

P A R - 0 7

# 大規模生体認証システム **NEC I:Delight** で見えた！ マネージドサービスに甘えない 信頼性の作り方

奥山 嘉昭  
日本電気株式会社  
生体認証・映像分析統括部  
シニアプロフェッショナル

大竹 孝昌  
日本電気株式会社  
サービスプラットフォーム統括部  
シニアプロフェッショナル



# 日本初のコーポレートレベルの戦略的協業を米国AWSと締結

(2020年11月プレスリリース : [https://jpn.nec.com/press/202011/20201113\\_01.html](https://jpn.nec.com/press/202011/20201113_01.html))

世界トップレベルの品質を官公庁を含む全てのお客様に最適な形で提供します

## 戦略協業を軸とした3つの活動



NECとAWS社の強力タッグで  
行政の在り方そのものを変革する  
「デジタル・ガバメント」への取り組み

官公庁・金融・医療機関など、業種特有の  
レギュレーションに対応したマネージドサービスを提供

お客様のデジタルジャーニーをサポートする  
オファリングメニューを共同開発

NECグループのAWS認定資格保有者を倍増し  
デリバリーモードを強化

2021年9月、グローバル5Gやデジタル・ガバメントなどの領域で協業をさらに拡大

[https://jpn.nec.com/press/202109/20210908\\_01.html](https://jpn.nec.com/press/202109/20210908_01.html)

# AWS活用したNECの生体認証サービス構築とソリューション事例

日本電気株式会社  
生体認証・映像分析統括部  
奥山 嘉昭

# 登壇者ご紹介

奥山 嘉昭(おくやま よしあき)

NEC 生体認証・映像分析統括部 シニアプロフェッショナル

生体認証・映像分析ソリューション推進グループの責任者。主な職務内容は、生体認証・映像分析システムの他社への提案、ソリューション構築。2021年まではSaaSサービスである生体認証・映像分析サービスの開発をプロジェクトマネージャーとして統括した。

# 本日のアジェンダ

1. 生体認証の社会的意義
2. 生体認証のサービス化
3. 生体認証のSaaS実現に立ちふさがった問題
4. NECにおけるデジタル変革の実践
5. NEC Digital Platform 他社適応事例

# \Orchestrating a brighter world

NECは、安全・安心・公平・効率という社会価値を創造し、  
誰もが人間性を十分に発揮できる持続可能な社会の実現を目指します。

# 1. 生体認証の社会的意義

# 社会の変化と生体認証の発展

## 2001 安全な社会を実現するために



国ごとの  
対策  
入国管理

- ・顔写真が世界中のパスポートのICチップに搭載
- ・顔認証の認知・社会的受容性が高まる

## 2020 社会・経済活動を止めないために



感染症防止  
両立  
社会持続性

- ・非接触が新たな価値として登場
- ・衛生面に優れた生体認証への期待が向上

# 生体認証SaaSに期待する諸元

価値検証では安価・スピーディに提供でき、社会・企業の基盤化の際は高性能・高信頼性を担保できることを両立できるサービスが必須。マルチテナントで収容することで保守コストを下げ、かつ各案件毎に可能な限りスケールできるサービスにする。

	生体認証SaaS	旅客機搭乗	入国審査	大企業ビル入退
処理性能	最大は1000TPS想定	30TPS	150TPS	10TPS
照合対象	1000万人	5万人	15万人	1万
照合規模 (1:N)	1:100万  公共系案件と比較しても、マルチテナントのサービスにすると要求性能は群を抜き高い	1:500人 ・搭乗人数単位で実施	1:5万人 ・通過後に削除	1:1万人 ・入館人数に合わせて永続的
停止時間	無停止 ・様々な要求に応える	サービス稼働時間	無停止	夜間などメンテナンス停止可能

ソリューション毎にオンプレで提供していたものを一元化し、  
安価で使いやすいマルチテナントサービスとして提供したい  
そもそも作れるのか??

## 2. 生体認証のサービス化

# OSSベースで作られたシステムのAWSへの移行

内製構造を変更せず、移行・維持コストを最小化し、可用性を向上するため、AWSを活用

## 移行コスト・期間最小化

⇒生体認証周りの内製コードが大きく、APIが異なるサービスへの移行が難しい。APIベースで変わらないマネージドサービスを選択し、移行コストと期間を最小化する

## 維持コスト低減

⇒OSS保守要員、OSSサポートの購入を削減。内製コードの開発に集中できるようにする。

## 可用性の向上

⇒可用性のためのAWSのマネージドサービスを使い多重化、バックアップ検討。可用性設計、構築のコストを削減。

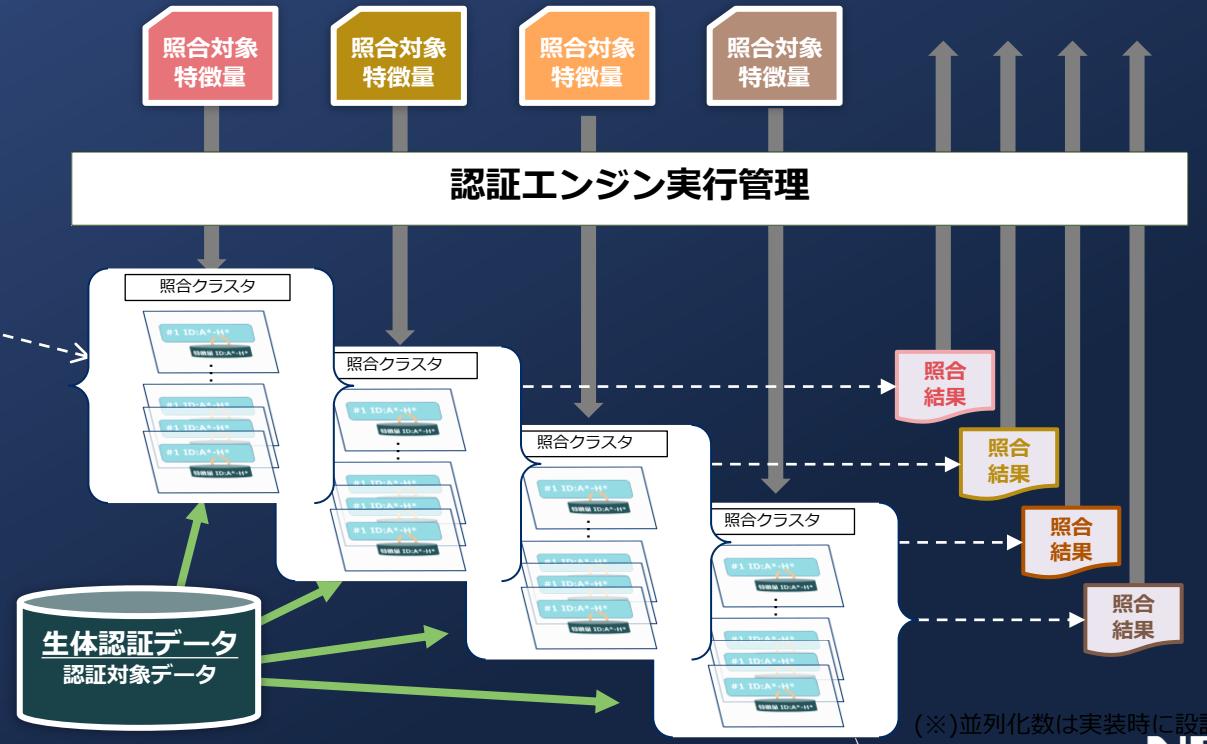


# 生体認証の照合処理の分散実行

個人個人に応じたサービスを様々な利用シーンで活用するためには、多数の生体認証要求の処理が必要。照合速度の向上のため照合クラスタを複数起動して、認証処理の多重化を図る。

