



ダイナミックなシステムへの進化を見据えた 構築・運用の自動化のすすめ

世の中はクラウドネイティブなシステムを目指して切磋琢磨しています

新技術を取り込みにくい

密結合で保守しづらいアプリ

従来のスタティックな環境

手動構築・手動運用



Mobile

Social

IoT

5G

Tech

スケーラブルなアプリ

コンテナ

サービスメッシュ

イミュータブルインフラストラクチャ

宣言型API

AP

近代的でダイナミックな環境

パブリッククラウド

プライベートクラウド

ハイブリッドクラウド

PF

自動構築・自律運用

OP

課題はクラウドネイティブなシステムを提供できる技術者の不足です

新技术を取り込みにくい

密結合で保守しづらいアプリ

従来のスタティックな環境

手動構築・手動運用

多くのITエンジニアが張り付いている

Mobile

Social

IoT

5G

Tech

ITエンジニア不足 スケーラブルなアプリ

コンテナ
サービスメッシュ
イミュータブルインフラストラクチャ
宣言型API

AP

近代的でダイナミックな環境

パブリッククラウド
プライベートクラウド
ハイブリッドクラウド

PF

自動構築・自律運用

OP

現行システムの構築・運用を自動化して技術者を確保する必要があります



(1)構築・運用を自動化・省力化して

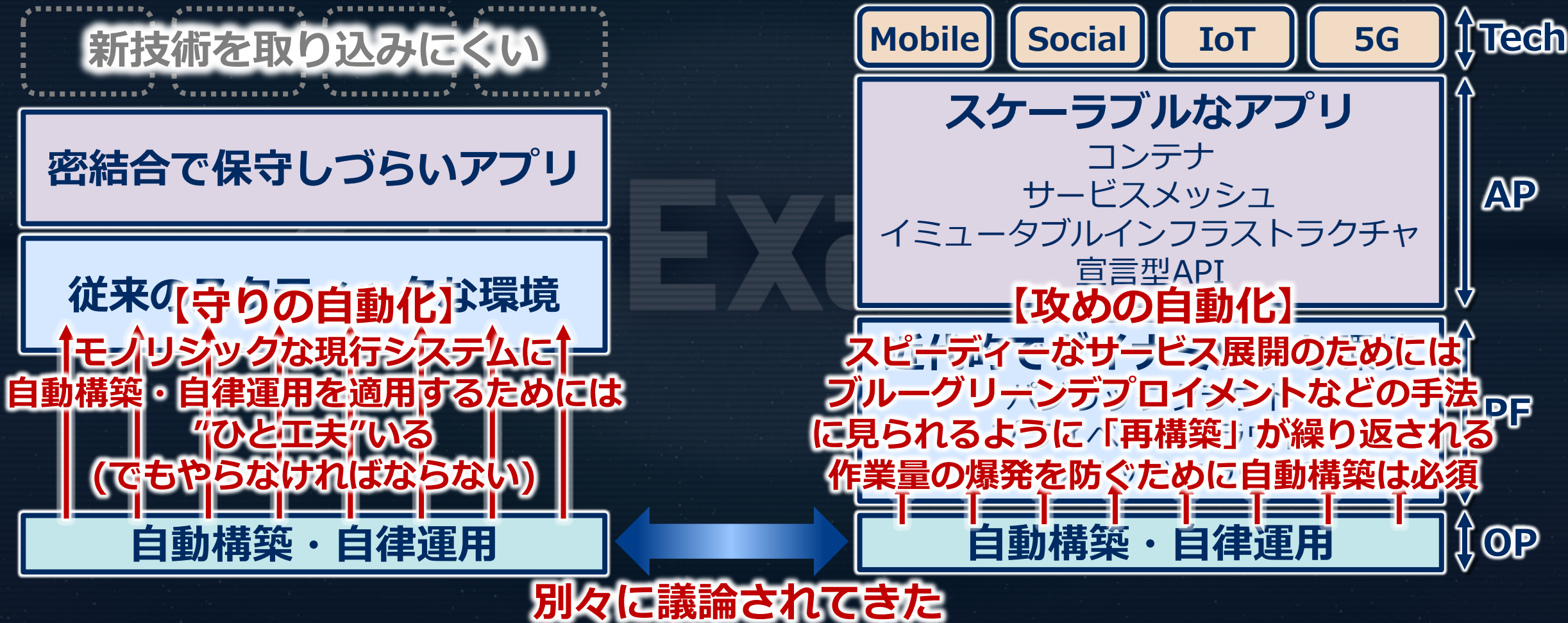
現行システムの構築・運用を自動化して技術者を確保する必要があります



現行システムの構築・運用を自動化して技術者を確保する必要があります



これまで2つの領域のアプローチが別々に議論されてきました



2つの領域の自動化は財務的な視点でも**目的を異にします**



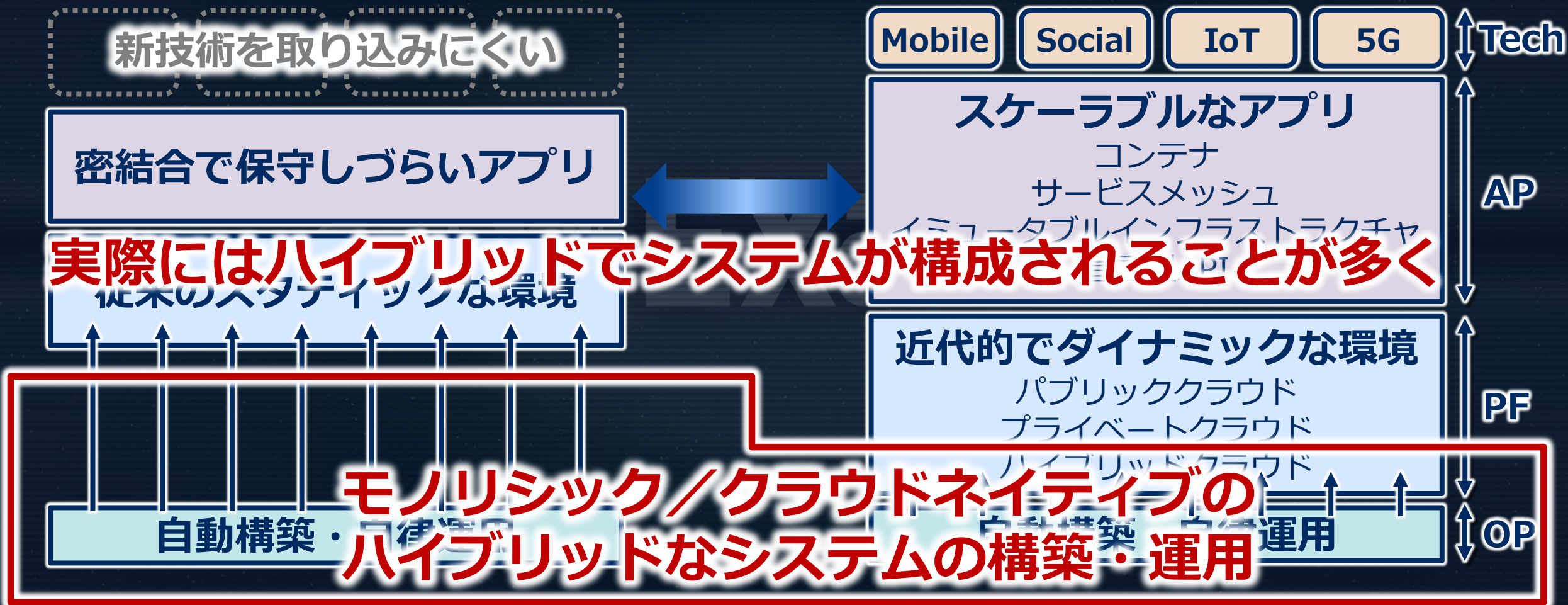
守りの自動化はOPEX(維持費)の効率化を目的とすることが多いです



攻めの自動化はROI(投資対効果)の引き上げを目的とすることが多いです



しかし2つの領域は実際にはアプローチを融合して考える必要があります



それでは「**守りの自動化**」と「**攻めの自動化**」について解説します

新技术を取り込みにくい

密結合で保守しづらいアプリ

従来のスタティックな環境

モノリシックな
システムの自動化
(守りの自動化)

