

キャパシティ管理による システムレジリエンスとビジネス改善

～パフォーマンスデータからの気づき、活用～

はじめに



製品 導入企業一覧 導入事例 セミナー コラム 企業情報 採用情報

資料ダウンロード

お問い合わせ



パフォーマンス管理セミナー

メインフレームからオープンシステム、アプリケーションまで、各リソースのパフォーマンス管理のポイントをご紹介します。システムパフォーマンスの全体を学びたい方、システムに興味のある方ならどなたでもご受講いただけます。また、配属1年目の方向けにアレンジした2日間の研修コースもご用意しています。

セミナーの特長

- ・社内では困難なパフォーマンス管理者の育成をサポート！
- ・メインフレームやオープンシステム、アプリケーションなどの性能管理に必要な知識をご紹介します！
- ・リソースからデータベース、ネットワークまで幅広い領域を網羅！

対象となる方

- ・システムパフォーマンス全体を学びたい方
- ・システムに興味のある方

ITILをベースとした「キャパシティ管理入門」セミナーを通じて、キャパシティ管理のサブプロセスである以下の分野を中心に、コスト管理の視点も交えた内容を提供してまいりました。

- ・ビジネスキャパシティ管理（BCM）
- ・サービスキャパシティ管理（SCM）
- ・コンポーネントキャパシティ管理（CCM）

今回、このセミナー内容を刷新し、システムのレジリエンス強化に焦点を当てるとともに、キャパシティ管理と密接に関連する「インシデント管理」「問題管理」「変更管理」「リリース管理」などの重要分野を加えた、より包括的な内容にアップデートいたしました。

本日は、この改訂版セミナーの一部を抜粋し、ミニセミナーとしてご紹介いたしますので、2025年2月に開催予定の本セミナーにも、ぜひご参加頂けますと幸いです。

運用管理フレームワークITILの変遷

ITIL v4: デジタル時代に対応

- ・ 企業活動とITを分けずにとらえる
- ・ 組み換え可能なシステム構成 (Short Value listS)
- ・ 会社全体で共創、常に変化させていく

Agile DevOps
Microservice
CloudNativeの世界

ITIL v3: ライフサイクル観点で再編

- ・ ビジネス要求をITで解決する
- ・ ITにかかわるすべての人を対象
- ・ 正常稼働ではなく、提供価値に注目

ITIL v2: プロセス単位での活動

- ・ 業務フローが定義、導入しやすい
- ・ 個々のプロセスの 効率性、有効性
- ・ 運用現場中心のフレームワーク

Waterfall
Monolithic
Datacenter / On-premises

ユーザ要望と実現プロセスの変化

ユーザ要望 = うまく動くこと

重

ITIL

Design for Robustness

堅牢性: 壊れないことを前提にシステム構築

DevOps

Design for Failure

故障前提: 壊れることを前提にシステム構築

軽

SRE

Design for Resiliency

回復性: 故障時の復旧を見越してシステム構築(アプリ開発)