## 分散実行

照合処理を並列起動することで多数の要求を分散実行可能とする

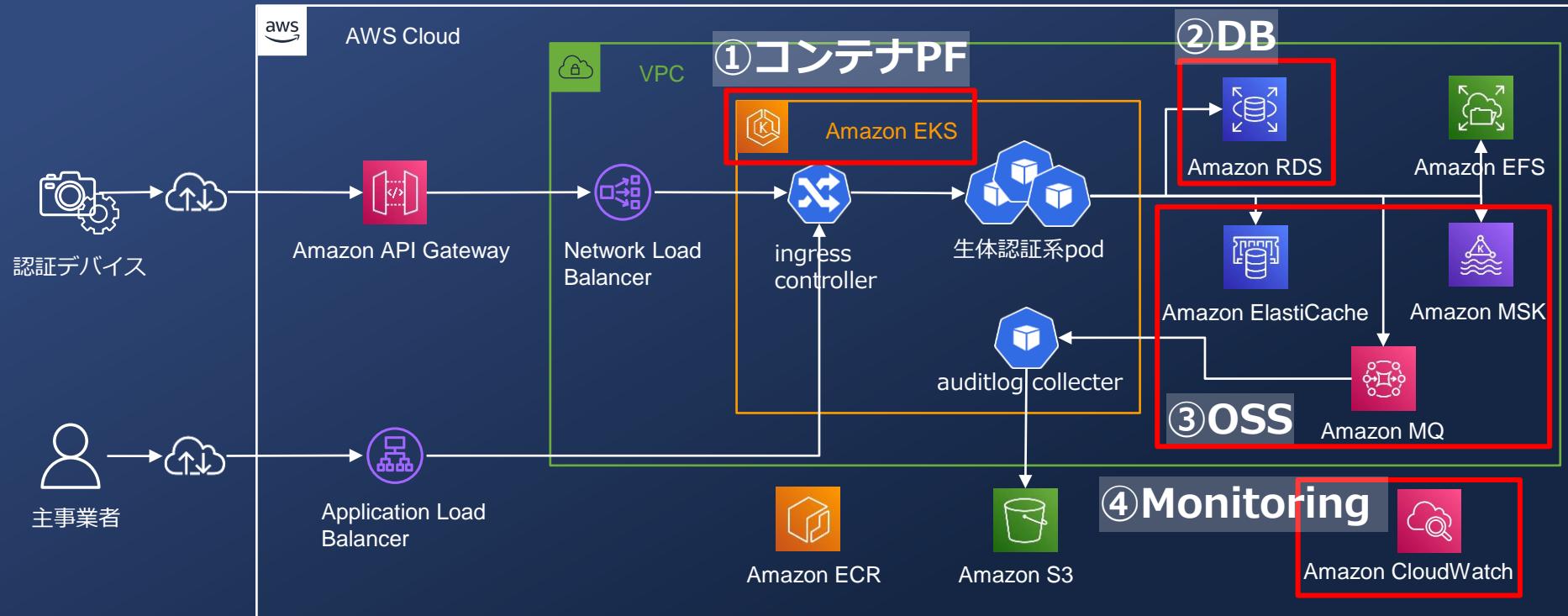


# ベアメタルからAWSへ

高性能・高信頼の生体認証サービスを低価格で実現するためAWSを選択。

コンテナ実装による生体認証処理の並列化で高い性能を獲得。

下回りの機能をAWSマネージドサービス群に移行し、OSS保守コストの削減と高可用なシステムを実現。

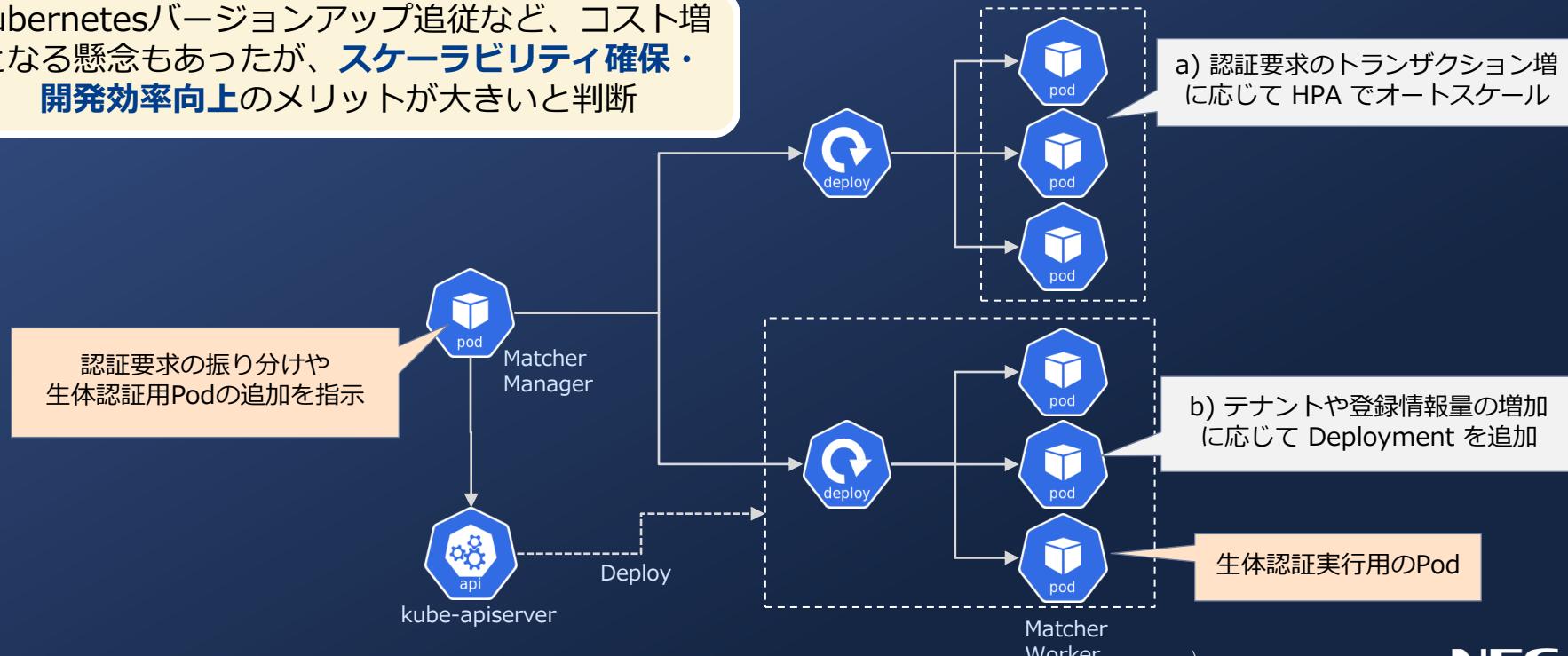


- 
- ①コンテナPF
  - ②DB
  - ③OSS
  - ④Monitoring

# 生体認証コンテナのスケールアウト

内製で行っていたスケールアウトをKubernetesに移譲して開発コストを低減。登録情報量増(メモリ使用増)、トランザクション増(CPU負荷増)の両者を考慮したスケールアウトを実現。

