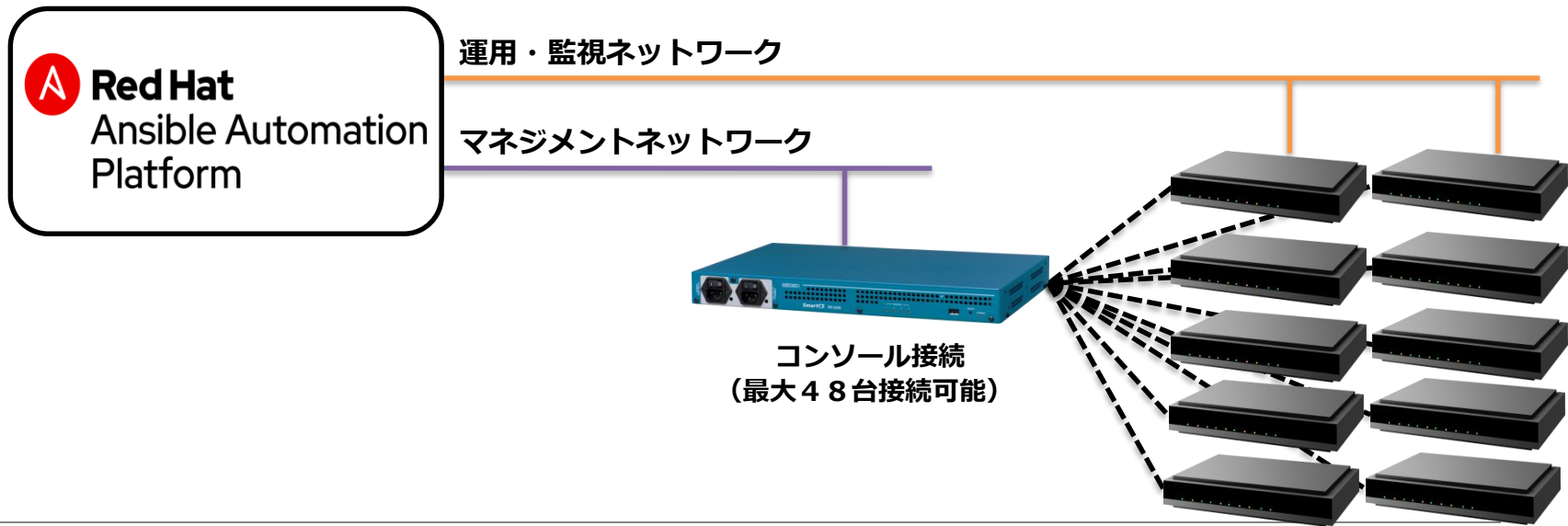
追加の設定投入
(ベンダーモジュール)

疎通確認・設定確認
(ベンダーモジュール)