IT運用の重要性向上と安定運用の難しさ



ビジネス部門

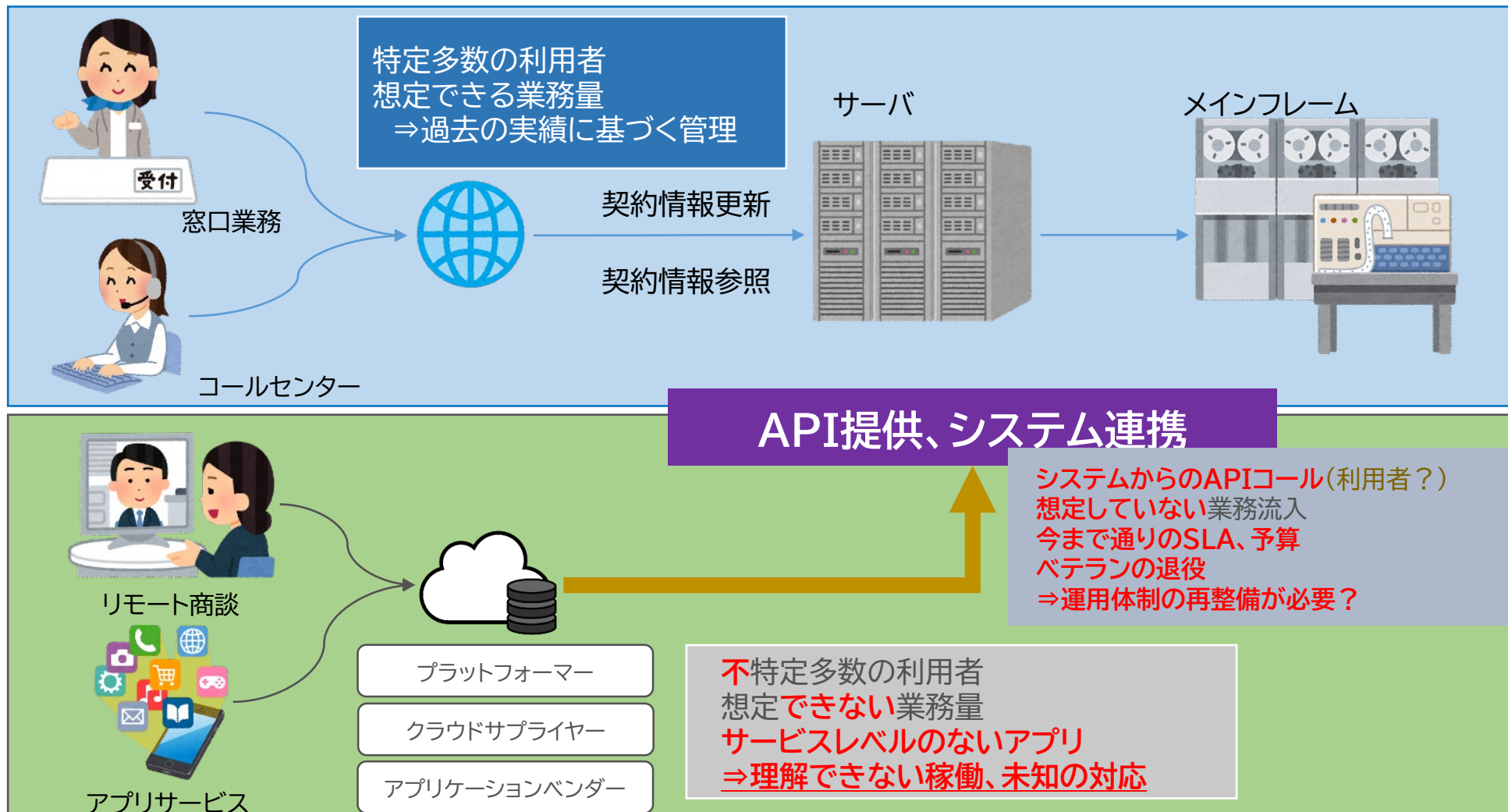
DX*

CX**

xxtech

コロナ対策

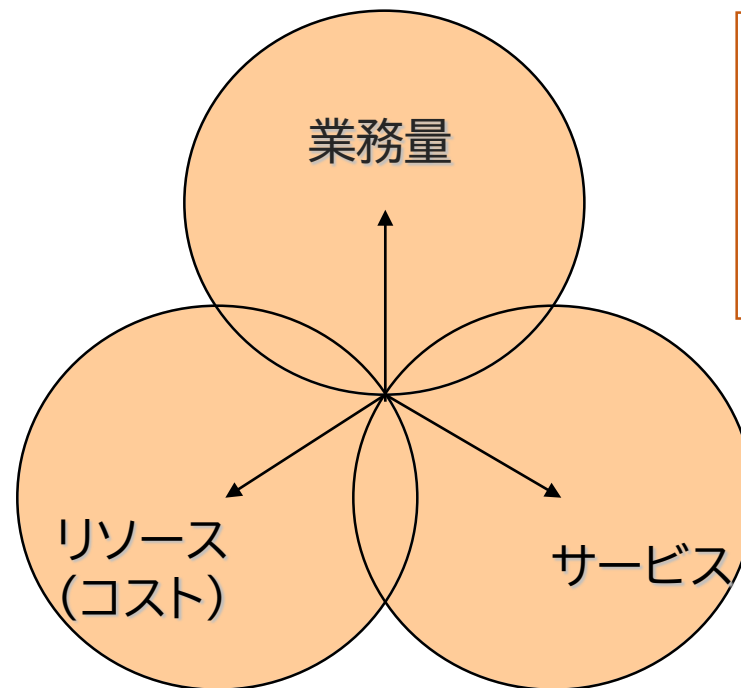
* Digital Transformation
** Customer Experience



改めて性能管理を考え直してみた

投入される業務量が
ダイナミックに変動する
業務量計測がビジネス成功
コスト適正化のカギとなる

リソース調達コストは減
調達スピードは高速化
⇒リソース管理の価値は
相対的に低下



システムの正常稼働を
示す指標として
サービスの計測、可視化が
重要性を増す

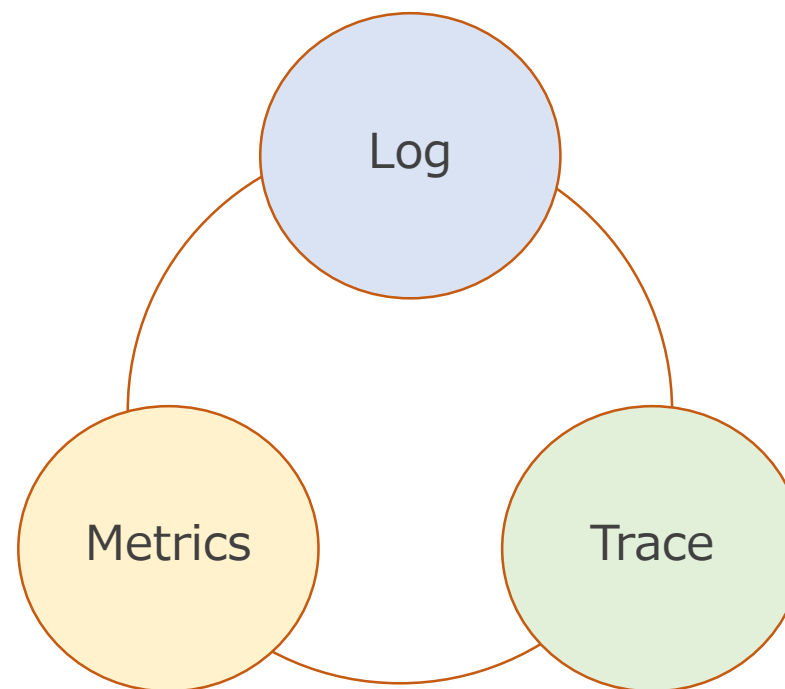
業務量のみが外的要因によって変化する値であり、可視化が必須
サービスの安定化は運用におけるKPI