Mobile

Social

IoT

5G

Tech

スケーラブルなアプリ

コンテナ

サービスメッシュ

イミュータブルインフラストラクチャ

宣言型API

AP

近代的でダイナミックな環境

マイクロサービス

クラウドネイティブな

システムの自動化

(攻めの自動化)

PF

【守りの自動化】
モノリシックなシステムの構築・運用に携わる
ITエンジニアの「苦」

モノリシックシステムの SIに携わるITエンジニアの現場の声をまとめてみました

設計

- ✓ チーム間の情報伝達に遅延やミスが発生する
- ✓ データの二重管理や独自文言が設計ミスにつながる
- ✓ 多重開発により設計書(帳票)の管理が煩雑化する
- ✓ 結果として設定の前後性を確認できない

作業準備

- ✓ チーム間の作業順序が複雑で毎回タイムチャートを作成しては使い捨てる
- ✓ 作業ごとに手順書を作成/レビューしては使い捨てる
- ✓ 手順ごとにコンフィグを埋め込んでいて、新機種/新OSを追加するごとに手順書のパターンが増える(マルチベンダー対応の障壁)

作業実施

- ✓ 人手作業なので作業時間が一定でない
⇒チーム間で作業待ちが発生
- ✓ 人手作業なので人為ミスの懸念から逃れられない

モノリシックシステムの 運用に携わるITエンジニアの現場の声をまとめてみました



- ✓ 運用上変更してよいパラメータと変更してはいけないパラメータが把握できていない
- ✓ システムのパラメータの現在値や過去の変更履歴を管理できていない
- ✓ 結果としてせつかくパラメータ化されているのに運用で利用できていない

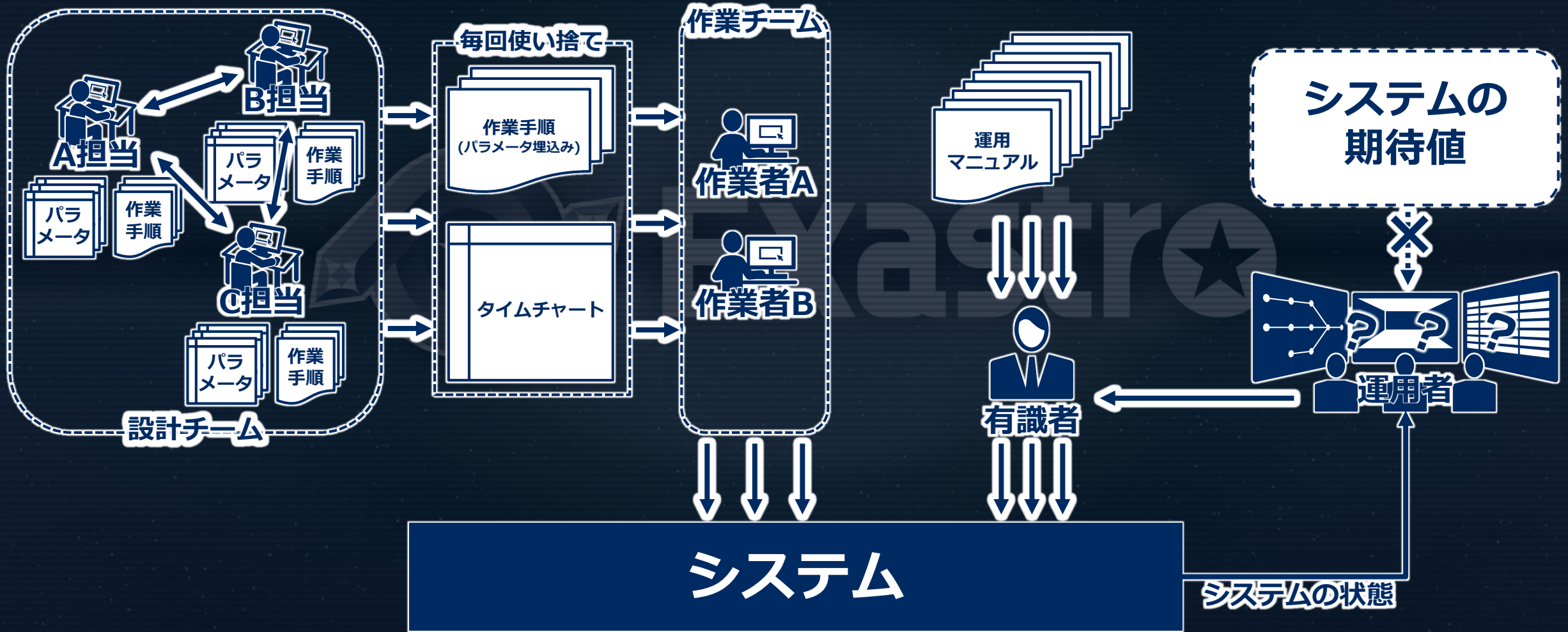


- ✓ システムは複雑化の一途を辿っており作業量は増大するばかり
- ✓ 何か起こるとExcelで書かれた大量のマニュアルを読み替えながら複数人体制で1作業ずつ慎重に実行するしかない
- ✓ 結果としてシステムの故障時間が長くなりサービスにも影響が出る