ケース② 大量のNW機器のキッティング作業

AnsibleのPlaybookで IP設定の情報を変数化し、コンソール経由で設定を投入する事で大量のNW機器のキッティング作業を効率よく実現する事ができます。

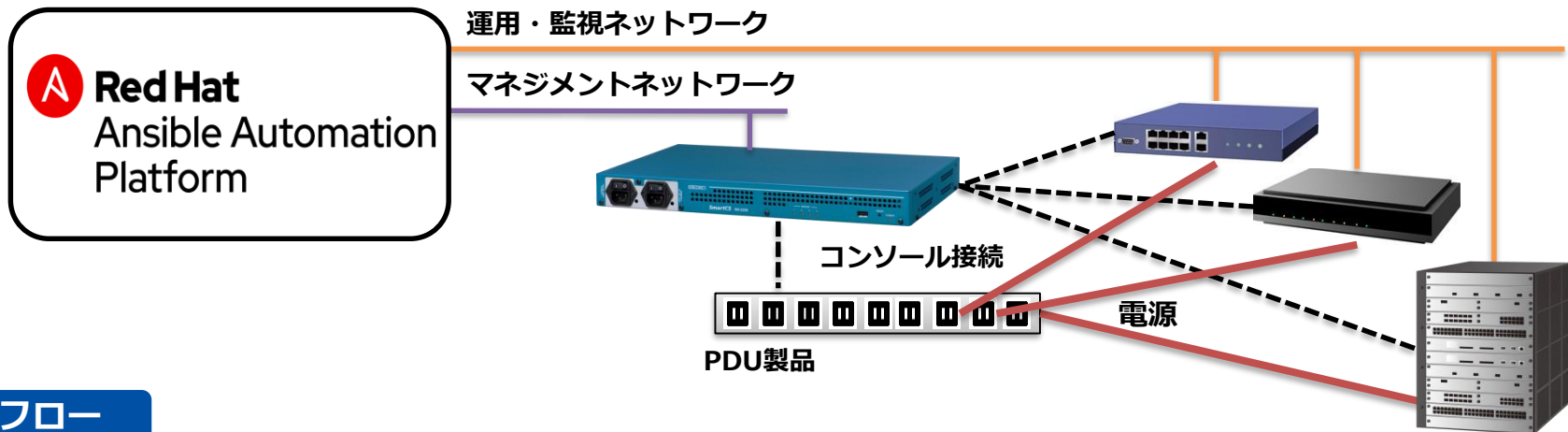
構成



ケース③ お手軽ZTP (Zero Touch Provisioning)

SmartCSとAnsibleだけあれば、環境構築に必要なサーバ（DHCP、TFTP）を用意する事なく、ZTP環境を構築する事ができます。

構成



ワークフロー

— On Success
— On Fail

PDU製品から電源ON
(SmartCS用 モジュール)

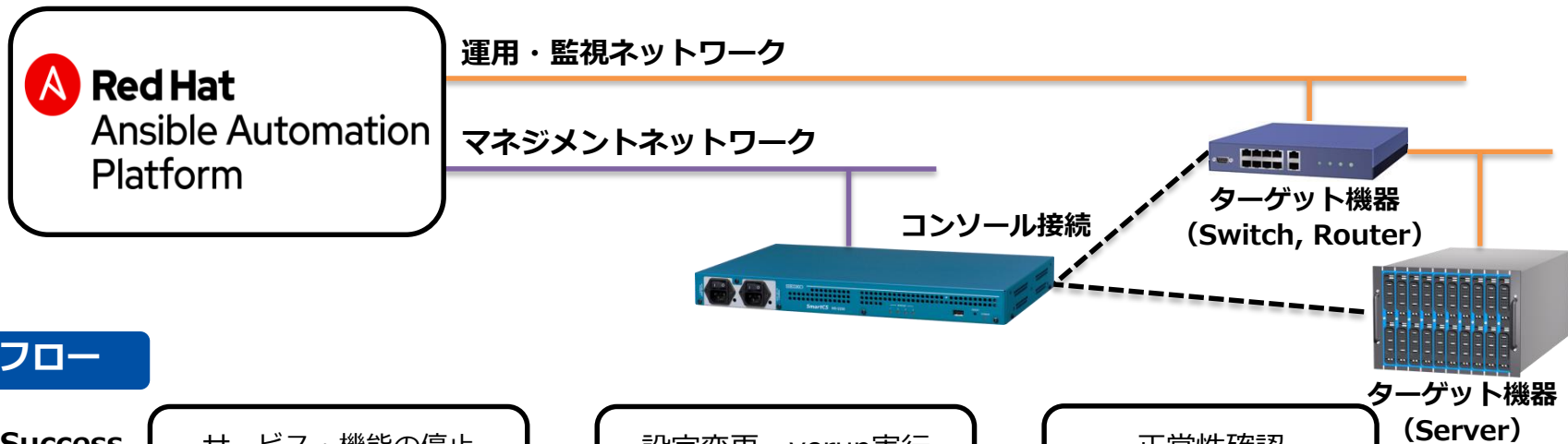
初期構築の設定投入
(SmartCS用 モジュール)