オブザーバビリティ

- ログ
 - OS・ミドルウェア・アプリケーションから出力されるテキストデータ
 - 実装が簡単で、何が起こったかを把握しやすい
- メトリクス
 - CPU・メモリや応答時間・トランザクション数など、数値的なデータ
 - 閾値やトレンド分析に利用しやすい
- トレース
 - 処理のプロセスを可視化したデータ
 - どこで問題が起こったかを把握しやすい

これら全てを可視化できる状態にすることが重要

オブザーバビリティとは「**システムの状態を観測する** (いつ・どこで・何が起きているかが分かる)能力」
これは以下の3つの要素によって構成されます。



かわりつつある監視～サービス監視の重要性

The USE Method

Utilization

使用率:リソースが処理中でbusyだった時間の平均

Saturation

リソースの飽和、ひっ迫、枯渇

Errors

エラー:エラーイベントの数

Brendan D. Gregg氏によって提唱されたパフォーマンスの問題分析を目的とした取得すべきデータ項目や分析手順などのベストプラクティス



Four Golden Signal

Latency:通信遅延時間

Taffic:通信量、データ送受信量

Errors:リクエストの失敗数

Saturation:リソースの飽和、ひっ迫、枯渇

O'REILLY サイトリライアビリティエンジニアリング

The RED Method

Rate

レート:秒あたりのリクエスト数

Errors

エラー:リクエストの失敗数

Duration

時間:リクエストの処理にかかる時間

元GoogleのSREのTom Wilkie氏によって提唱された監視のベストプラクティス

可用性、MTTRの世界

ユーザ視点サービスの世界

SLA達成のための実践方法

SLA

Service Level Agreement

SLAは提供者と対価を払う利用者との間の契約
※保障するもの

SLOs

Service Level Objectives

SLOはサービスレベル目標。保障するものではなく目標のためSLAよりも厳しめな値を設定し、チーム内での改善を促すトリガーとする

SLIs

Service Level Indicators

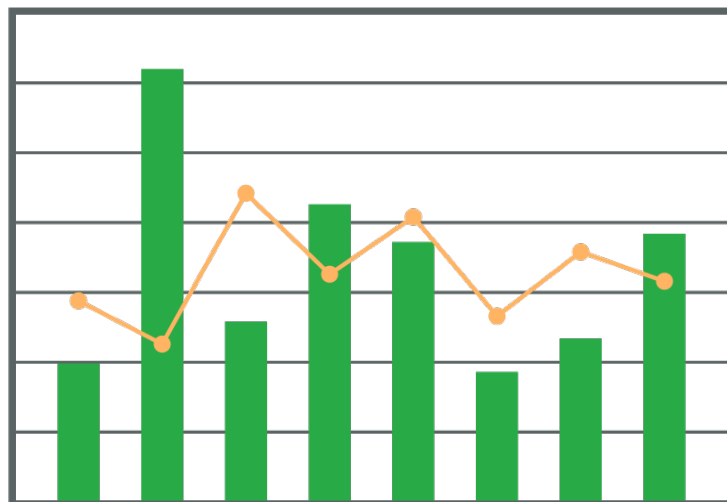
上記のサービスレベル目標を定量的に計測し、目標値達成を判断するための指標

TIPS

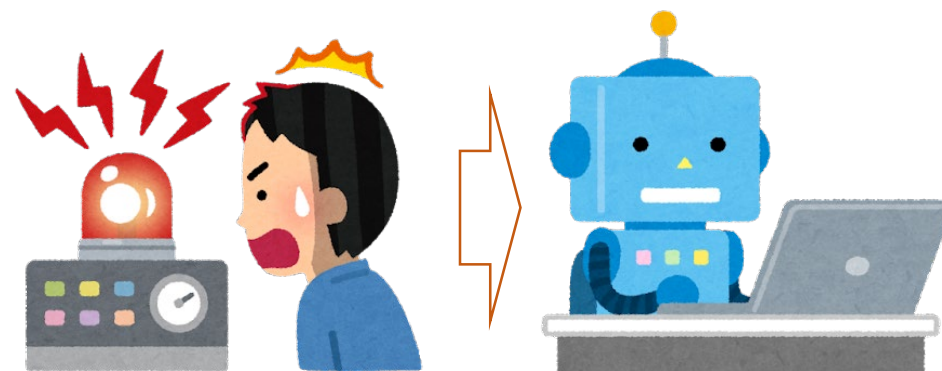
- ・わかりやすい言葉で、顧客の要望に沿ったSLAを作成する ÷ 項目を多くしない
- ・達成困難なSLAにしない。(困難に挑戦しない+エラー予算を確保する)