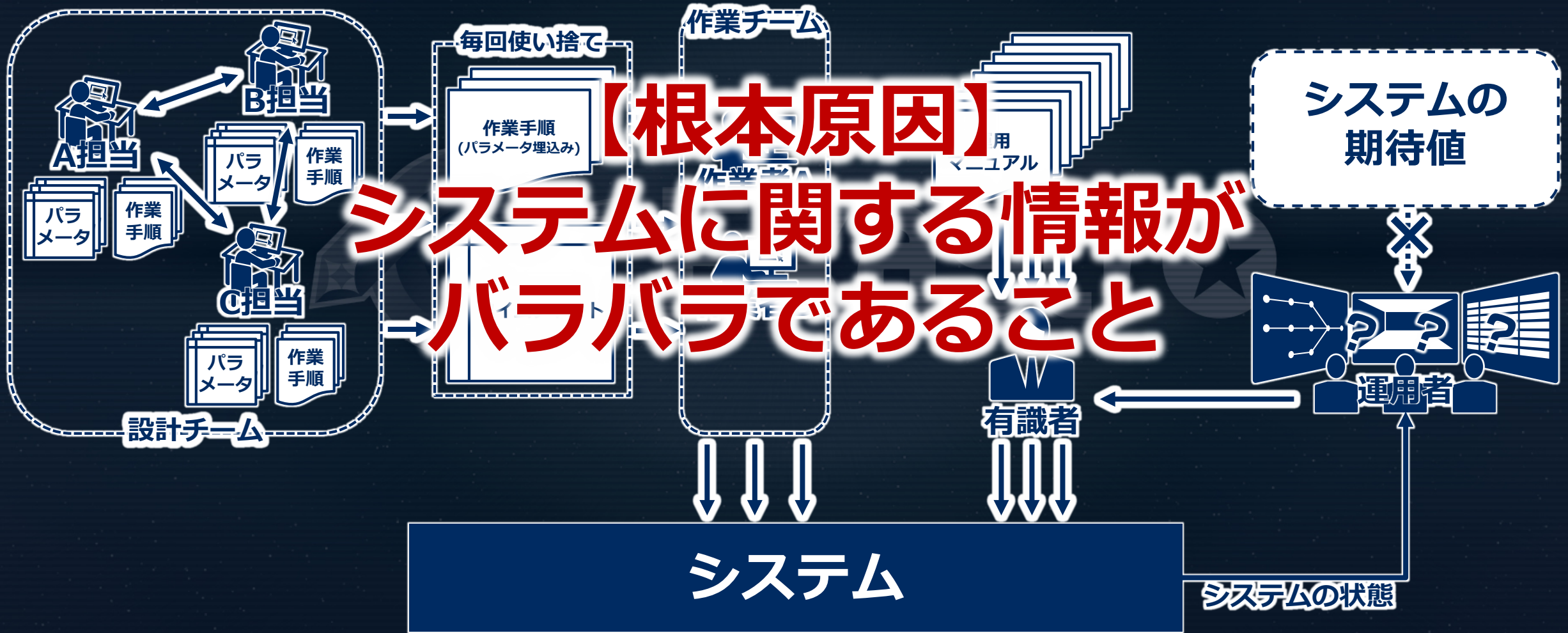


- ✓ 有識者不在により作業が進まない
- ✓ 有識者がいなくなるとノウハウは消失する
- ✓ 既知事象/未知事象の切り分けが難しく有識者の経験に頼らざるを得ない
- ✓ 結果として有識者を異動させられない

関係者は散乱した情報を正確に伝えることに多くの時間を費やしています

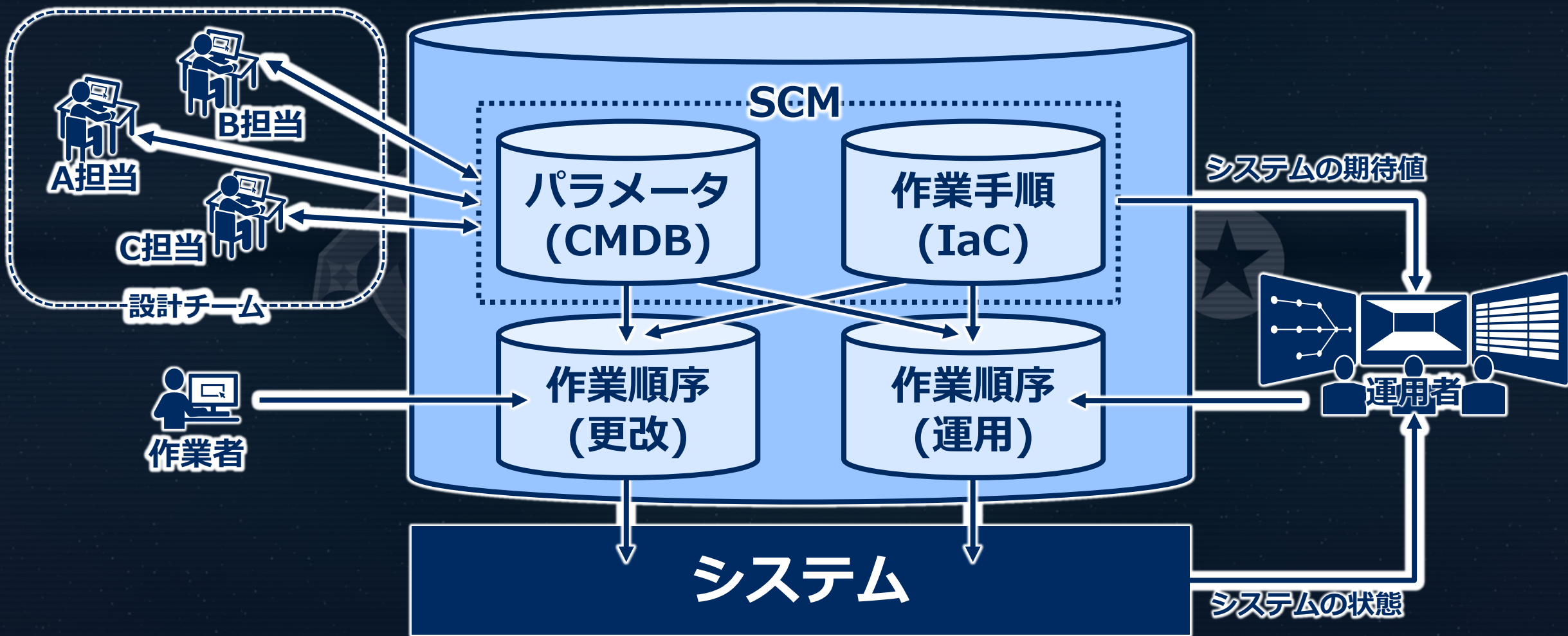


関係者は散乱した情報を正確に伝えることに多くの時間を費やしています



ITエンジニアの「苦」を解決するためには？

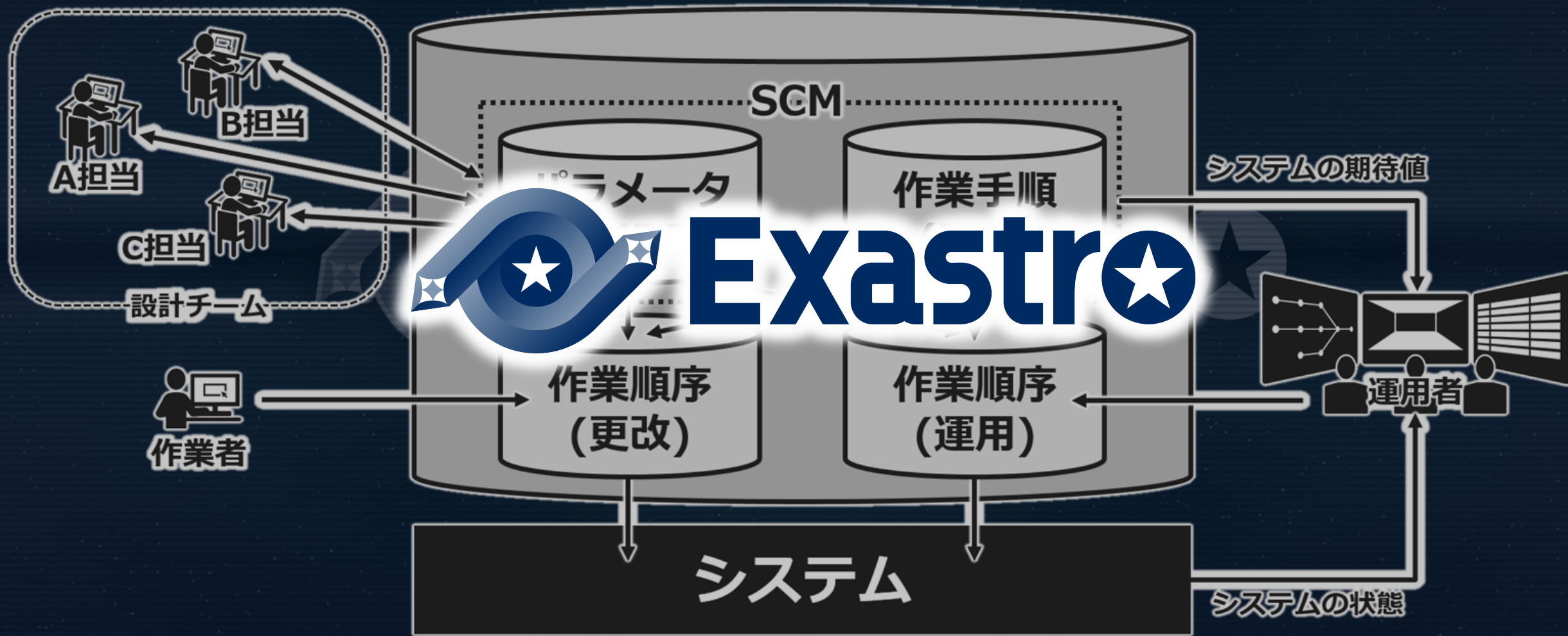
システムに関する情報をデジタル化して一元管理すればよいのですが...



