



2022.06.27

# クラウド時代、DX時代に必要なモニタリング - O11y(Observability) -

New Relic株式会社

技術統括本部 コンサルティング部

Koji Aizawa (@kaojiri)

# Safe Harbor

This presentation and the information herein (including any information that may be incorporated by reference) is provided for informational purposes only and should not be construed as an offer, commitment, promise or obligation on behalf of New Relic, Inc. (“New Relic”) to sell securities or deliver any product, material, code, functionality, or other feature. Any information provided hereby is proprietary to New Relic and may not be replicated or disclosed without New Relic’s express written permission.

Such information may contain forward-looking statements within the meaning of federal securities laws. Any statement that is not a historical fact or refers to expectations, projections, future plans, objectives, estimates, goals, or other characterizations of future events is a forward-looking statement. These forward-looking statements can often be identified as such because the context of the statement will include words such as “believes,” “anticipates,” “expects” or words of similar import.

Actual results may differ materially from those expressed in these forward-looking statements, which speak only as of the date hereof, and are subject to change at any time without notice. Existing and prospective investors, customers and other third parties transacting business with New Relic are cautioned not to place undue reliance on this forward-looking information. The achievement or success of the matters covered by such forward-looking statements are based on New Relic’s current assumptions, expectations, and beliefs and are subject to substantial risks, uncertainties, assumptions, and changes in circumstances that may cause the actual results, performance, or achievements to differ materially from those expressed or implied in any forward-looking statement. Further information on factors that could affect such forward-looking statements is included in the filings New Relic makes with the SEC from time to time. Copies of these documents may be obtained by visiting New Relic’s Investor Relations website at [ir.newrelic.com](http://ir.newrelic.com) or the SEC’s website at [www.sec.gov](http://www.sec.gov).

New Relic assumes no obligation and does not intend to update these forward-looking statements, except as required by law. New Relic makes no warranties, expressed or implied, in this presentation or otherwise, with respect to the information provided.

# Koji Aizawa

Business

*Building a Digital Platform and Services*

Cloud Computing

*AWS - EKS, ECS, Fargate, etc.*

Specialities

*kubernetes on AWS, containers*

## “Containers Power to Enterprise”



2021 APN AWS Top Engineer



Twitter: [@kaojiri](https://twitter.com/kaojiri)

# アジェンダ

1. 会社紹介
2. クラウド時代・DX時代に必要なモニタリング
3. New Relicで実現できること
4. Why New Relic?
5. まとめ

# New Relicの概要

事業内容	オブザーバビリティプラットフォームの提供
設立	2008年
創業者	ルー・サーニー (Lew Cirne)
上場市場	NYSE: NEWR
従業員数	グローバル約2,300人
拠点数	16拠点
売上	約\$673M