予防保全も、リアルタイムも

傾向分析による
予防保全

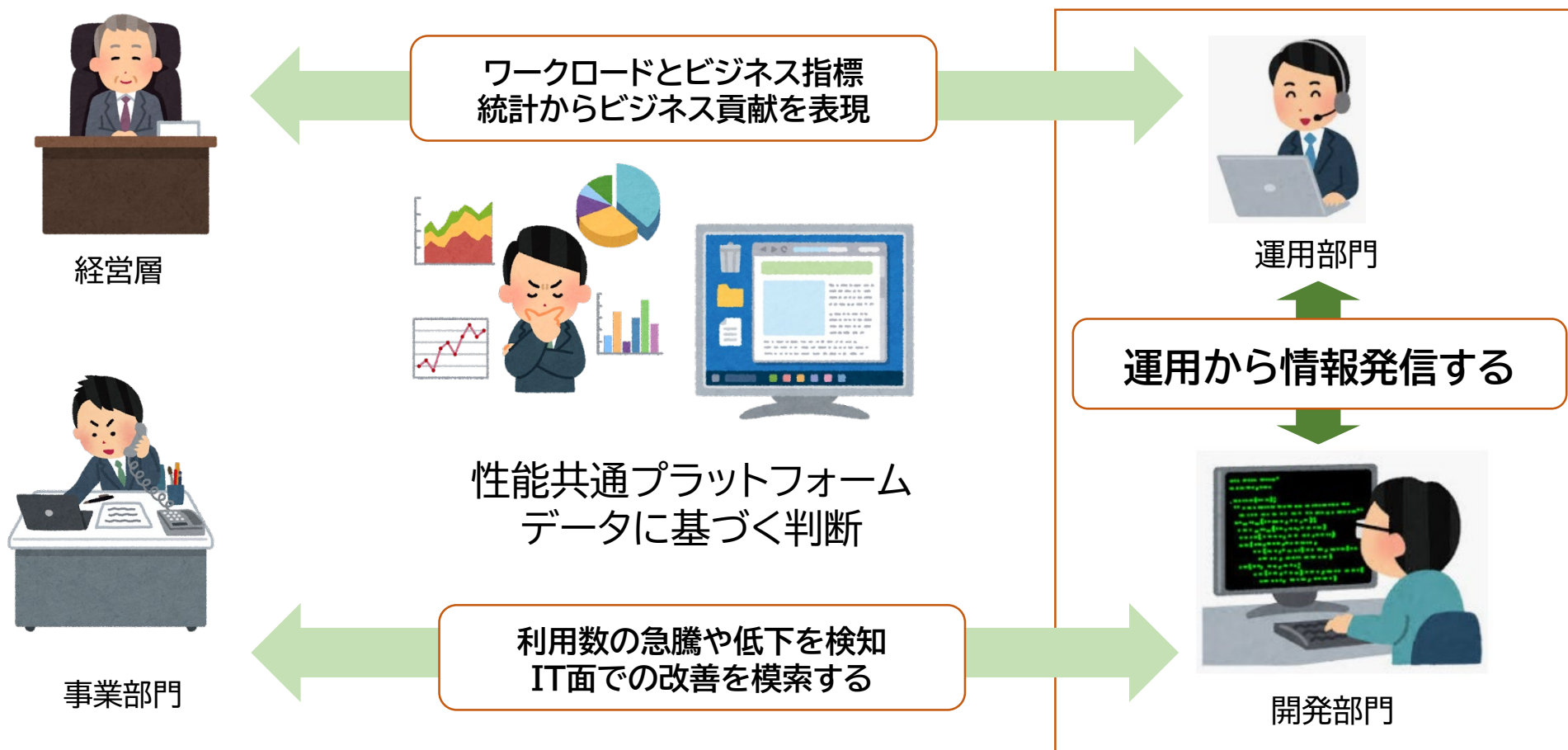


リアルタイム
検知による
即時回復



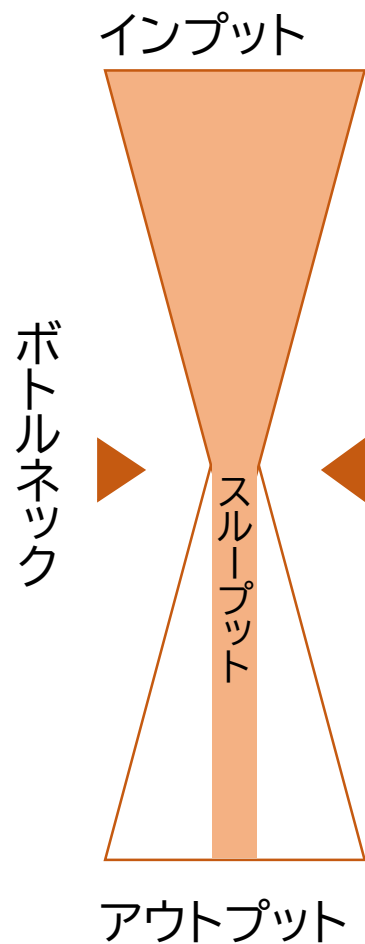
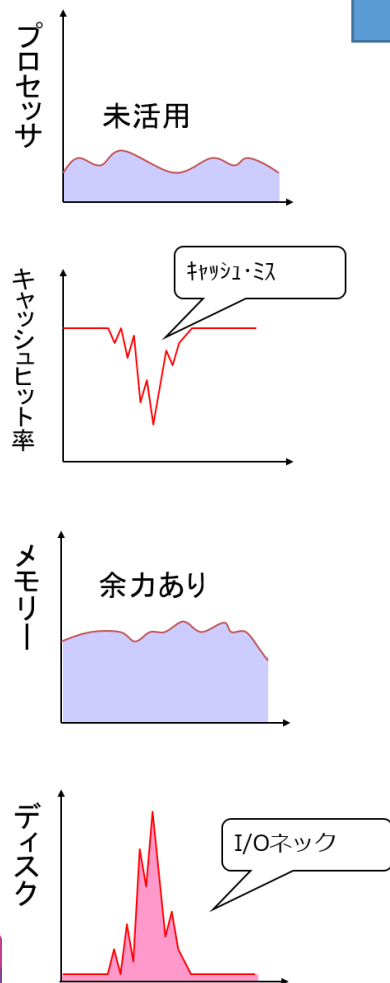