システムに関する情報をデジタル化して一元管理すればよいのですが...



Exastroはシステム情報をデジタル化して一元管理するのに役立ちます



【事例】大規模イベント向けに3万台の機器を自動キッティングしました

課題



大量のネットワーク機器の初期設定を行う必要があった。しかし、手作業では多くの工数が必要であり、また大量の設定値を表計算ソフトで管理するのにも限界があった。

解決策

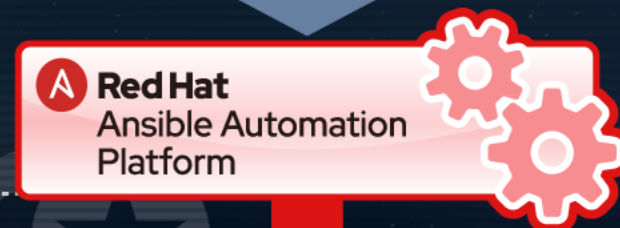


SmartCSを活用して複数のネットワーク機器を同時に設定可能にした。
また、設定値は**CMDBで一元管理**し、**Ansibleで設定を自動投入**することで課題を解決できた。

効果



スイッチなど6種類のネットワーク機器の初期設定を自動化することができ、
2,000台/月のリリースを達成した。



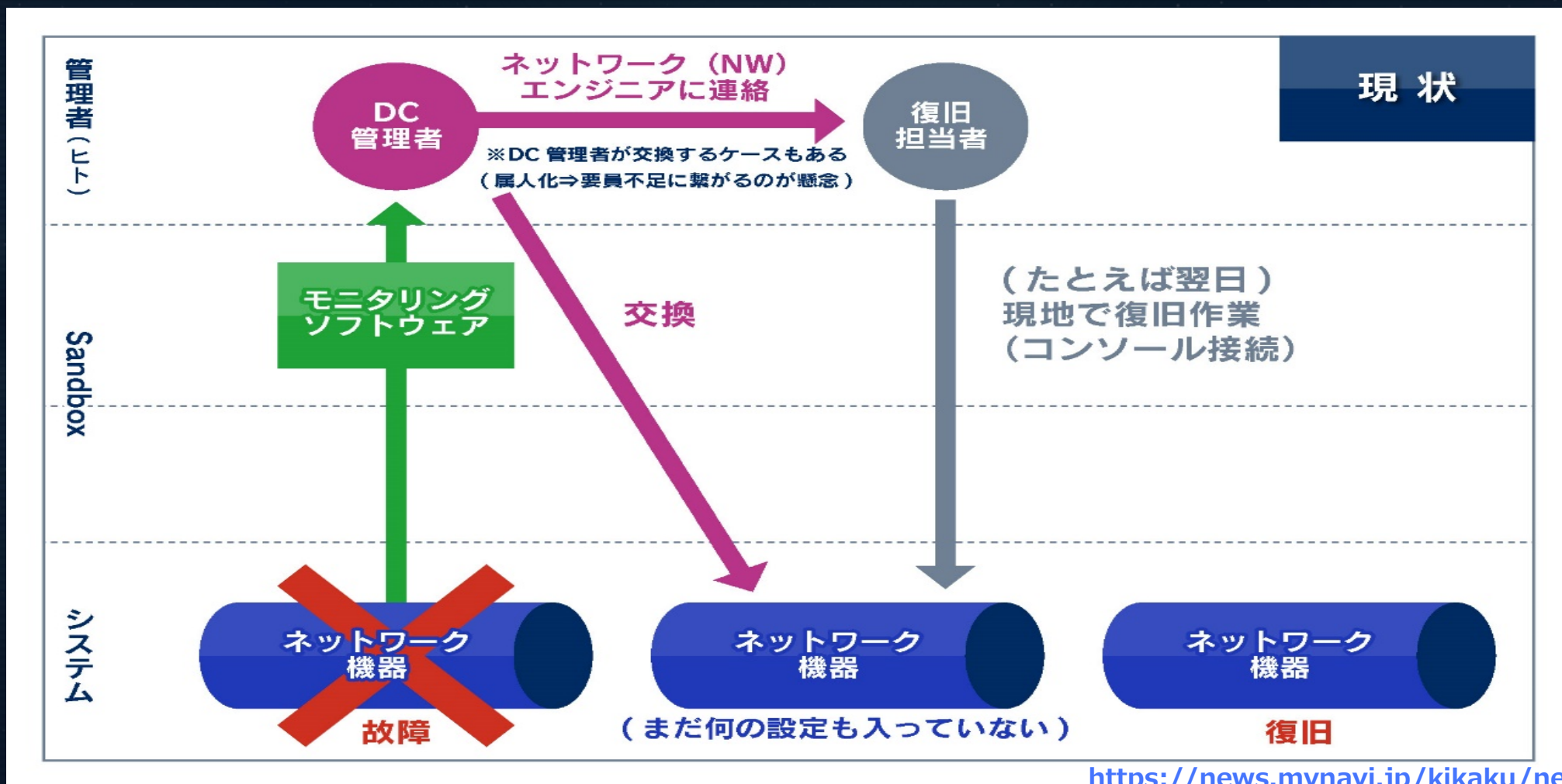
SSH