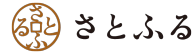
2018年8月に日本法人設立、日本語によるサポート可能



# New Relic Customers



ADASTRIA



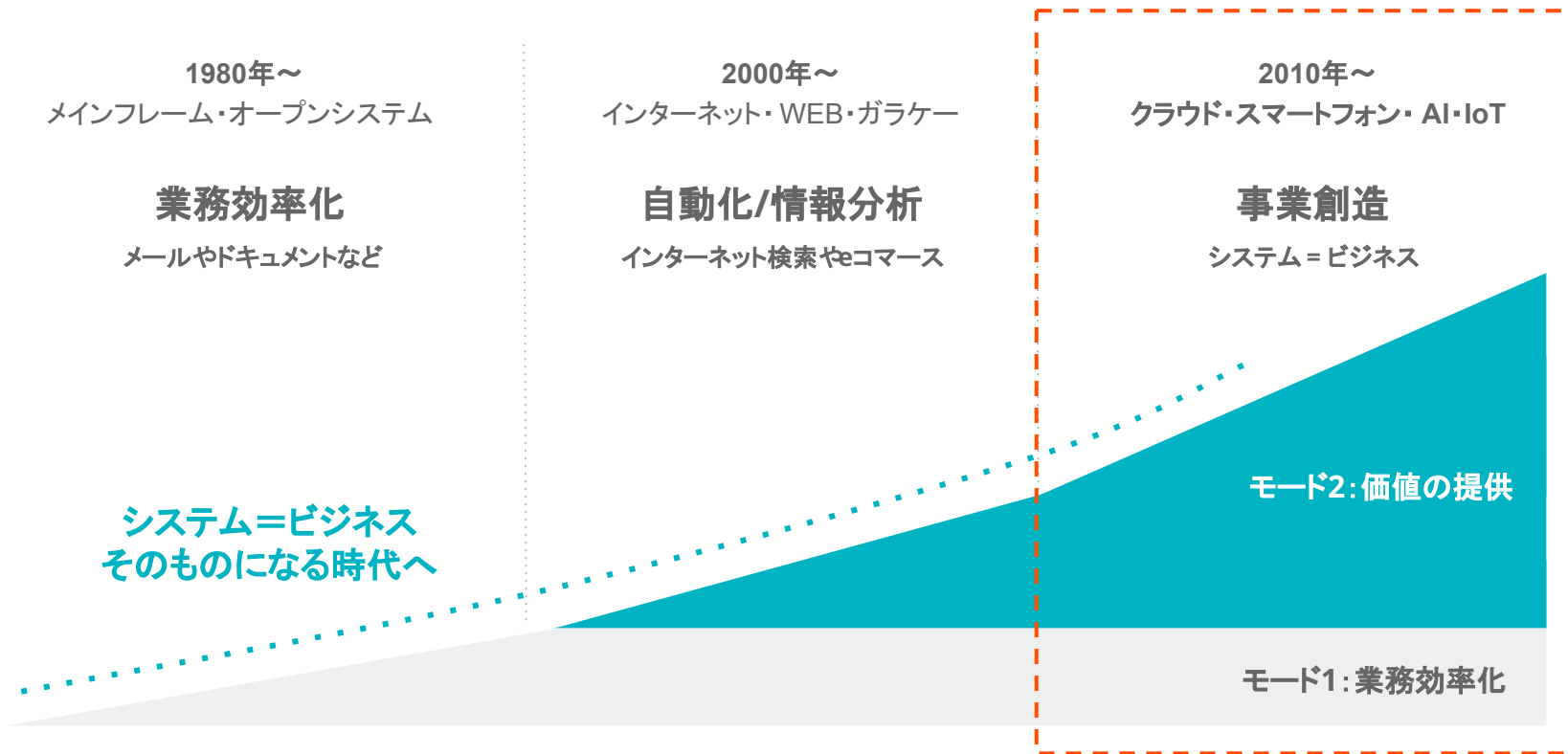
SUPER STUDIO



# クラウド時代・DX時代に必要なモニタリング

---

# システムはビジネスそのものになりつつある

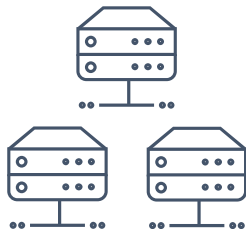


# ITシステムと技術の進化

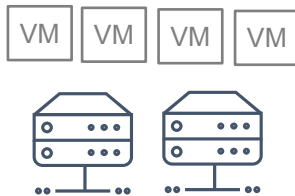
メインフレーム



オープンシステム



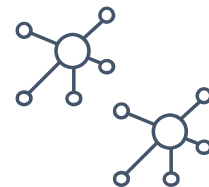
サーバー仮想化



クラウド



コンテナ・サーバレス



利用者

少

一部の社員

社員全員

取引先

世界中の人

多

システムに対する需要が読めない → より簡単にスケールする仕組みへ

重視されていること

安定稼働

新機能追加 / 改善

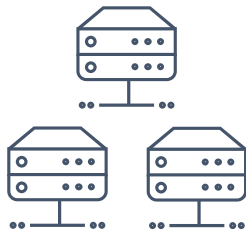
市場ニーズに対する早急な対応や新規需要の創出を目指す  
→ 基盤のコード化 (IaC) やマイクロサービス化へ