業務量変動は有効なビジネス指標になりえる

IT部門が取得したデータから経営判断につなげる



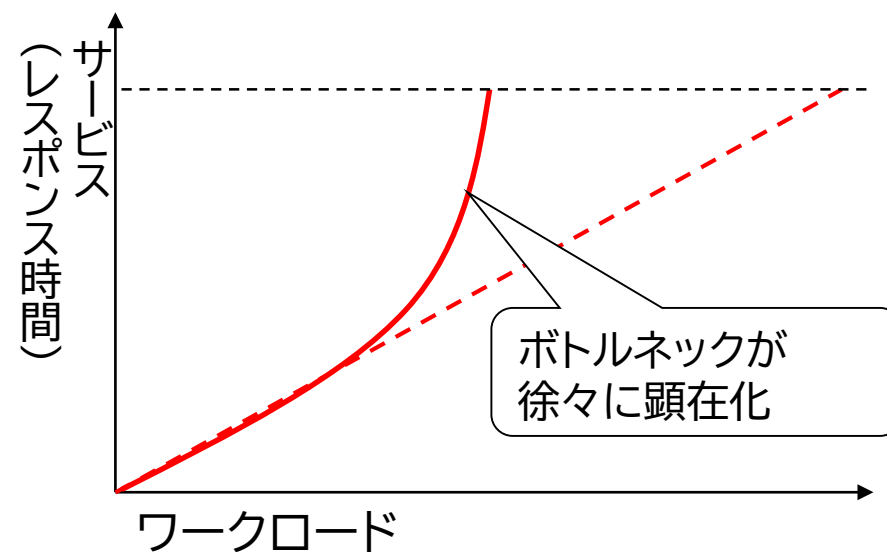
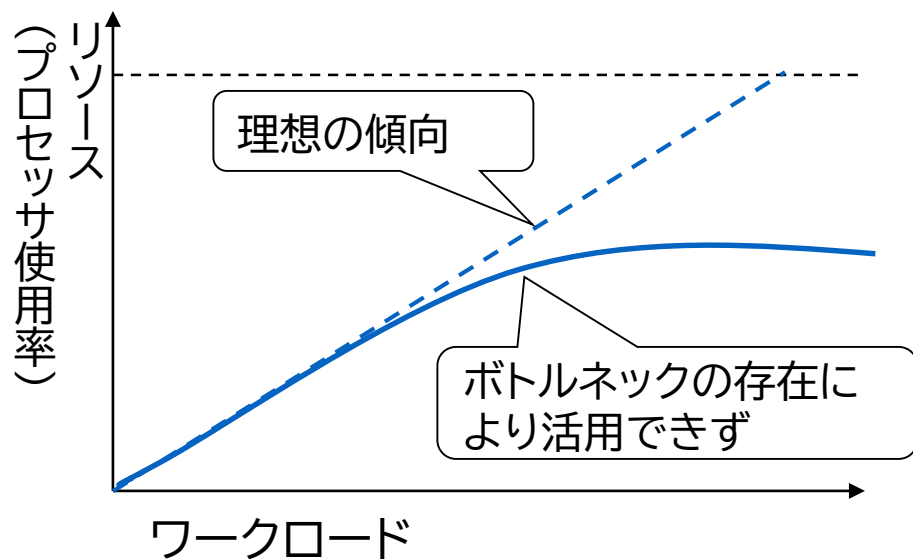
ボトルネックは必ずどこかに発生する