追加設定・疎通/設定確認
(ベンダーモジュール)

ケース④ 運用での利用

NW機器やサーバ（Hypervisorのホスト）機器の設定変更・バージョンアップ作業について
コンソール経由で安全に行う事ができます。

構成



ワークフロー

— On Success
— On Fail

サービス・機能の停止
(バンダーモジュール)

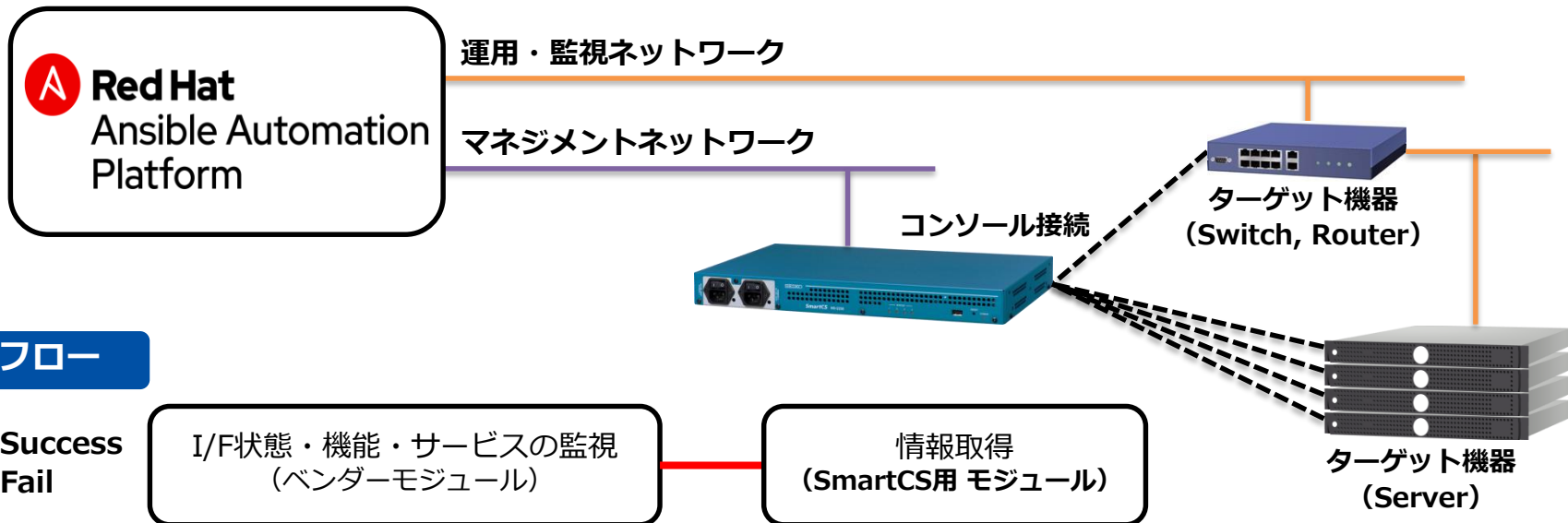
設定変更・verup実行
(SmartCS用 モジュール)

正常性確認
(バンダーモジュール)