# ITシステムの進化に伴う監視の変化

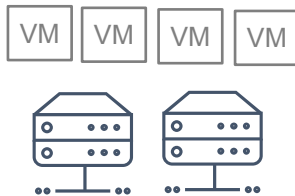
メインフレーム



オープンシステム



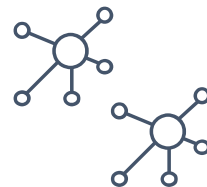
サーバー仮想化



クラウド



コンテナ・サーバレス



監視対象  
の数

少

サーバー

仮想マシン

クラウドリソース

コンテナ 多

監視対象はより多く、より複雑に

監視対象  
の状態

静的

動的

監視対象は常に変化するように

監視の観点

システム内部

ユーザー体験

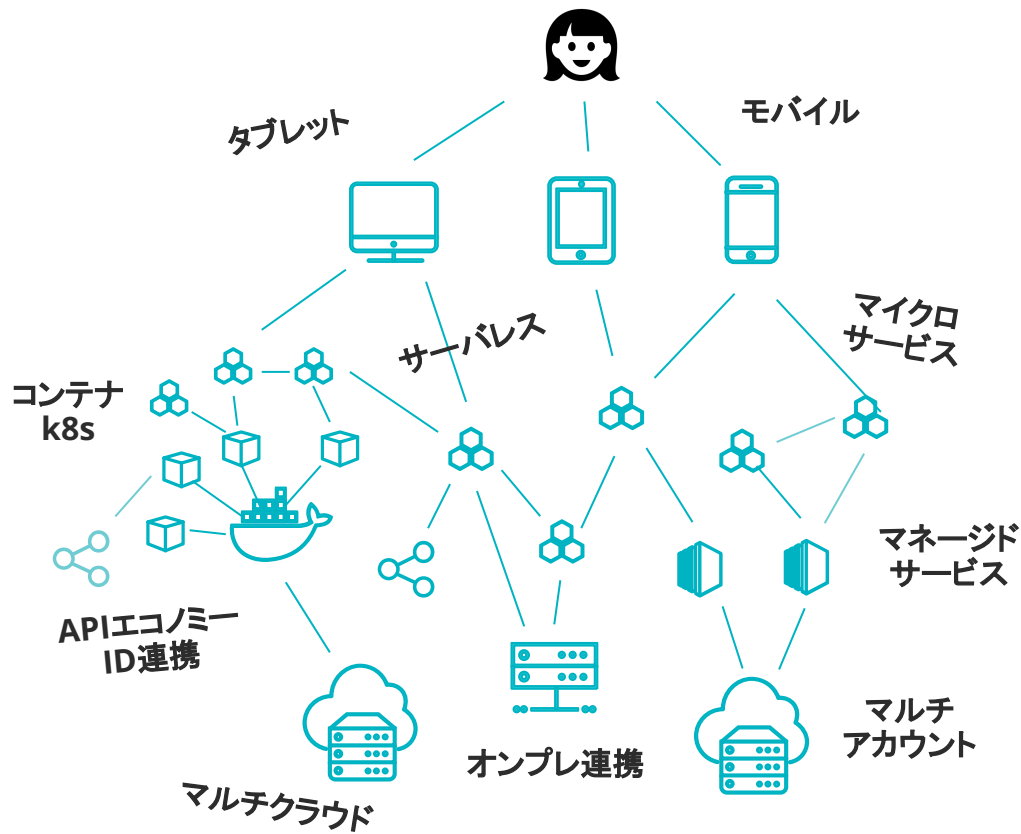
ユーザー視点の監視が重視されるように

デジタルビジネスの  
加速を後押しする  
テクノロジーの進化



開発・構築は速くなったが  
アーキテクチャは複雑化

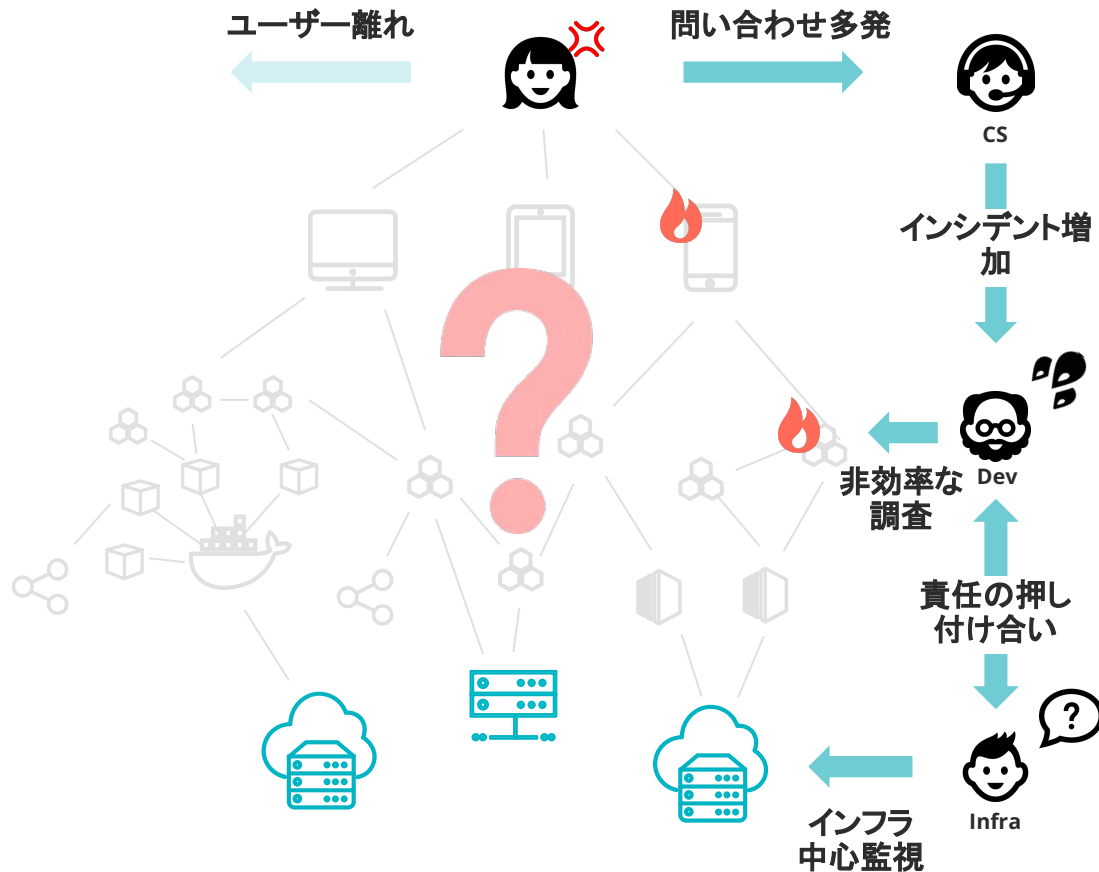
アジリティのある改善や  
運用の効率化には  
システム全体の観測が  
必要不可欠に