TOC(theory of Constraints)理論



ボトルネックがあると予測もうまくいかない

- ディスク性能にボトルネックを抱えている場合
 - 業務量の増加に伴い、さらにディスクに処理が集中
 - ディスク処理が完了しないためプロセッサを活用できない
 - プロセッサに余裕があるように見えるがサービスは低下
 - このような状況下では、プロセッサのサイジングは検討できない
 - サイジングは業務量に対してリソースが線形に増加する前提



時代とともに変わるリソース調達方法

いままで

- ・ハードウェアの調達には(数か月単位の)時間がかかる
- ・リソースは高価で途中での追加には大きな工数がかかる
- ・稼働期間中の構成変更は行わない(例外的にサーバ追加での補強はある)

⇒ 上記の理由により、資源計画／キャパシティプランニングが重要

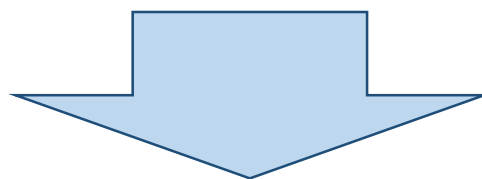
現在↓これから

- ・クラウド、仮想基盤ではリソースの調達は瞬間的にできる
- ・リソースの追加、削減は動的に変更可能
(クラウドでは従量課金なので使わなければコスト削減につながる)
- ・稼働期間中の変更も柔軟に対応できるように抽象化が進んでいる
(OSやパッケージのバージョンからの解放案としてのコンテナやSaaSの隆盛)

⇒ 稼働データから必要量の設定をするほうが簡単、且つ正しい設定がしやすい

システムレジリエンスを高めるために

- システムのレジリエンスを高めるためには以下の要素が不可欠
 - 意味のあるアラート(アラートストームの抑止)
 - インシデント管理、問題管理、変更管理、リリース管理プロセスの高度化
 - ナレッジの活用
 - オートヒーリング



事象の早期発見や正しい原因の切り分けには、
オブザーバビリティの観点による計測データの高度化が必須

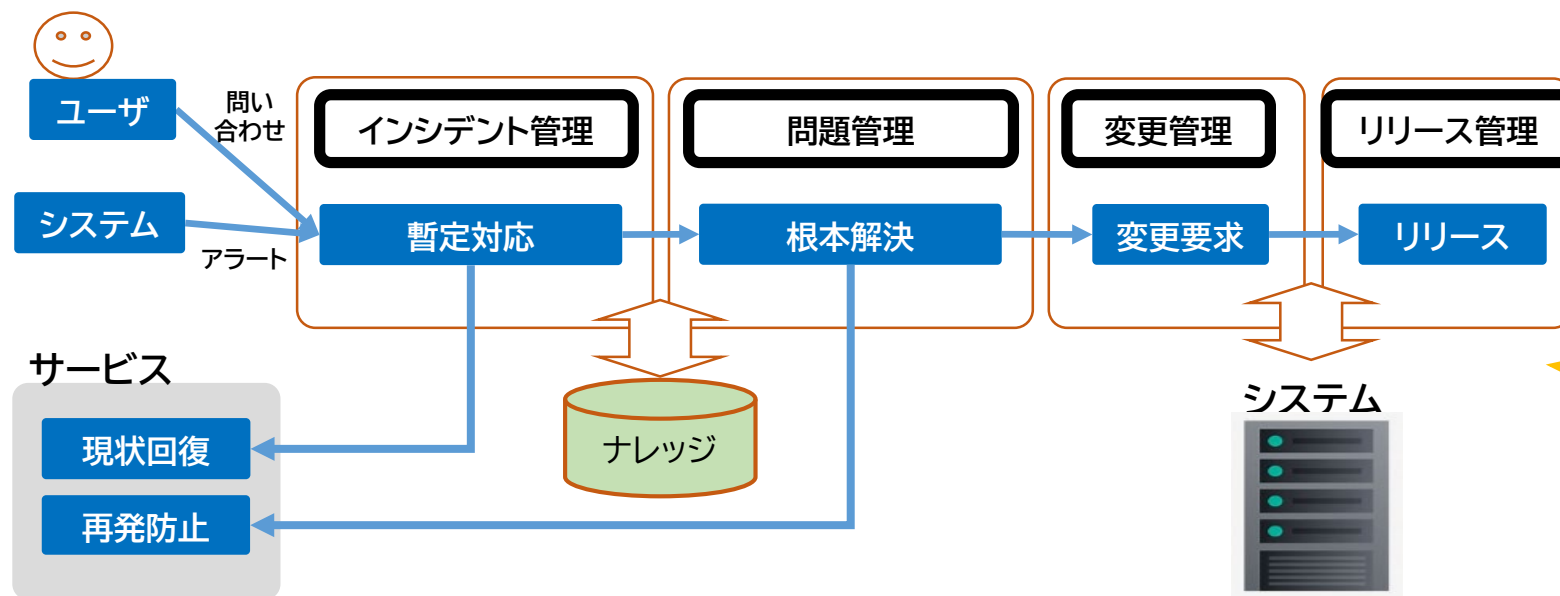
インシデント管理、問題管理、変更管理、リリース管理の関わり

インシデント管理:

サービスの正常な運用を妨げる全てのインシデントに対して迅速かつ効果的に対応するためのプロセス。インシデントとは、計画された品質のサービス提供を妨げるまたは削減するイベント。= 暫定対応・現状回復を優先

問題管理:

サービスの中断や劣化を防ぐために、問題の原因を特定し、それを排除または回避する方法を見つけること。= 根本原因解決・再発防止

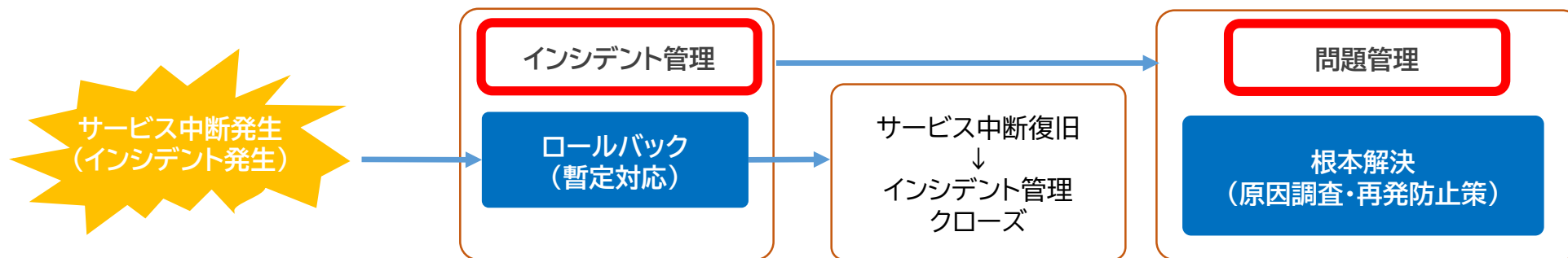


動かないだけでなく、
遅いこともインシデント！

根本解決は問題管理によって
なされるが、そのための設定変更
やバージョンアップは変更管理、
リリース管理と連動してリスクを
抑えつつ品質を保証

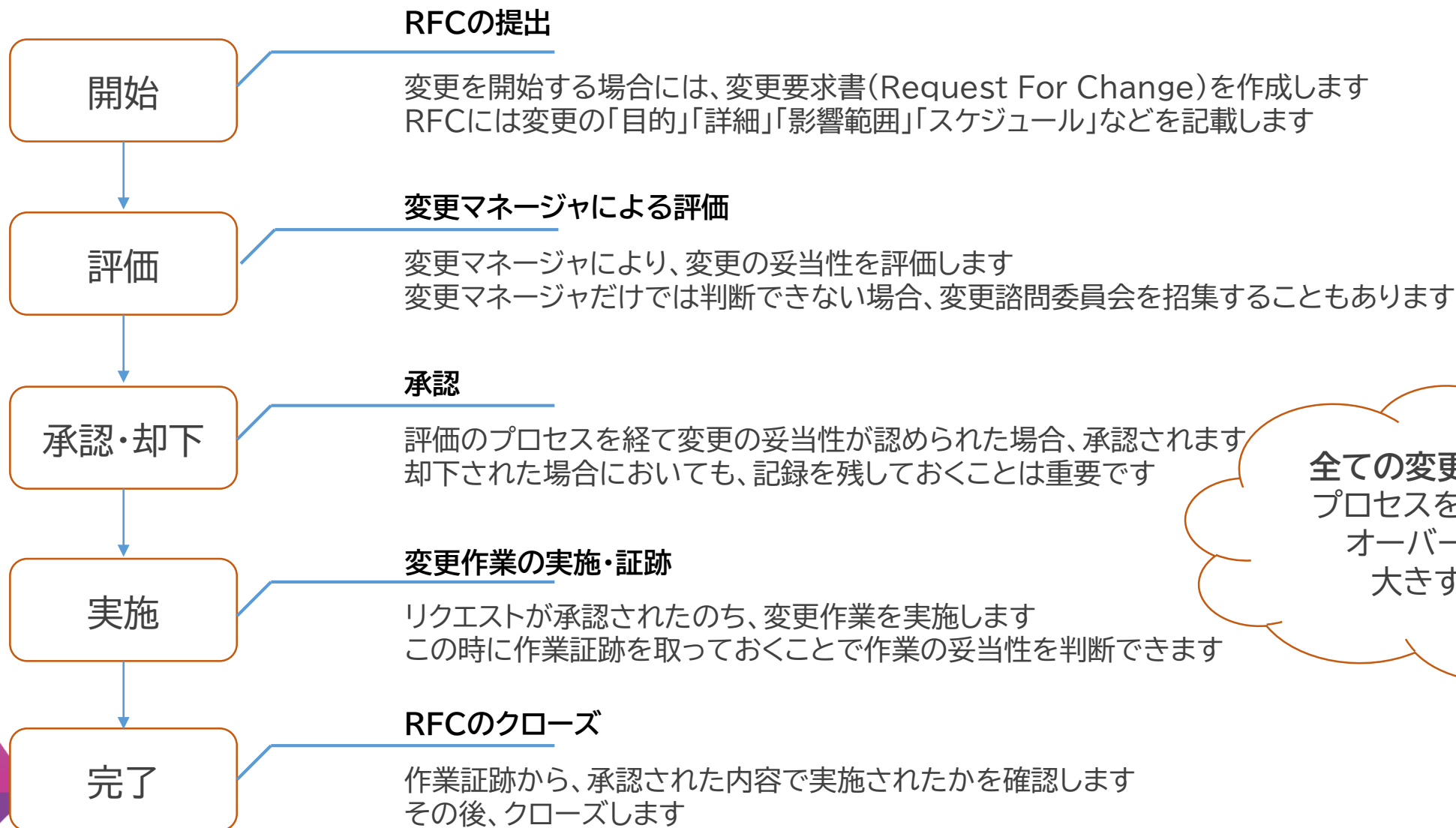
インシデント管理と問題管理のつながり

- 問題からは、1 つ以上のインシデントが発生する場合がある
…1つの問題管理に、複数のインシデント管理が関連する場合がある