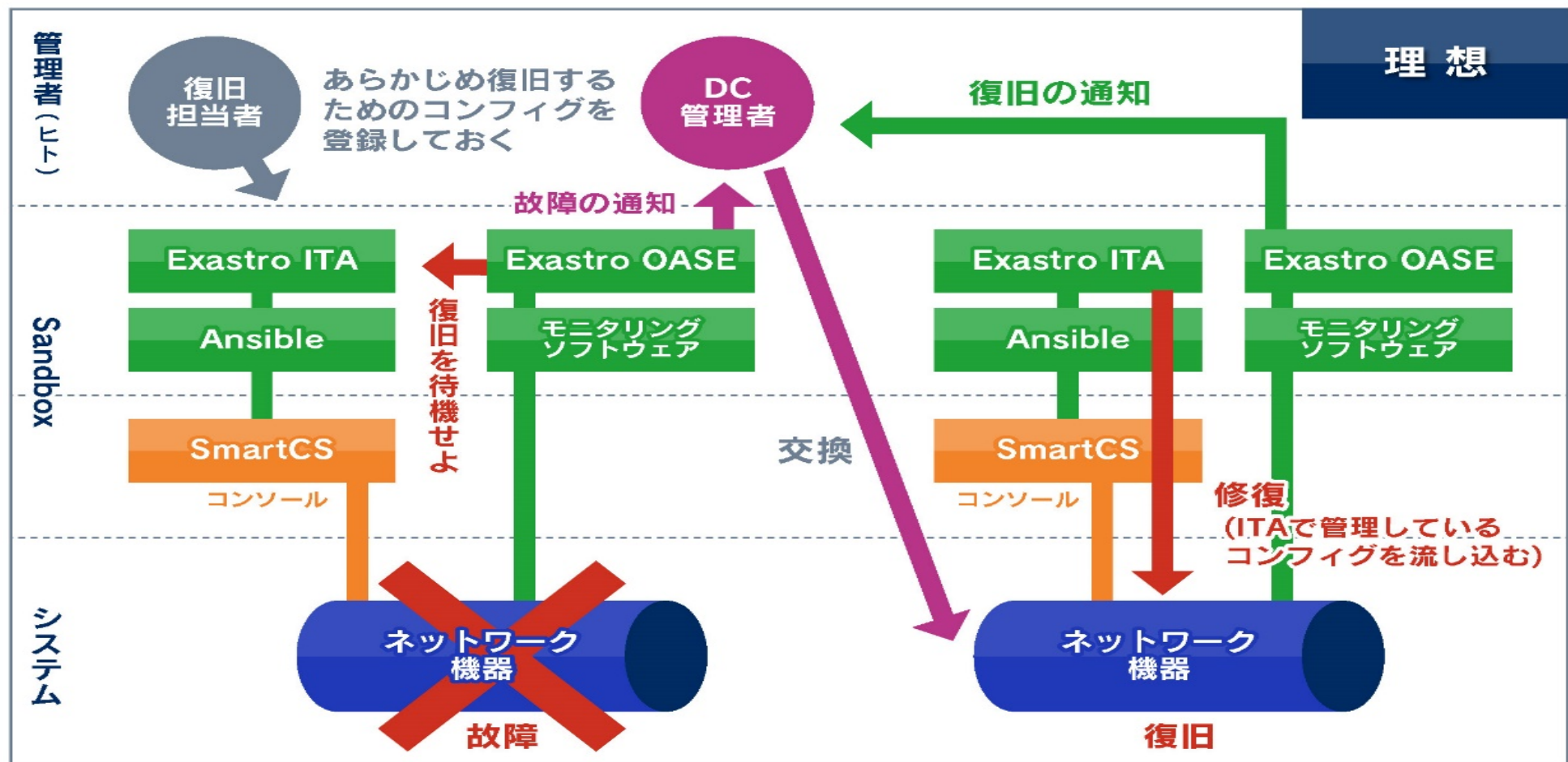
Console



データセンターに設置している機器に障害が発生すると、エンジニアは現地で再設定作業を実施する必要があります



Exastro × Ansible × SmartCS により、エンジニアが現地まで行かずとも、ネットワーク機器の交換や初期設定が可能となります (=ZTPの実現)



IPリーチャビリティがない領域

ネットワーク機器の運用をいかに
リモート化&省力化すべきか?

日本電気株式会社
サービス&プラットフォームSI事業部
プロジェクトマネージャー

吉田 功一 氏

FOR MORE DETAILS

https://news.mynavi.jp/kikaku/nec_exastro-2/



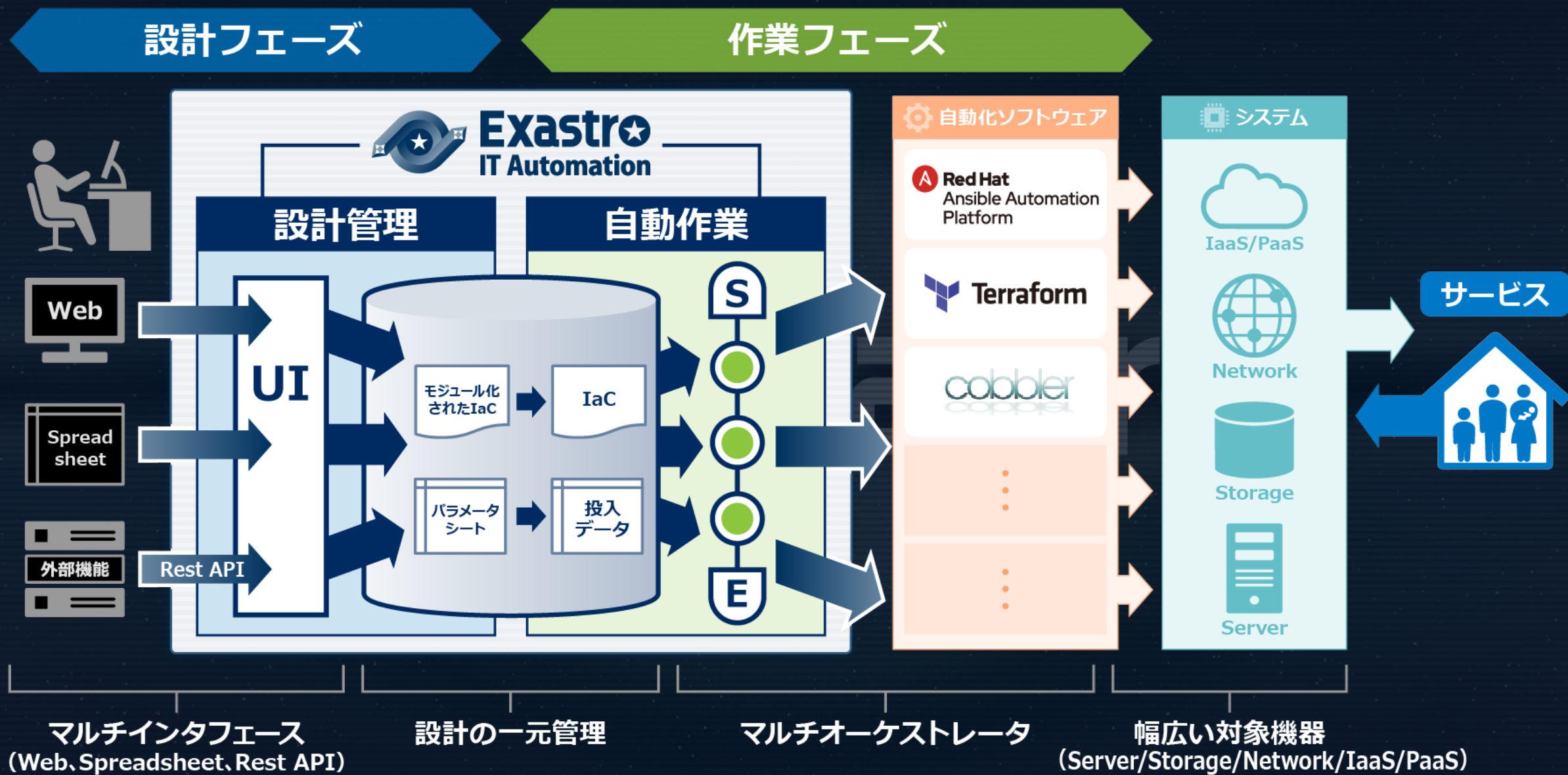
セイコーソリューションズ株式会社
ネットワークソリューション本部
NS開発部 NS開発4課