ケース⑤ 監視用途

運用・監視ネットワークでターゲット機器の状態をスケジューリングで確認し、障害発生時に情報取得（一次切り分け情報等）を行うことができます。

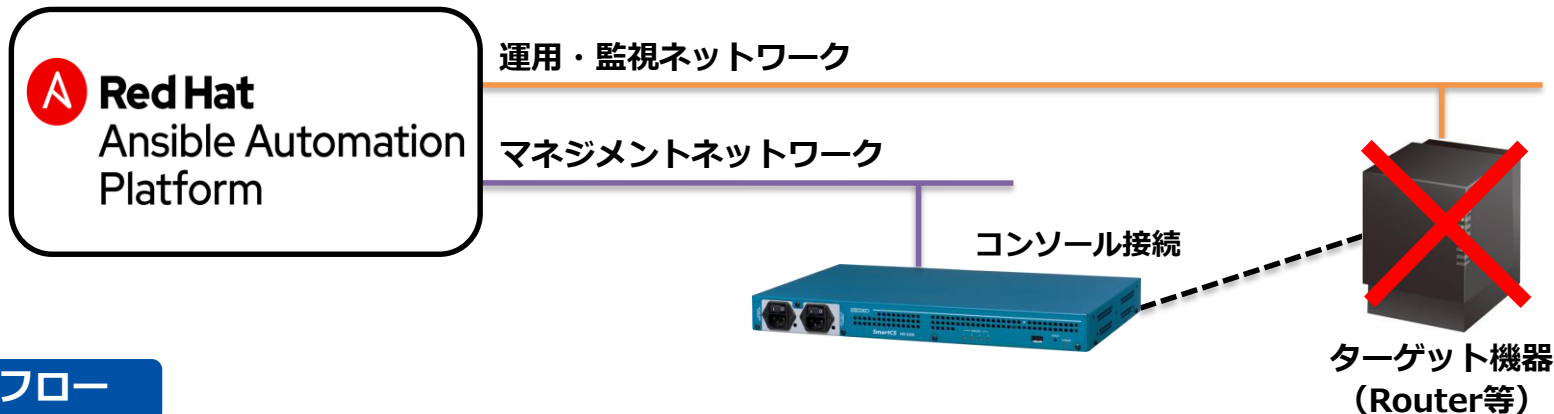
構成



ケース⑥ 緊急時のコンソールオペレーション

運用・監視ネットワークの経由でターゲット機器にアクセスできなくなった場合に、
コンソール経由で **reboot**や**shutdown**を実行し、障害の早期復旧の自動化を手助けします。

構成



ワークフロー

— On Success
— On Fail

定期的な確認
(ベンダーモジュール)

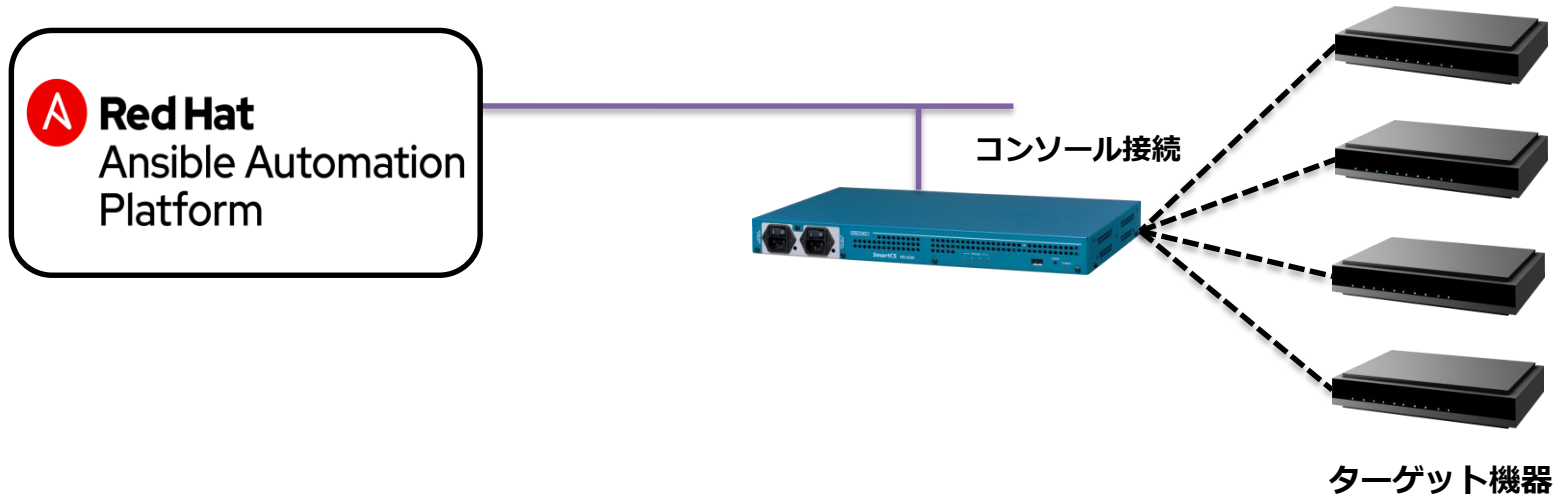
情報取得
(SmartCS用 モジュール)

reboot・shutdown実行
(SmartCS用 モジュール)