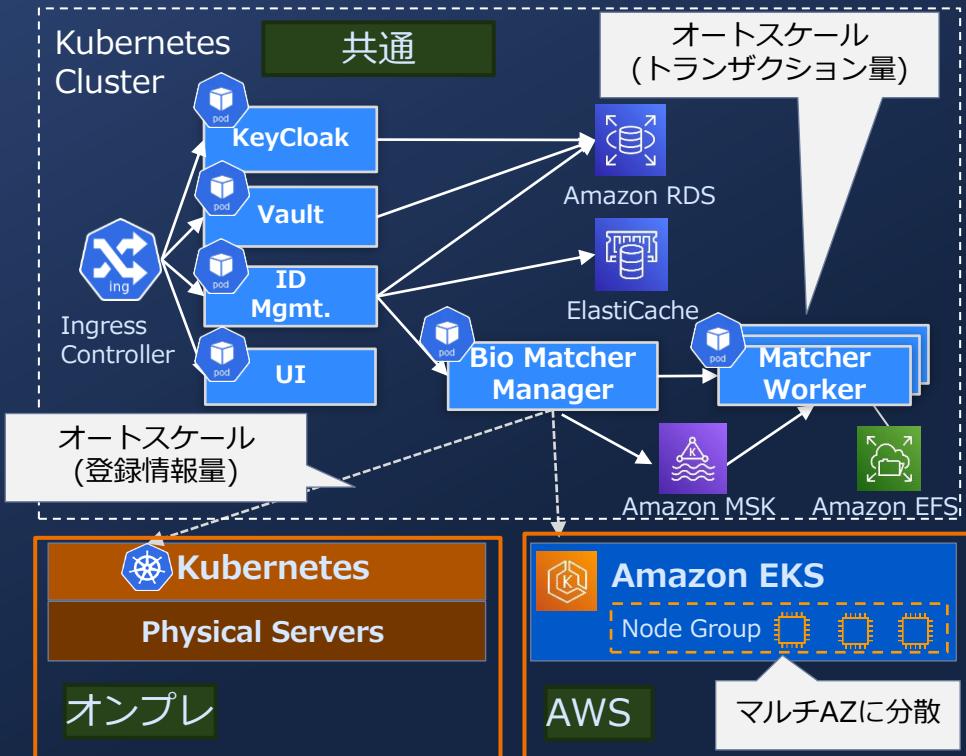
Kubernetesバージョンアップ追従など、コスト増となる懸念もあったが、**スケーラビリティ確保・開発効率向上**のメリットが大きいと判断



# コンテナPF(EKS)の選定方針

オンプレ案件・クラウド案件でPF共通化を図るためにEKSを選定。  
高可用性、高性能の実現に向け、マルチAZ・オートスケールを活用。

- ・ オンプレ対応も必要  
⇒ オンプレ・クラウドでPF共通化のため  
**Kubernetes / Amazon EKS採用**  
※ 生体認証は大量のvCPUを使用するため  
vCPU数制限があるAWS Fargateは除外
- ・ 入退場・決済システムなどで使用するため  
**高いSLAが求められる**  
⇒ マルチAZを使用し可用性を高める



# EKSワーカーノードのセキュリティの強化

生体情報など重要な個人情報を取り扱うため、Podを実行するEC2に対して徹底的なセキュリティ対策を実施。

生体認証では**個人情報を取り扱うため**、

通常より**高いセキュリティが必要**

EC2へ侵入されると、情報流出につながる

⇒強固なセキュリティ対策を追加したカスタムAMIを作成

## ポイント

- Amazon EKS最適化AMIベースをセキュア化  
Amazon Inspector で徹底的に穴を塞ぐ・要塞化を実施
- セキュリティ対策の強化として **IDS/IPS を導入**  
内部犯行などで発生した場合でも侵入検知・防御



Amazon EKS最適化AMI



Amazon Inspector



IDS/IPS



カスタムセキュアAMI

- 
- ①コンテナPF
  - ②DB
  - ③OSS
  - ④Monitoring

# DBの選定

移行容易性、MySQLのサポートバージョンなど。高性能・高可用性の実現のために、  
**RDS vs Aurora vs Aurora serverlessを比較して選定。**

当初はAmazon Aurora Serverless利用を検討していたが、日本語対応・運用負荷の低減・高性能/高性能の観点から**Amazon RDS for MySQL**を選定。

## POINT1

運用メンテナンス  
マネージドサービス



Amazon  
RDS



Amazon  
Aurora



Aurora  
Serverless

## POINT2

ワークロード  
高負荷・高性能



Amazon  
RDS



Amazon  
Aurora

## POINT3

MySQL 8.x互換  
※ 2021年8月頃



Amazon  
RDS

# DBのハイパフォーマンスチューニング

RDSへ移行後、適切なパフォーマンスチューニングが必須。  
性能問題ポイントを押さえ、ボトルネックを素早く摘出して、高性能化へ。

## 高性能化に向けたパフォーマンスチューニングポイント

- 適切なインスタンスタイプの選択
  - AWS Graviton2の活用
  - マシン性能だけでなく、**最大接続数(max\_connections)**にも注意
- MySQLパラメータを適切にチューニング
  - DNS名前解決の省略(skip\_name\_resolve)の有効化
  - タイムアウトオプション(connect\_timeout, wait\_timeout etc)の最適化



## 性能問題が発生した場合は、ボトルネックポイントの確認

- 時間のかかるDBクエリがある場合  
⇒遅延とみなす閾値を設定後、スロークエリを確認。  
クエリの実行計画から適切なindexが有効か、テーブルフルスキャンがないか確認  
※ MySQLの場合、複合indexをwhere句での参照順序に合わせておくと効果的
- DBクエリが無応答になる場合  
⇒サーバ側だけでなく、クライアント側のtimeout値関連も確認  
※ JDBC socketTimeoutの初期値は無限待ちになるため、DBクエリの想定最長時間より大きくなるよう設定

- 
- ①コンテナPF
  - ②DB
  - ③OSS
  - ④Monitoring

# 既存OSSの移行(Replatform/Retain/Retire)

既存OSSの維持コスト低減と、移行後のランニングコスト・可用性を考慮して移行方法を検討。OSS利用状況から移行戦略を立案し、短期間の移行を実現。

## 課題

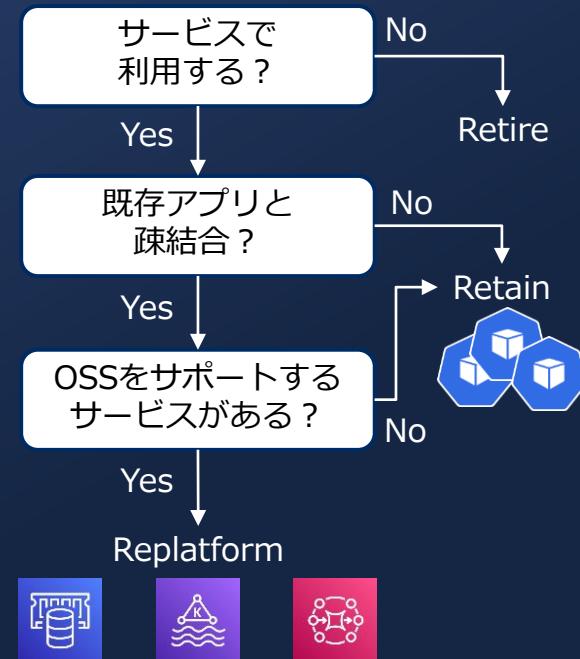
1. OSSを直接移行すると、OSS**保守コストが高い**
2. 高可用性の実現方式を**OSS毎に個別に検討必要**