中山 真一 氏

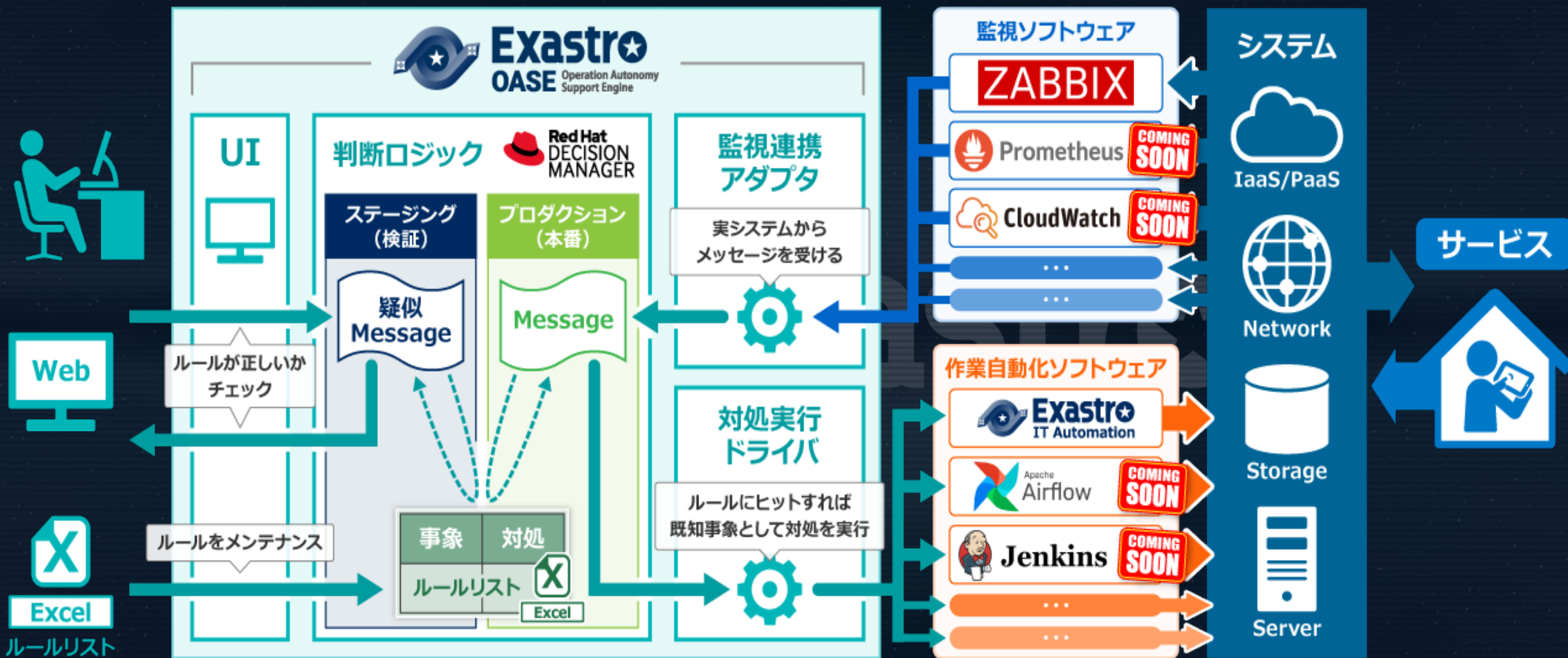
Exastro Suite



Exastro IT Automation : システム情報をデジタル管理するためのフレームワーク



Exastro OASE : システム運用の自動化を支援するためのソフトウェア



詳しくはコミュニティサイトへ！



Exastro

 Search

Exastro

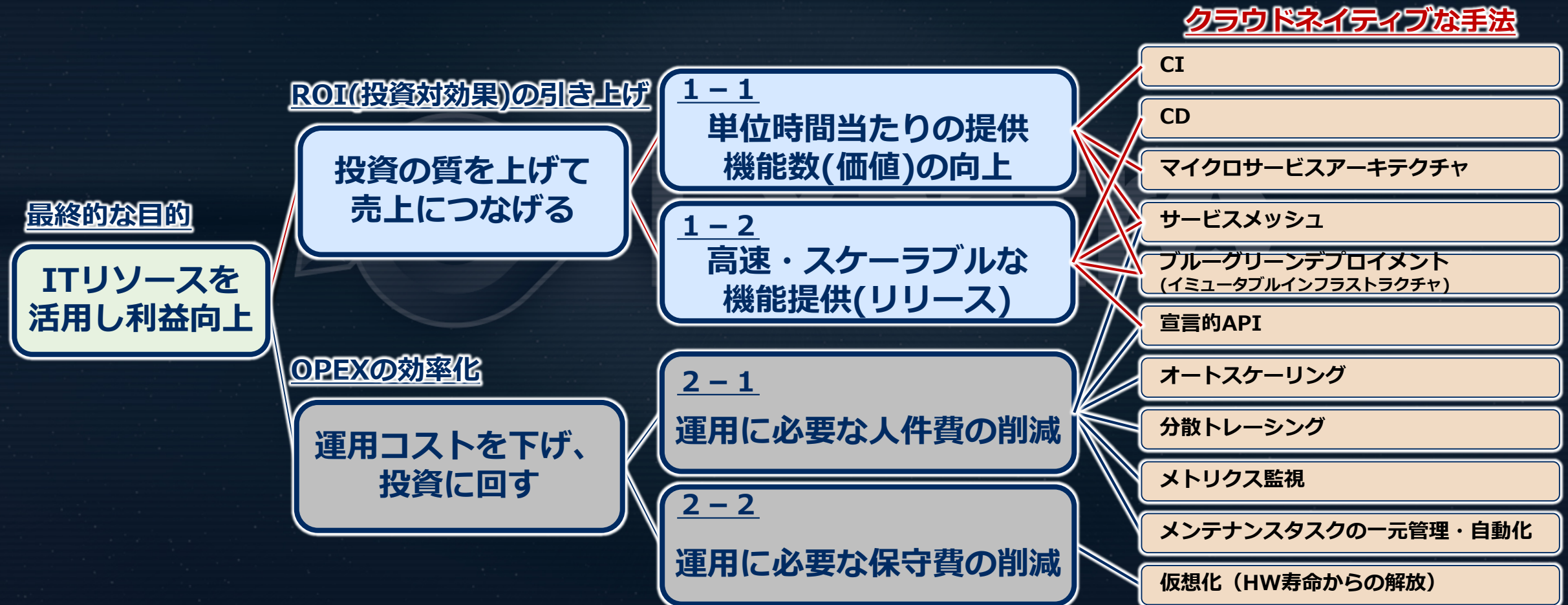
https://exastro-suite.github.io/docs/index_ja.html



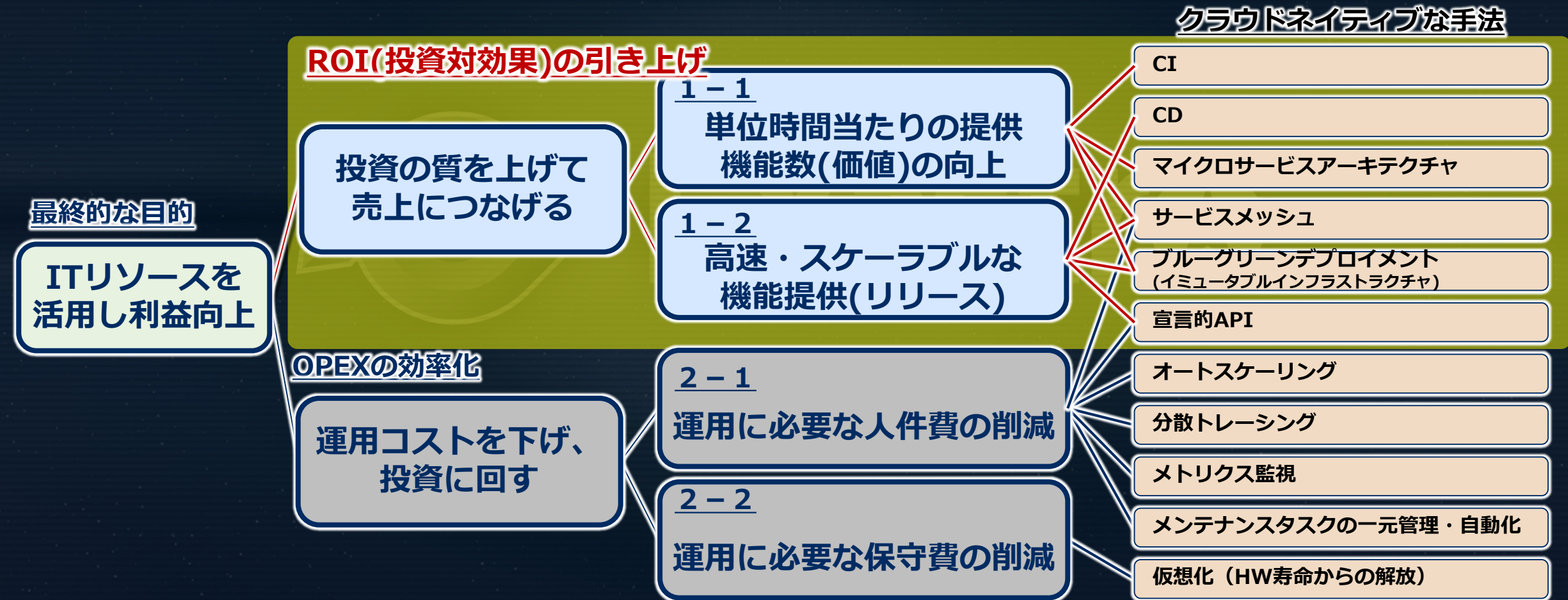
【攻めの自動化】
ROI(投資対効果)を引き上げるために必要な
クラウドネイティブのテクニック