ケース⑦ NW機器の初期化作業

NW機器のコンフィグクリア作業 や 工場出荷状態に戻す といった 初期化用途で利用できます。
CLI実行による初期化、初期化プロセスの実行 などのコンソールで行う作業を自動実行できます。

構成



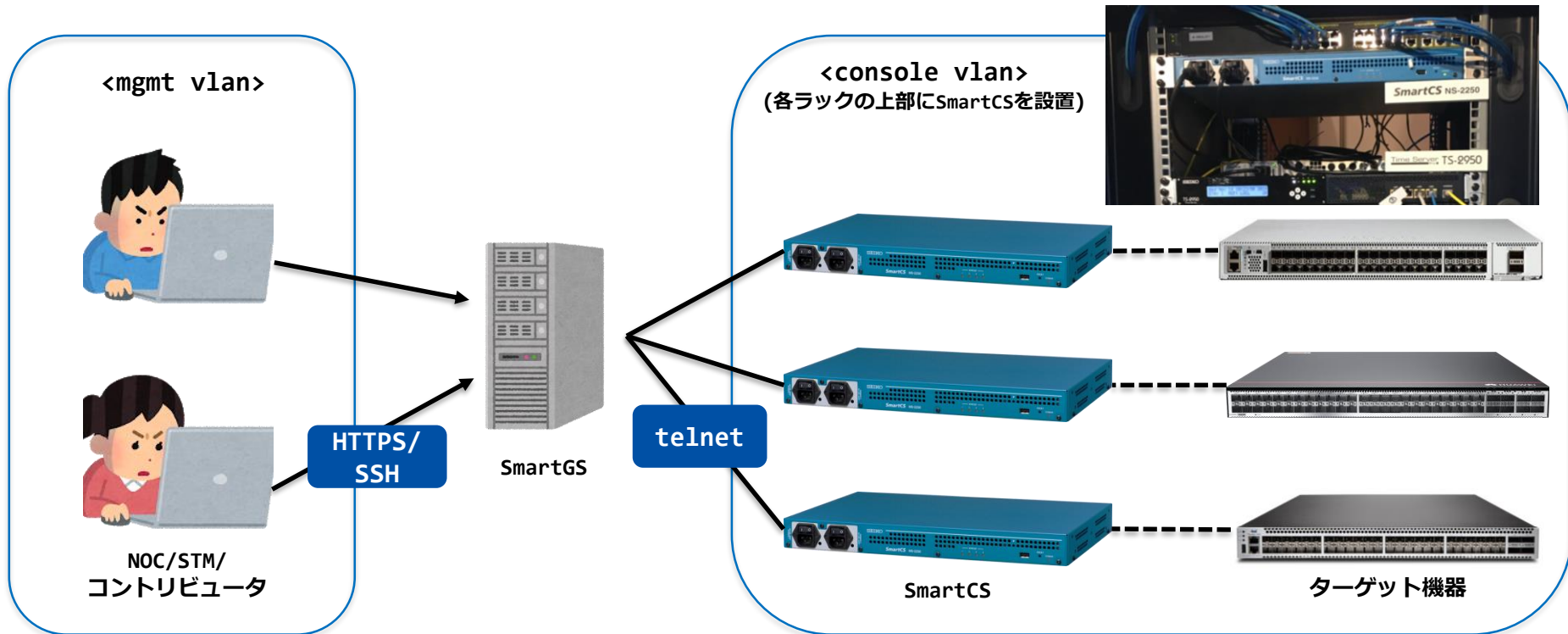
ShowNet2021での取り組み

- ShowNetにおけるSmartCS/SmartGS
- SmartCS × Ansibleの連携

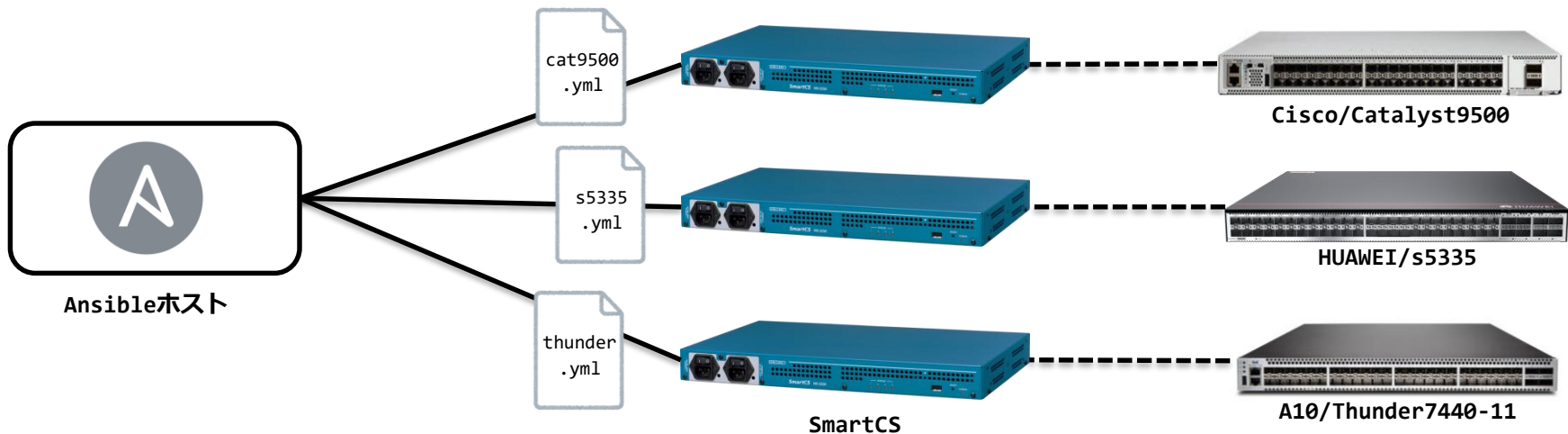


ShowNet2021での取り組み

- ShowNet2021では初期設定/緊急時のコンソールアクセス経路として、SmartCS(コンソールサーバ)/SmartGS(踏み台サーバ)がDay2から稼働しています。



- Interop出展社のブースネットワークを収容するスイッチ/ルータの設定自動化を実施。マルチベンダーで構成される環境で、ターゲット機種ごとのplaybookを作成し出展社ブース用のconfig(vlan/IP等)をコンソール経由で設定投入しました。



- Ansibleと連携し、コンソール経由でのconfig投入自動化に成功
- 連携実績とplaybook情報などのノウハウ蓄積 (3ベンダー/7機種)