## OSS移行戦略

- Replatform : AWSマネージドサービスへ移行
- Retain : 現状維持しつつ、高可用化
- Retire : サービスで利用しない機能は移行廃止

## 効果

- OSS**保守コストの削減**
- SLA保証がある**高可用性構成の実現**
- OSSの**素早いクラウド移行**の実現



- 
- ①コンテナPF
  - ②DB
  - ③OSS
  - ④Monitoring

# モニタリングのマネージドサービス化

点在する監視に必要なデータを、オブザーバビリティの起点となるCloudWatchに集約。高可用性、高信頼性を保ちつつ、システム監視に集中可能となつた。

## CloudWatch集約で既存問題を解決

1. データの永続性が低い  
⇒ 高可用性、高信頼性のCloudWatchに移行したため、  
**永続化のための設計コストを削減**
2. 監視設計の作り込みが非効率  
⇒ マネージドサービスにより、監視設計・構築などの  
**必要な作業に集中**できた
3. 必要なデータの点在  
⇒ CloudWatchに集約して管理効率を向上
4. OSSライセンスリスク  
⇒ マネージドサービス利用で**ライセンス問題から解放**

加えて、ログ分析サービスとの連携といった**拡張性も獲得**



Kibana icon by Icons8

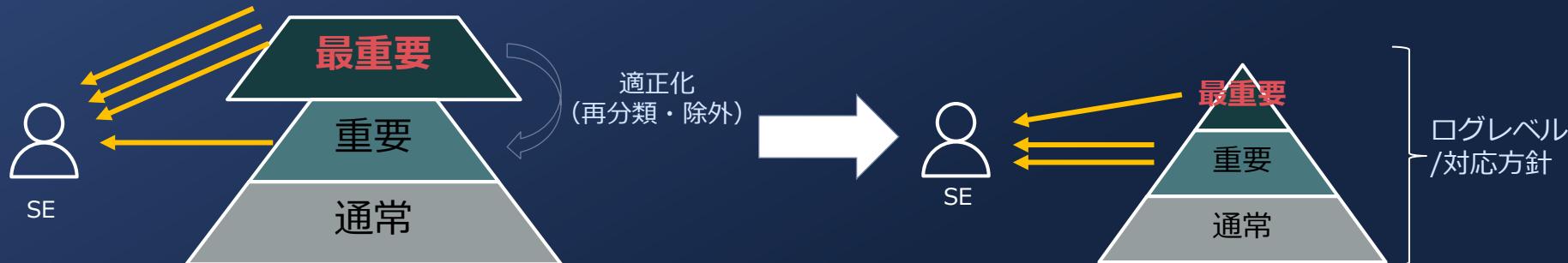
Elasticsearch icon by Icons8

# サービスデリバリ設計（監視設計・運用）

サービス品質を適正な状態に維持するための仕組みづくりが重要なポイント。

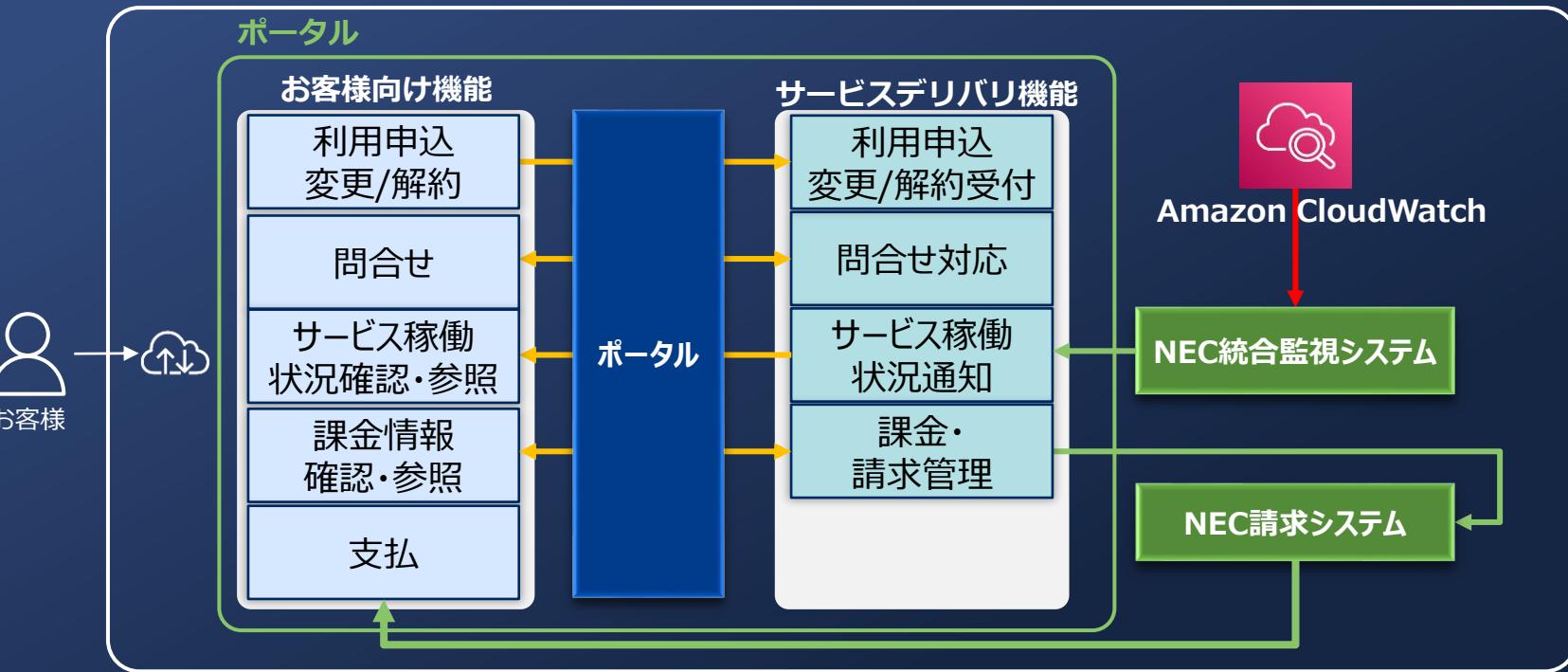
## ログ・アラート設計／運用

- ・ ログをサービス品質への影響度に応じたログレベルに分類し、それぞれ対応方針を定義
- ・ 運用開始後も適正状態への変更が柔軟にできるようにログフォーマットを設計し、監視アラートシステムと連携
- ・ アラート発生時、自動対応が困難な領域はSEによる手動対応が必要となるが、アラート大量発生時に適正な判断ができない可能性がある。アラート状況を継続管理し、影響度・必要性有無を検証するサイクルを回すことで、運用品質を適正な状態を維持