ITリソースを活用し利益を向上したいという最終的な目標のために 様々な「クラウドネイティブな手法」が考えられています

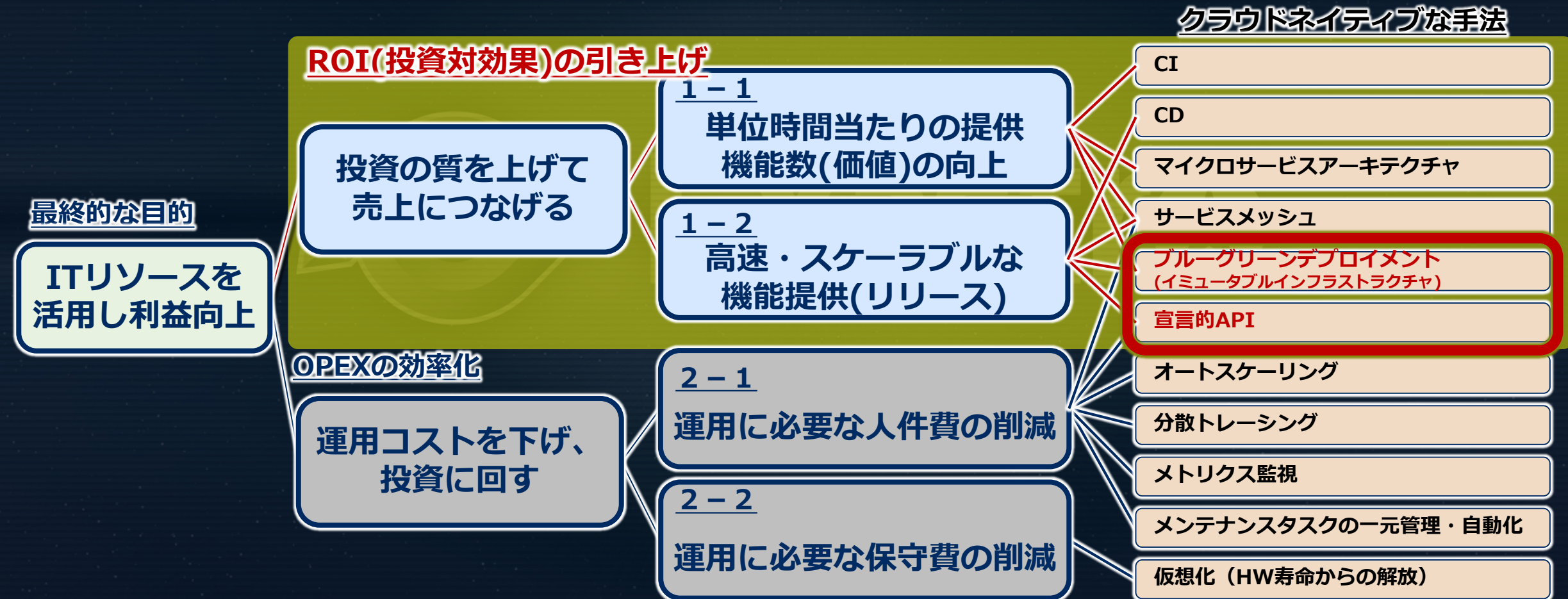


そのうち下表の「CI/CD」～「宣言的API」といった手法では
ROI(投資対効果)の引き上げを見込むことができます



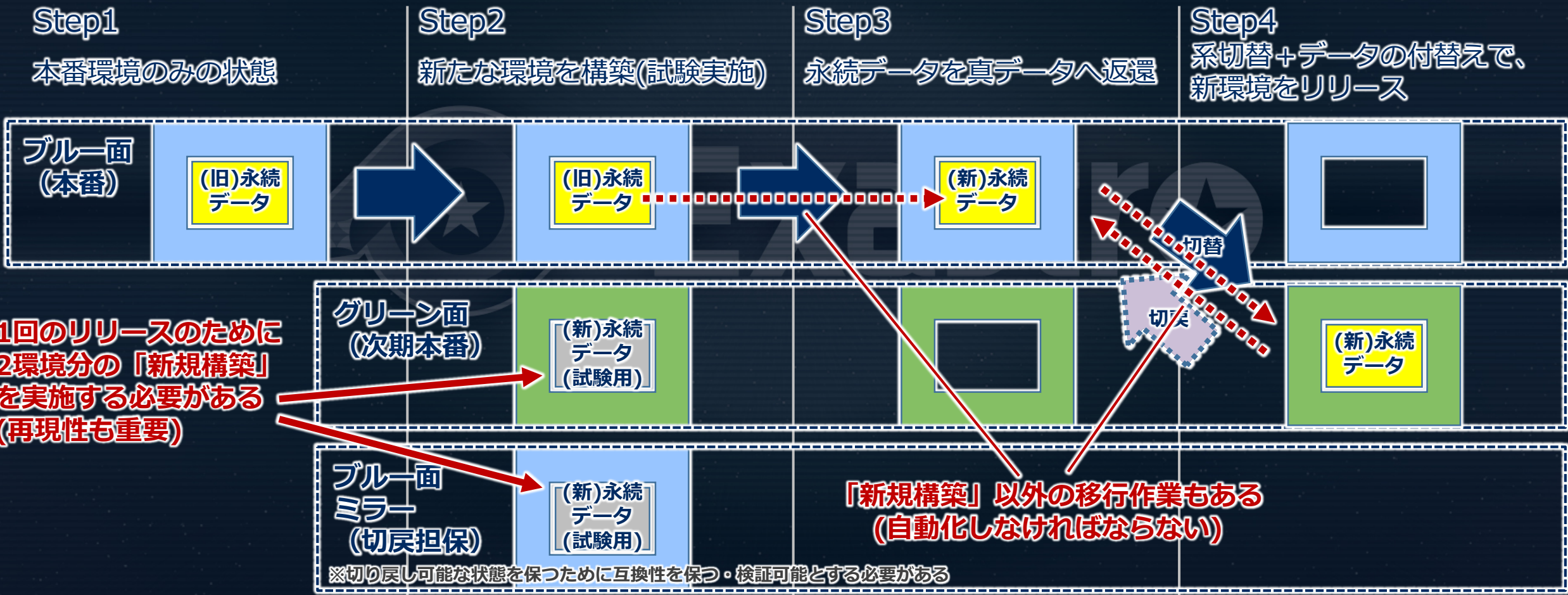
ROI(投資対効果)を引き上げるために必要なクラウドネイティブのテクニック

さらに「**ブルーグリーンデプロイメント**」と「**宣言的API**」は自動化と密接な関係があり、「**攻めの自動化**」に不可欠な要素です



ROI(投資対効果)を引き上げるために必要なクラウドネイティブのテクニック

**ブルーグリーンデプロイメントでは部分的に「新規構築」を繰り返します
手作業でやっていたのでは作業量が爆発しますので自動構築は必須です**



「新規構築」可能な部分は**宣言的API(あるべき姿を定義)**が効果的です
一方で、幾許か残る移行作業には**命令型IaCでの自動化**が必要です

宣言型のIaC (宣言的API)