Update情報

- Ansible Collectionsへの対応
- RedHat社 Certifiedの取得
- RestAPIへの対応



{ REST : API }

■ Ansible Collectionsへの対応

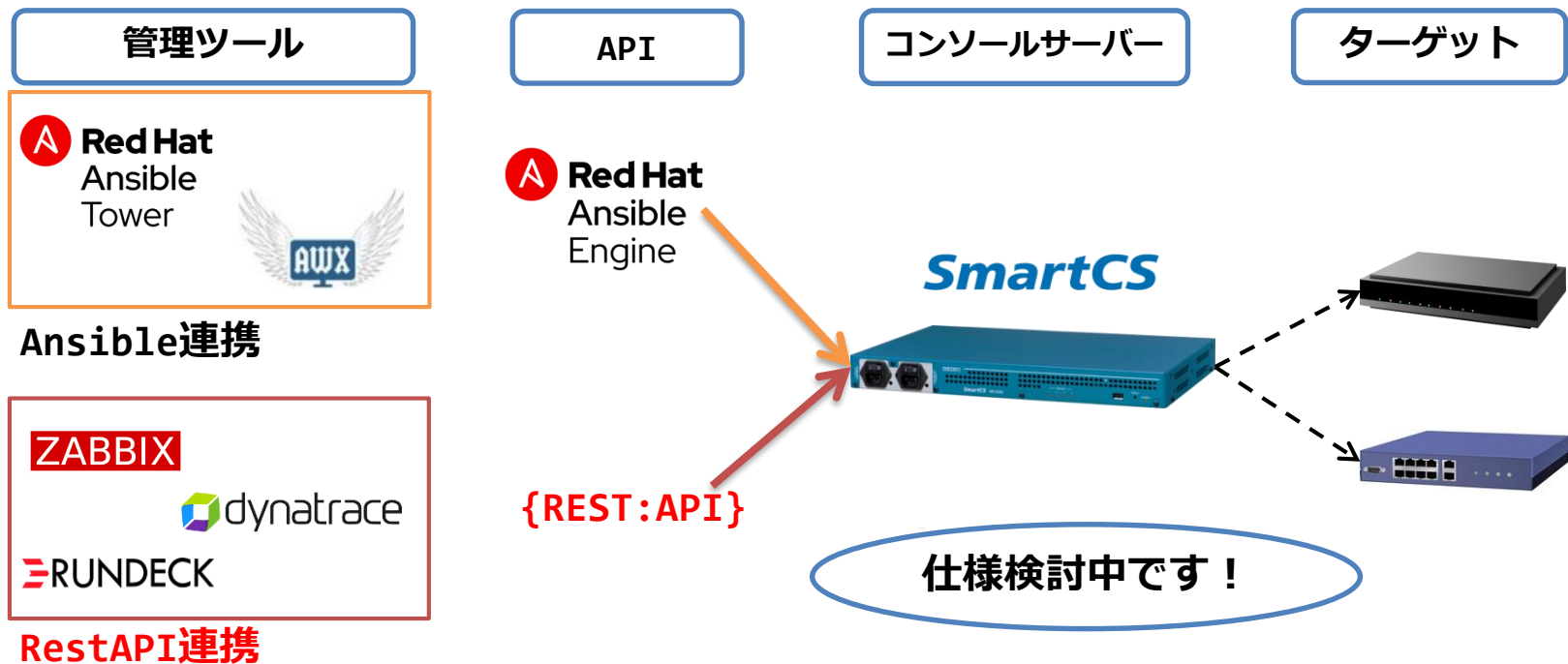
- 4/12(月) Ansible Collectionsに対応したモジュール(v1.3.0)をリリースしました！
- Ansible Galaxy
<https://galaxy.ansible.com/seiko>
- Github
<https://github.com/ssol-smartcs>

■ RedHat社 Certifiedの取得

- RedHat社 Certifiedモジュール認定を目指しています！

■ RestAPIへの対応

コンソールアクセスを『運用自動化』のパーツとしてより使いやすくする為、Ansible以外のAPI連携についても機能拡張を検討しています！



■ 弊社ブースへ是非お越しください！

ブース



出展製品

- コンソールサーバー SmartCS
- アクセス管理ソフトウェア SmartGS
- PTPグランドマスタークロック Time Server Pro. シリーズ
- タイムサーバー
- セイコータイムマネージドサービス
- ネットワーク型アナログ/デジタル時計 NTPクロック
- ソフトウェアロードバランサー Netwiser Virtual Edition
- DX Challenge 2021 (参考出品)
- マルチキャリアLTE対応IoTルーター SkyBridge
- Wi-Fiアクセスポイント SkySpider
- アプリケーションパフォーマンス管理ソリューション Dynatrace
- インシデント管理ソリューション PagerDuty
- AI型予兆予測サービス LUiNa