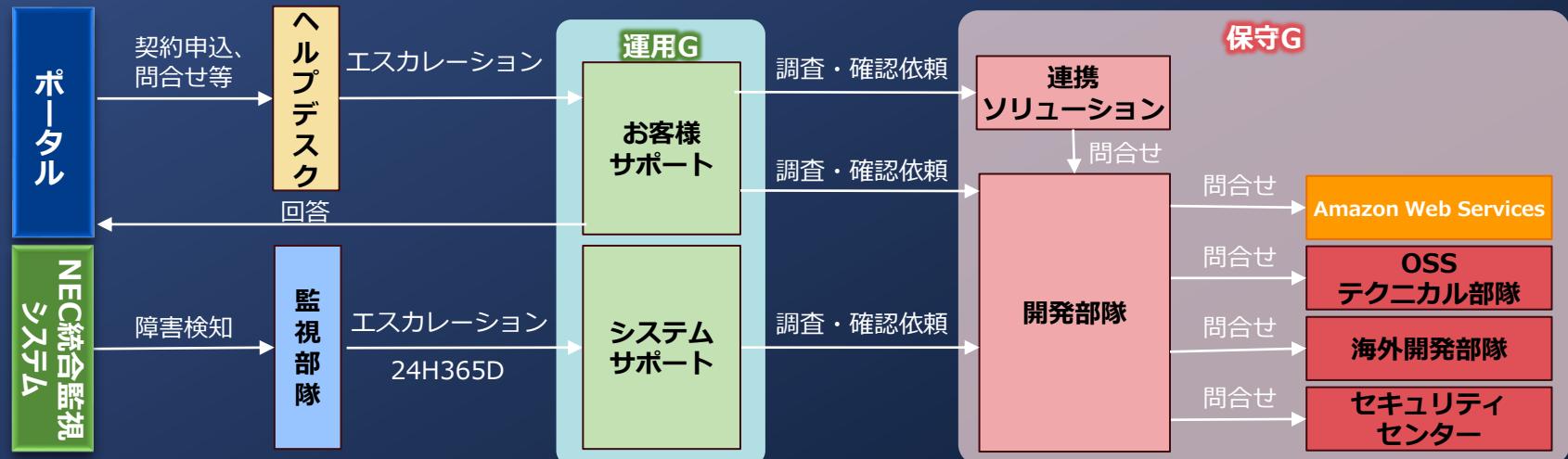
# サービスデリバリ設計（お客様とのインターフェース）

お客様に安心してサービスをご利用いただくため、ポータルを介して運用に必要な機能を提供。  
⇒ サービス事業者固有の社内システム・プロセス・レギュレーション等と合わせて一気通貫のサービスを提供。



# サービスデリバリ設計（体制・運用ルール）

サービスを安心してご利用いただくため、障害等の対応を迅速に行う運用・支援体制を構築。  
問題発生時を想定した運用ルールを準備し、継続的に改善活動を実施。



### 3. 生体認証のSaaS実現に 立ちふさがった問題

# 高性能、高信頼性を担保するための評価プロセス

弊社独自の非機能検証(MC6要件)を実施しシステムが設計通り実装されているか評価。

評価観点	評価内容（概要）
重要 高性能(性能・拡張性)	<ul style="list-style-type: none"><li>同時に大量のリクエストが発生しても全リクエストを所定時間内に完了すること</li><li>特定のリクエストが極端に遅延しないこと</li></ul>
高信頼性(可用性)	<ul style="list-style-type: none"><li>24時間365日稼働のシステムで断続的にリクエストが発生しても問題なく処理可能であること</li></ul>
運用性・保守性	<ul style="list-style-type: none"><li>運用時間、バックアップ、運用監視、サポート体制が定義に従って評価されること</li></ul>
移行性	<ul style="list-style-type: none"><li>システム移行が容易であること(AWS⇒Azureへの移行など)</li></ul>
安全性(セキュリティ)	<ul style="list-style-type: none"><li>脆弱性診断、データ暗号化、不正監視、セキュリティ試験が実施され対処されていること</li></ul>
システム環境	<ul style="list-style-type: none"><li>システム特性：ユーザ数・クライアント数・拠点数の有無が定義されること</li></ul>

## 検出した問題の原因特定・対処

- プロフェッショナルメンバーを投入して仮説・検証に基づくトラブルシューティングで解決に繋げる。
- 本プロセスをアジャイル的に実施。複数チームが並行して対応することで解決までの期間を短縮。
  - ✓ 環境構築をAWS上で簡単に準備することができるため実現できた



# 高性能観点①：性能要件と直面した課題（1/2）

同時に大量のリクエストが発生した場合に全リクエストが所定時間内に処理可能なことに加え、一部のリクエストのTATが極端に悪化しないことが要求される。

## 動作検証

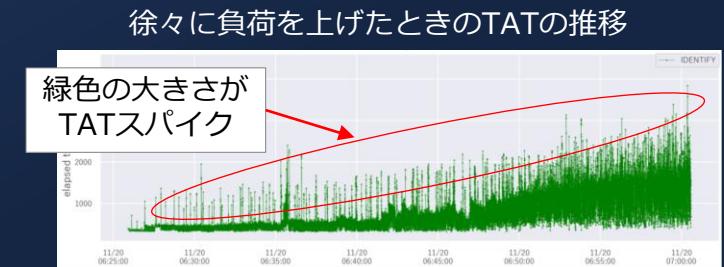
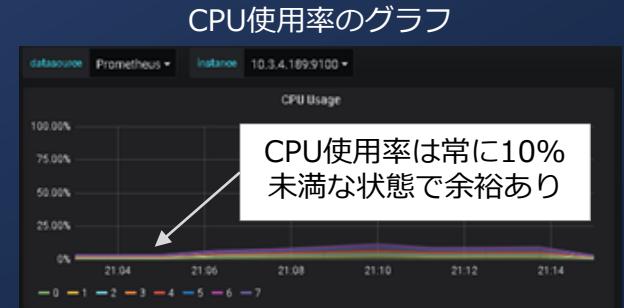
- 徐々に負荷を上げていき、応答性能が安定しているかを確認
  - ✓ 負荷を上げるとTATスパイクが発生する

## 事実整理

- CPUやメモリには余裕がある
- データ量が多い照合の方がデータ量が少ない照合より高性能  
(通常とは逆の傾向)

## 仮説立案

- ネットワーク観点の下記根拠からTCPコネクション枯渇と推測
  - ✓ 段階的負荷を上げるとスパイクが発生
  - ✓ トラフィックの少ないリクエストで傾向が顕著

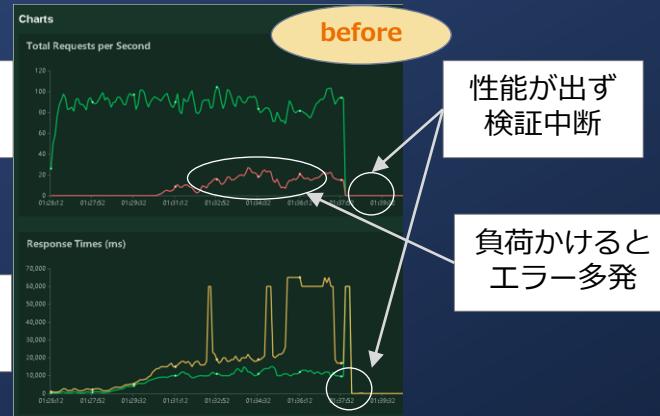


# 高性能観点①：性能要件と直面した課題 (2/2)

## 仮説検証・原因特定

- 今回のシステムは、一つのリクエスト複数コンテナに分散して処理しており、更にその処理を同時並列で動かすことで大量のリクエストを捌けるようにする必要がある。
    - ✓ TCPソケットの使用可能な範囲を最大限に広げる必要がある
    - ✓ TCPソケットの再利用を即時可能とする必要がある
- ⇒ 対象のPodのクライアント側にTCPソケットの上記カーネルパラメータをチューニング  
使用可能なソケットの範囲を最大化し、即時再利用可能となるようにチューニング

上段の図がリクエスト負荷(緑)とエラー(赤)



下段の図が平均TAT(緑)と95%タイル(赤)