最終的に以下の通り。

- 〇〇が1個
- が2個
- △△が3個

クラウドリソースの
新規構築に向いている



命令型のIaC (手続き的)

1. まず〇〇を1個用意する
2. 次にAとBを混ぜて△△を3個作る
3. 最後にCとDを混ぜて□□を2個作る

クラウドリソース以外は
命令型IaCで作業を自動化
する必要がある

 **Exastro** は宣言型と命令型の両IaCを駆使した自動化に役立ちます

まとめ：「攻めの自動化」を実現するうえで重要なポイント

I. 作業量に糸目をかけない。

そのためにあらゆる作業は例外なく自動化する。

(短いインターバルで何度でも「新規構築」を可能とする)

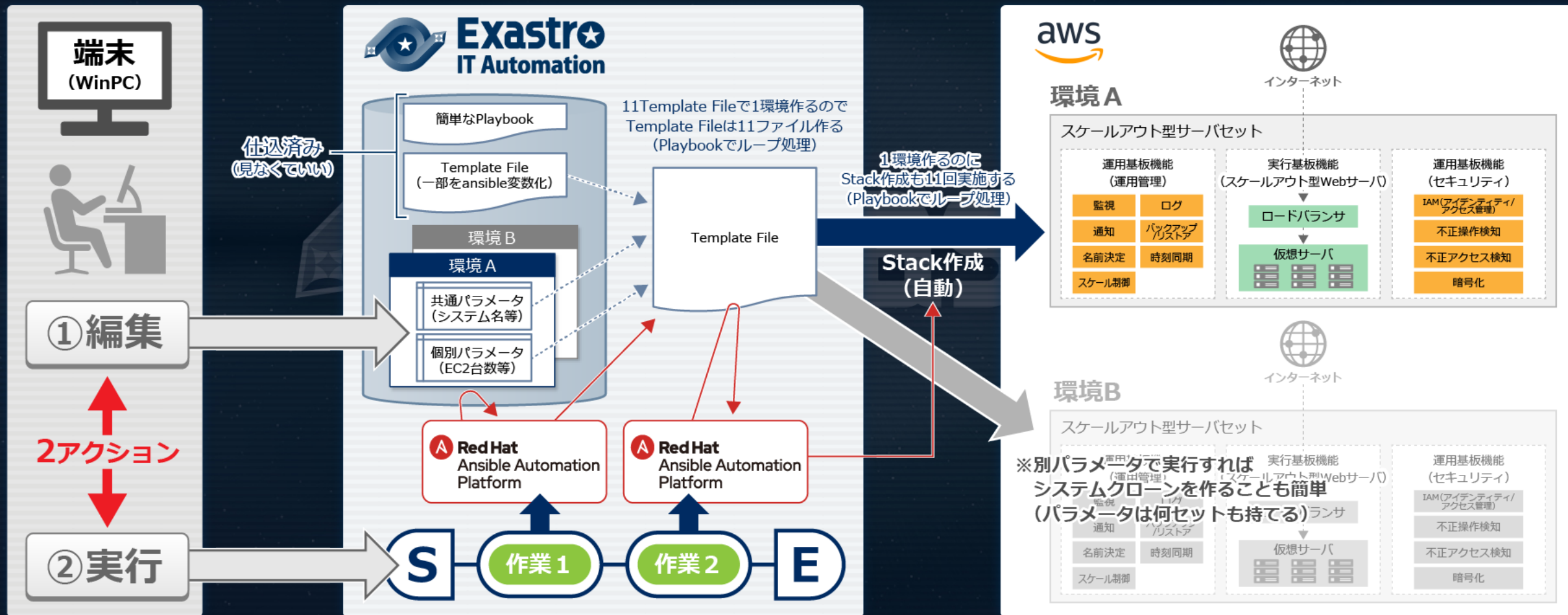
II. 宣言的APIはとっても重要。

ただし、それだけではあらゆる作業を自動化できない。

クラウドネイティブでも命令型IaCの必要性を忘れない。

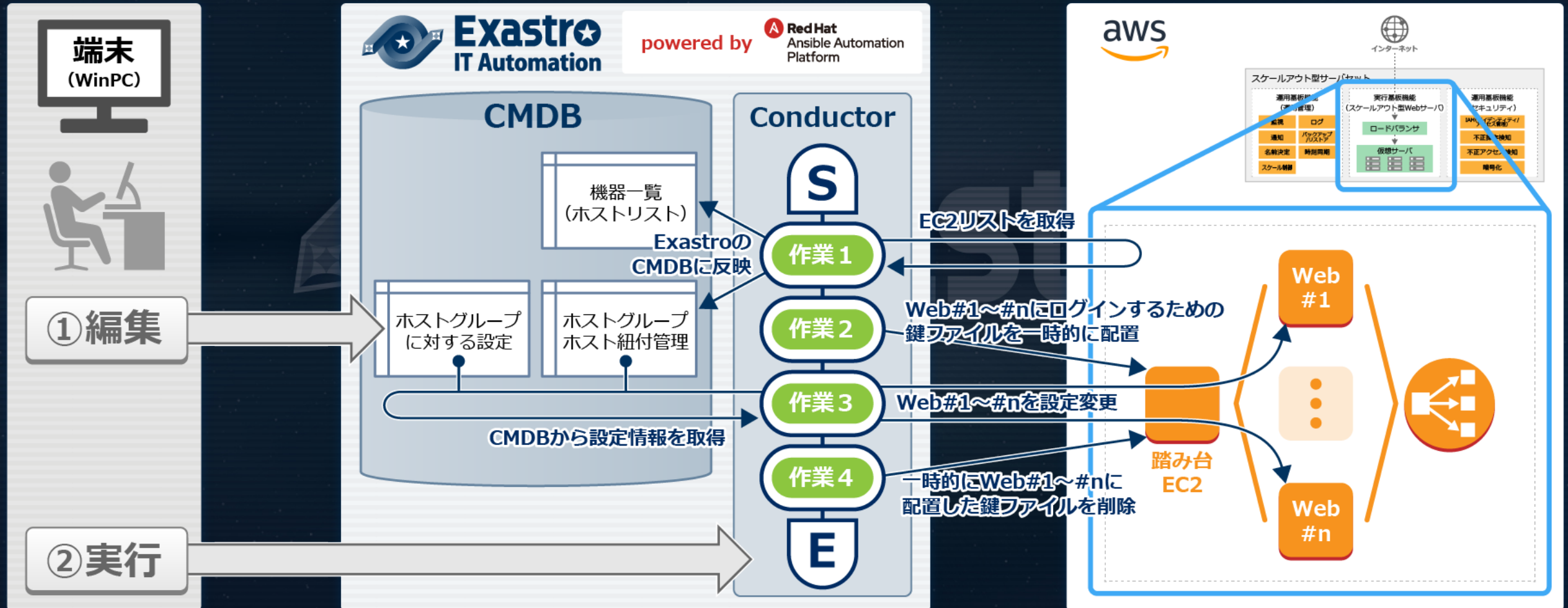
【事例】企業の運用部門が各部門のAWSシステムを統制する仕組み (1/2)

Exastro IT Automationで「CloudFormationテンプレート」を管理し、各部門にガバナンスの効いたAWS環境を払い出す仕組みを提供しました。



【事例】企業の運用部門が各部門のAWSシステムを統制する仕組み (2/2)

稼働中のEC2(オートスケール)に緊急でパッチ適用する、
といった運用シナリオにも対応しました





Exastro