- インシデント管理と問題管理の違い



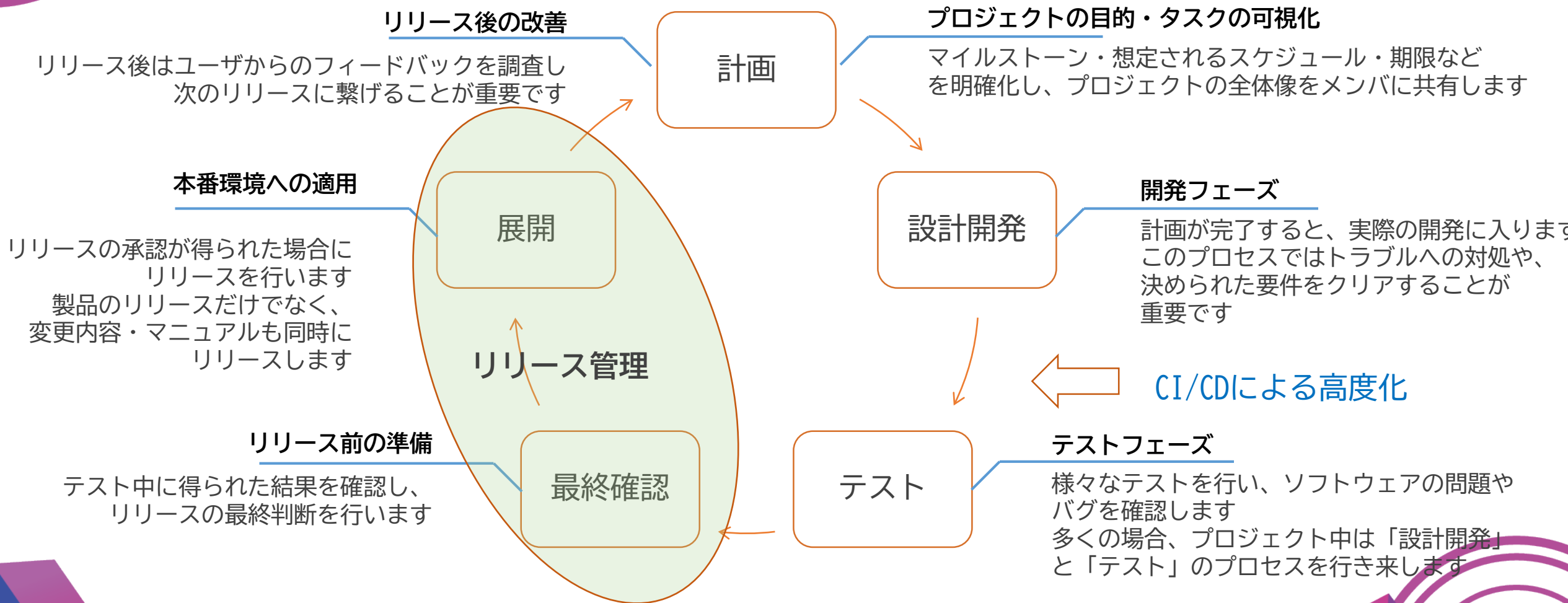
	期間	主な目的
インシデント管理	短期 即時性が重要	インシデントを解決し、サービスを元の状態に戻すこと
問題管理	長期 問題が二度と起こらないようにする	インシデントの下に潜むもの・インシデントの発生理由を明確にし、再発防止策を特定する

変更管理の大まかなプロセス



全ての変更これらの
プロセスを適用すると
オーバーヘッドが
大きすぎる…

リリースまでの一連の流れ





今後に向けて

- 本日は時代とともに変化するキャパシティ管理を含めた性能に関する運用の在り方を中心にご説明させていただきました。
- 2024年2月開催予定の本セミナーでは、実践に向けたアプローチも含め、より詳細にご説明いたします。皆様のご参加を心よりお待ちしております。



株式会社 アイ・アイ・エム