# 高性能観点②：性能要件と直面した課題（1/3）

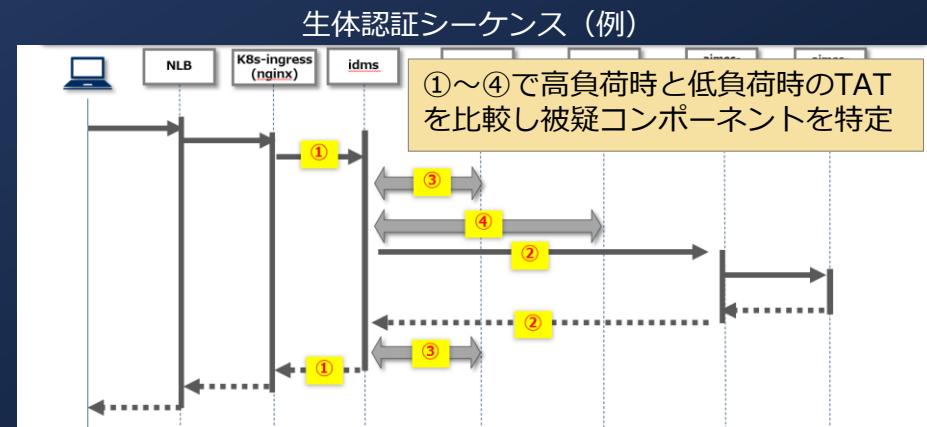
スタジアム/ビル入退のように同時にかつ大量な入場処理(リクエスト)が発生した場合、TATが極端に悪化しないこと。一部のリクエストのTATの極端な悪化もないこと。

## 動作検証

- 大規模構成かつ高負荷状態でも、TATが安定しているかを確認する。
  - ✓ 高負荷時に目標TATを満たさない問題を検出。  
※TCPカーネルパラメータはチューニング済

## 事実整理

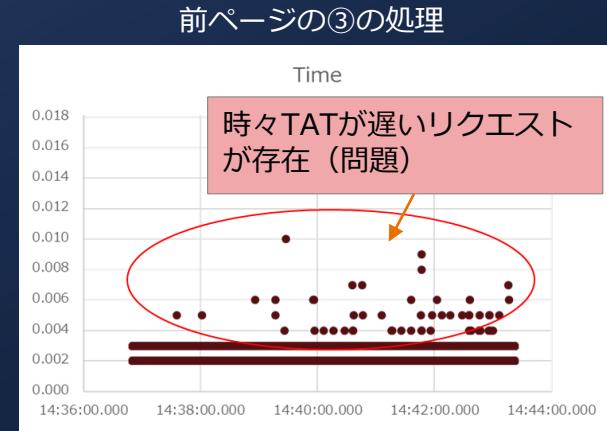
- システム全体を俯瞰するため、各コンテナにパケットキャプチャを仕掛けて処理の遅いコンポーネントを特定する必要がある。(右図)
- 高負荷時のパケット解析であるため、個別のパケット確認は現実的に不可能。(次に続く)



# 高性能観点②：性能要件と直面した課題（2/3）

## 事実整理(続き)・仮説立案

- 処理シーケンスを確認(前ページ①～④)。高負荷時と低負荷時のTATの違いをシーケンス毎に調査
- コンポーネントB $\Leftrightarrow$ C(RDS)の間で**時々性能劣化がみられる**ことを確認(下図右)
  - ✓ DBアクセスが時々遅い……**テーブル構造やSQL構文の最適化が不十分？(仮説)**
  - ✓ RDS内の問題特定が必要 ⇒ **CloudWatch で RDS メトリクスを確認**

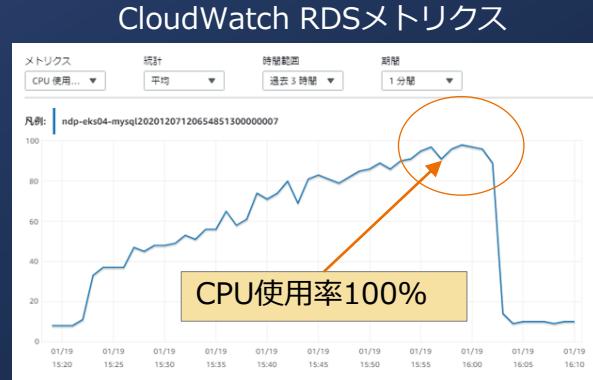


# 高性能観点②：性能要件と直面した課題（3/3）

## 仮説検証・原因特定

- 仮説に対する下記被疑ポイントをCloudWatch Logsから検証
  - ✓ テーブル構造やSQL構文は最適化されているか？
  - ✓ Lower関数の性能に対してテーブル構造が適切か？
  - ✓ Indexは機能しているか？(機能していない場合、適切なindex付与が必要)

⇒ slow query ログから処理時間が長いクエリを確認  
Explainで対象クエリを特定



処理に時間がかかる  
るクエリが存在

CloudWatch Logs slow query

行ラベル	データの個数 / SQL	平均 / QueryTime	平均 / LockTime	平均 / RowsExamined
select attributes0.EDGE_ID as EDGE_ID12_37_0_, attributes0.EDGE	3	0.118876667	0.007444667	2
select datasetgroup0.DS_GROUP_ID as DS_GROUP1_26_, datasetgrc	3	0.100751667	6.56667E-05	1
select datasetsou0.DATASET_SOURCE_KEY_ID as DATASET_1_27_	3	0.104901333	0.003489	10
select distinct datasetsou0.DATASET_ID as col_0_0_, datasetsou0.	1473	1.774983692	0.000440907	857694
select edge0.EDGE_ID as EDGE_ID1_35_, edge0.JENANT_ID as T	1	0.106874	0.000083	1
select membership0.MEMBERSHIP_ID as MEMBERSH1_193_, mem	13258	4.00509286	0.00434243	826236
update CFG_BATCHJOB_INFO set ACTIVE_FLAG=N, CREATE_BY=''	1	0.159841	0.000084	1
UPDATE QRTZ_TRIGGERS SET TRIGGER_STATE = 'S' WHERE SCH	2	0.174364	0.0055535	4
総計	14744	3.778870127	0.003951822	828650.3119

# 高信頼性観点：可用性要件と直面した課題(1/2)

ビル/ホテル入退のように、24時間365日稼働が必要なシステムでも安心してご利用いただけるように長時間連続運転中の断続的なリクエストが問題なく処理できること。

## 動作検証

- ・ 高負荷連続運転試験で、TATの悪化はないか確認
  - ✓ あるタイミングからTAT悪化および処理工エラーの発生が継続する
  - ✓ 事象発生までの時間は試行毎に変わる(12H未満での発生もあれば、24H以上未発生もあり)

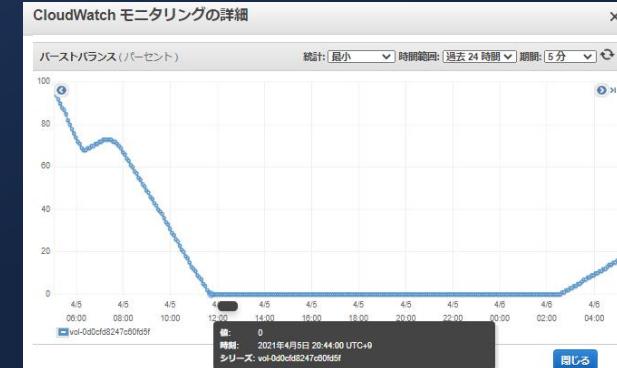
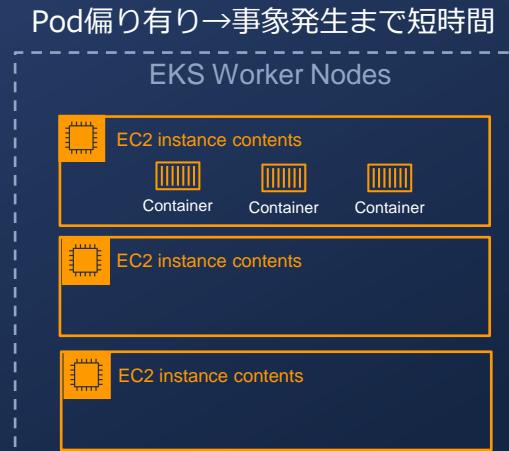
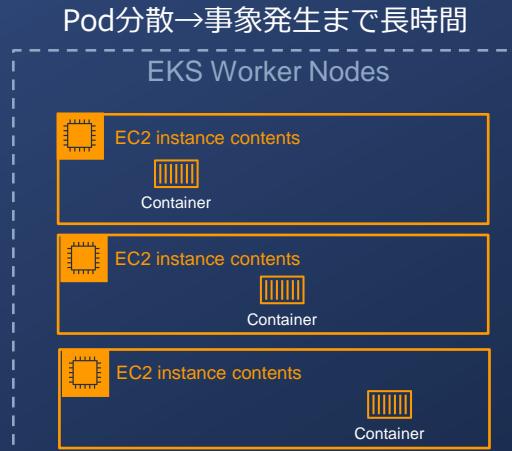
## 事実整理

- ・ 問題発生個所特定のため、「Pod観点」でCPU/MemoryやNWの使用状況を確認
  - ✓ 特に異常なし
- ・ プログラム内の占有時間可視化のため、**処理時間可視化ツール(NewRelic)を導入**  
⇒ SpringFramework内、「ファイルアクセス」処理の占有時間が長いことを特定
- ・ 原因となるファイルアクセス特定のため、Pod内のファイルアクセス状況をOSコマンドで確認  
⇒ 対象ファイル特定。**これで解決かと思いきや、TAT悪化の原因の特定には至らず。** (次に続く)

# 高信頼性観点：可用性要件と直面した課題(2/2)

## 事実整理(続き)・仮説立案/検証・原因特定

- **視点変更**: 事象発生時間が試行毎に異なることから、起動度に変化する**Podのノード配置が影響**と仮定。  
⇒ Podを特定Worker Nodeに配置し再現試験実施、事象発生までにかかる時間の短縮を確認  
⇒ 「**Worker Node**」側に着目
- 「ファイルアクセス」 + 「Worker Node」の条件より**EBS が原因**と仮説  
⇒ **EBSのメトリック情報をCloudWatchで確認し、クレジットの枯渇と判明**  
※EBSタイプを gp2 から gp3 に変更して現象の改善を確認



# 4. NECにおける変革の実践

# NEC オフィスのデジタル化 全体イメージ



# NEC Digital ID 入退場ゲート

ウォークスルー入退

マスク顔認証



社内売店での決済



体表温測定

顔認証+パターン認証

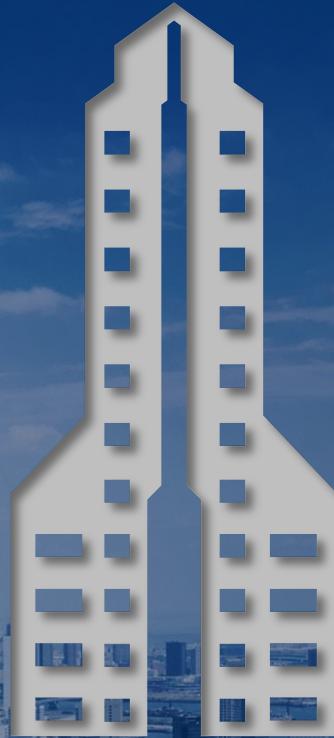


NEC Digital ID 決済POS

# さまざまなオフィスサービスをシームレスに利用



# NECにおけるコロナ禍における業務変革の目的



- 1 社員とお客様の安全を守る
  - 2 社員のモチベーション維持、不安解消
  - 3 更なる生産性向上、省力化
- そして、働く日常における「わくわく感」づくり

# 5. NEC Digital Platform 他社適応事例

# NECの映像技術による感染症対策に対しての取り込み

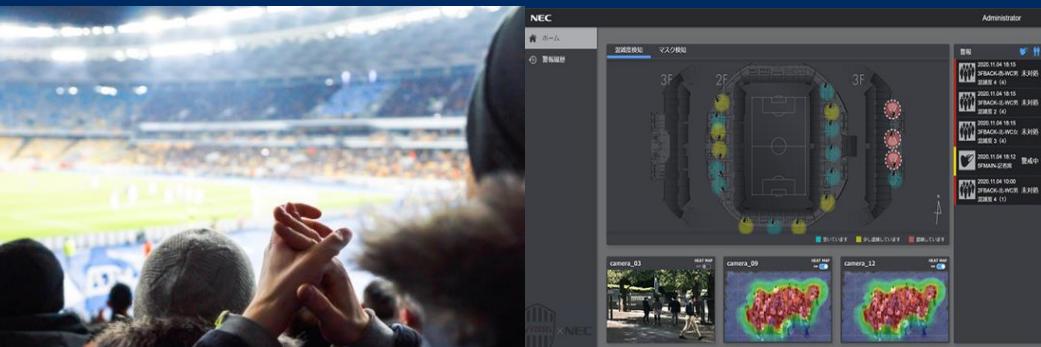
スマートな搭乗手続き、感染症対策で、安全・安心な空の旅を実現

- 搭乗ゲートでの体温確認、非接触で通過  
※ 今後、顔認証技術を活用した空港内外でのサービスを拡充予定
- 2019年7月よりスターアライアンス様との協業推進
- 2020年8月よりハワイ主要5空港に導入
- 2020年11月、フランクフルト空港、ミュンヘン空港から商用サービスを開始
- 2021年11月、石垣島にてPCR結果のスムースな確認



コロナ禍での安全・安心かつ快適な大規模イベント実現

- 便利で快適なスタジアム環境の整備
- 映像分析によるマスク未着用検知、混雑度モニタリングによるスタジアム内の滞留回避検討など、利用者・従業員の感染リスクを低減
- 会員様向けサービスとして、顔情報の事前登録による非対面での受付や、個人に適した情報の提供



# NECの生体認証・映像分析ソリューション提供例

## 来訪者検知システム／入退室管理システム

### 機能・特長

#### 来訪者検知



来訪者を登録し、来場時に通知して業務サポートします。商業施設、公共交通機関でのVIP検知、動線分析などにご利用いただけます

#### 入退場管理



サーマルカメラとも連携。スクリーニングを行いながら入退室管理の業務を向上します

#### 本人確認



イベント会場の受付などでスマートデバイスのカメラとQRコードを使用して本人確認を実施できます

### 利用イメージ



## 導入事例: 某製造業

### 1. 入退管理

電子錠と顔認証を連携し入退を管理

### 2. 勤怠システム連携

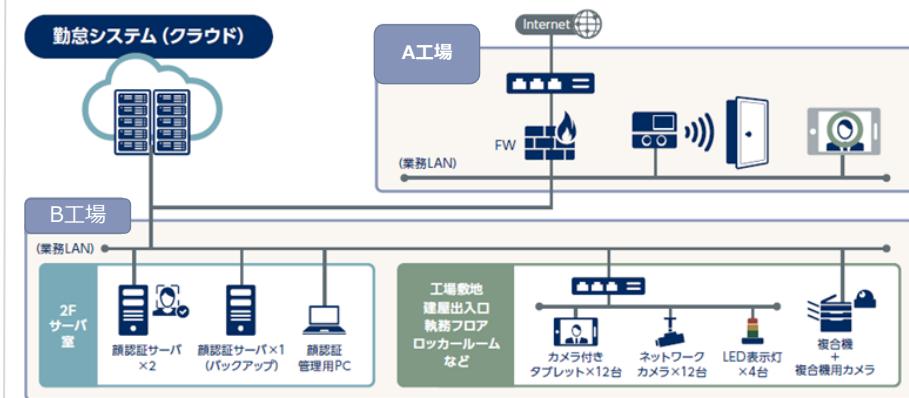
顔認証と連携し工場の入室・退出を記録

### 3. 残業申請

顔認証と連携し工場の入室・退出を記録

### 4. 複合機連携

複合機を顔認証で実施



# 導入事例：ビル入退

大規模なスケールでのマスク対応、かつウォークスルー顔認証入退を実現

課題 | 新本社ビル開業にむけ、利用者の利便性向上とセキュリティ強化の両立

ソリューション | マスク対応ウォークスルー顔認証システムの導入

効果 | 手ぶら認証による利便性向上  
マスク対応・非接触での認証により、安全性を確保しつつ新本社ビルの先進性もアピール

- エントランスゲート全てをマスク対応ウォークスルー顔認証ゲートに
- 一部VIP導線での入室時に顔認証を実施
- マスク対応、非接触での認証により、感染症拡大のリスク低減にも貢献
- 今後は、オフィス内での適用範囲の拡大について検討予定



エントランスゲート



VIP導線

# 導入事例：マスク着用率測定・人流調査

街の人流の変化を可視化することで行動変容の評価を定量化。  
感染防止施策の立案のインプットとしても活用できます。

## 活用法と効果

### ①リアルタイム分析による行動変容の効果測定

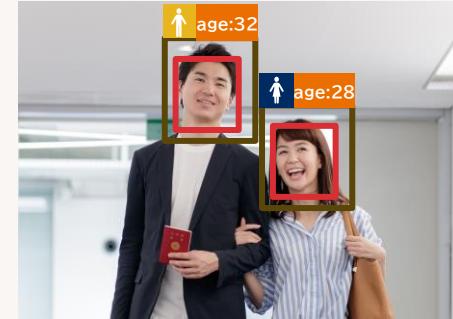
政府/自治体の外出自粛要請や緊急事態宣言等の前後、時間帯別の人流の変化を見る化  
要請先に情報提供することで、対策の効果測定が可能に

### ②マスク着用率を捉える

マスク着用者に対しても、人物を検出・追跡し、人流を捉え、着用率推定が可能

### ③雨の日、紫外線の強い日は、傘で人流を推定

頭部、顔が隠れる場合でも、人物と傘の組合せで、検出・追跡による人流推定が可能



## 活用事例

### ①自治体様公共施設など

駅前、観光施設にカメラを設置。自治体が住民にデータを公開  
取得した人流データは住民に公開されると同時に市のコロナ対策会議へ報告  
感染者情報と合わせて情報開示することで、住民に過度の不安を与えないよう配慮



### ②都内某大型商店街

都や関係各所へ人流データを定期的に報告。自粛要請への貢献度と効果測定を実施



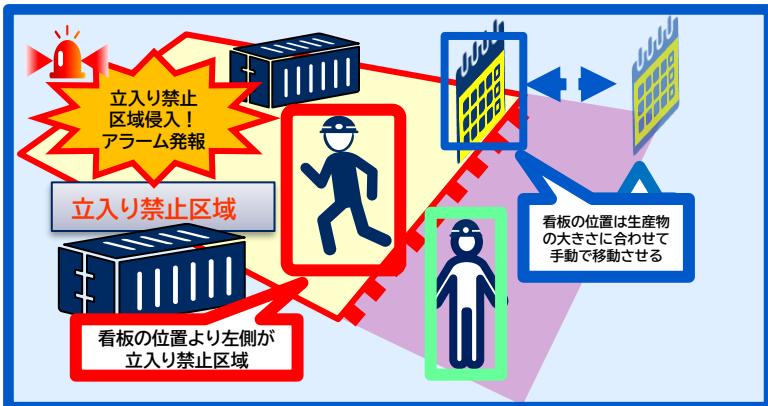
# 導入事例：工場従業員の安全管理

センサーによる作業員の安全管理で課題をお持ちだったお客様に対し、  
AIによる映像解析で作業員を検知しアラームを発報することで、作業員の安全を確保します。

## 1. 変化する立入り禁止区域でのアラーム

### 【課題】

センサーでは動的に変化する立入り禁止区域での適用は難しい



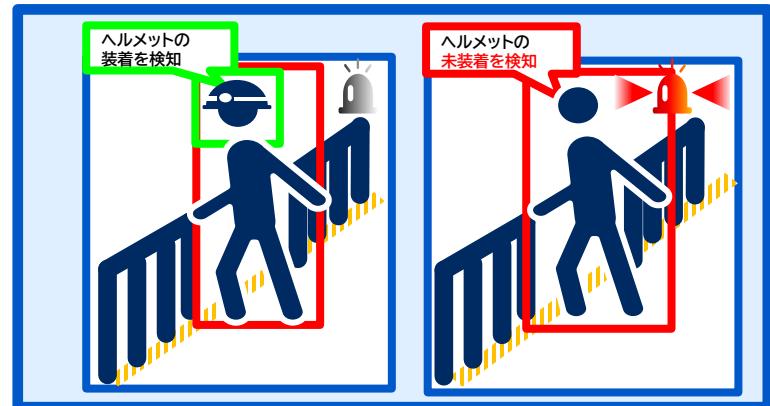
### 【改善策】

映像解析で看板の位置および作業員の立入り禁止区域の侵入を検出し、アラームを発報

## 2. 作業員が正しく保護具を着装しているか確認したい

### 【課題】

センサーでは作業員が正しく保護具を着装しているか確認ができない



### 【改善策】

- ・映像解析でヘルメットの装着の有無を検知
- ・入退場時刻を記録し一定時間内に退場しない場合アラームを発報し、事故を防止

# 導入事例：ナンバープレート認識および車両移動経路分析機能を活用した重要エリア移動経路分析

重要エリアにおける車両の特定と、異常運転の検知や目的地の到達確認を実施し、  
エリア内の安全管理を実現しています。



# お問い合わせお待ちしております

NECでは生体認証や映像分析の様々なソリューションを用意しお困りごとを解決できます。これらを使った業務変革やソリューション構築などをご検討でしたらお気軽にお問合せ下さい。貴社のお困りごとの最適な解決策を弊社の専門家と一緒に考えませんか。



## ■メールでのお問い合わせ

[contact@nids.jp.nec.com](mailto:contact@nids.jp.nec.com)

## ■WEBサイトからのお問い合わせ

※ページ内の「お問い合わせ」よりご連絡下さい

<https://jpn.nec.com/delight/services/index.html>

# 本日のまとめ

- 1 生体認証の新たな価値「非接触」
- 2 生体認証 + 映像分析による安全・安心の実現
- 3 NEC Digital PlatformにおけるAWS活用

AWSマネージドサービスフル活用による  
社会の変化へのスピーディな適応が重要

\Orchestrating a brighter world

**NEC**

# Thank you!

奥山 嘉昭

日本電気株式会社  
生体認証・映像分析統括部  
シニアプロフェッショナル

大竹 孝昌

日本電気株式会社  
サービスプラットフォーム統括部  
シニアプロフェッショナル

本セッション資料や記載内容については一切の転用を禁止しております