サービス視点・  
ユーザー視点での  
観測性・品質把握の欠如

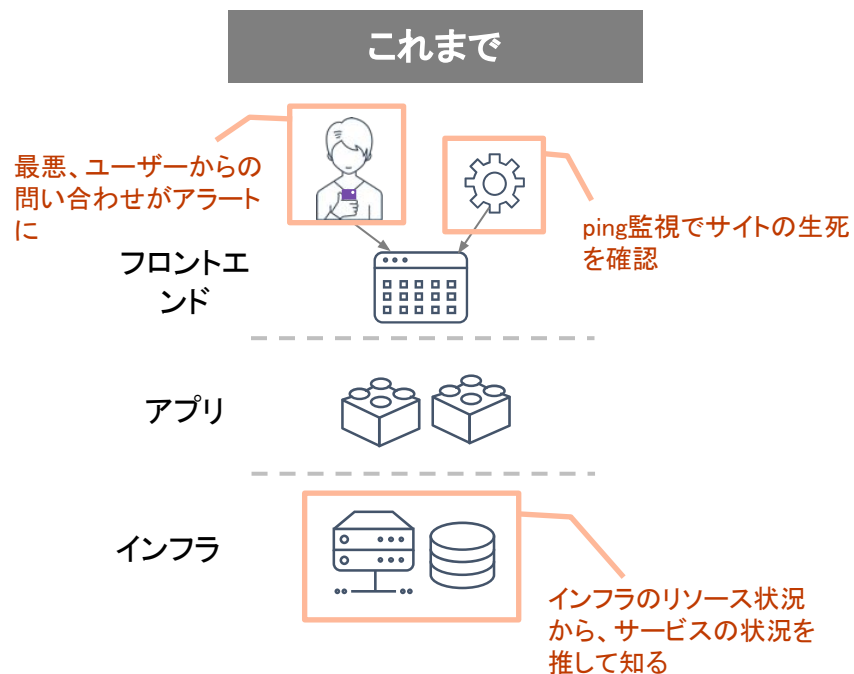


リアクティブな障害対応  
責任の押し付け合い  
非効率な調査  
インシデント対応増加  
顧客満足度低下



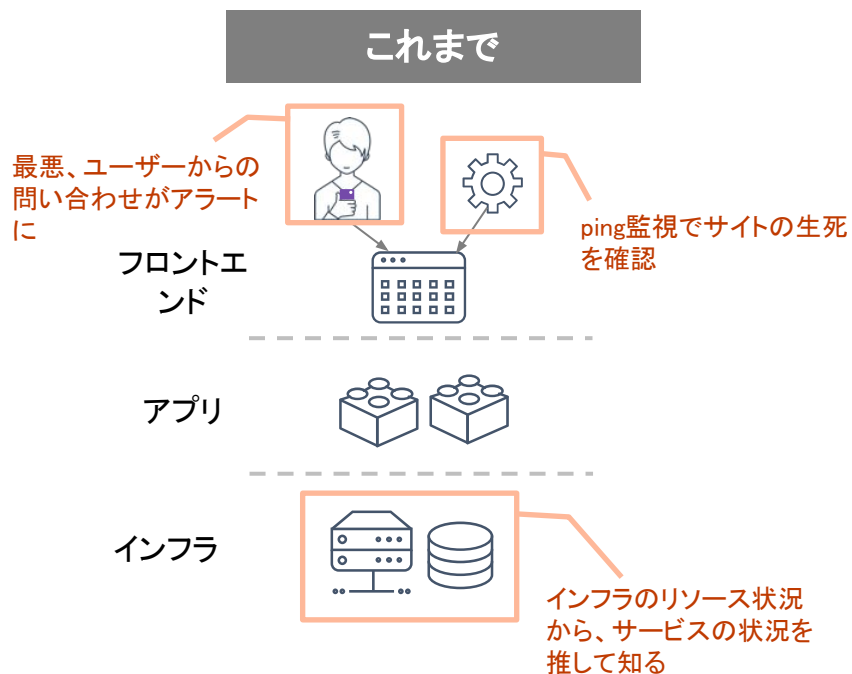


# システムモニタリングのこれまでとこれから



“システム”が問題なく動いているかを監視

# システムモニタリングのこれまでとこれから

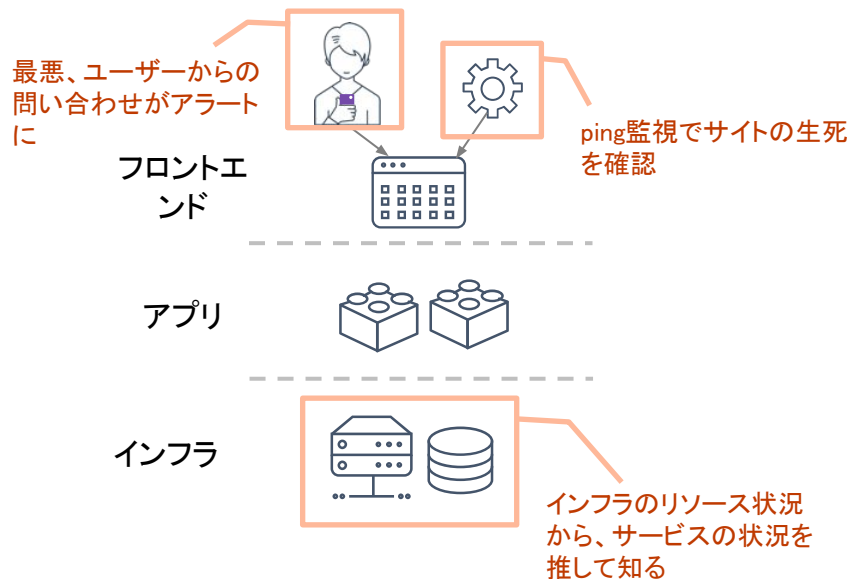


監視の種類	監視対象	
インフラ監視	ハードウェア監視	ハードウェアステータス
	リソース監視	メモリ
		CPU
		ディスク
		ネットワーク
	プロセス	プロセス
サービス		
セッション		
ログ監視	ログ	
サービス監視	URL監視	URL応答
	画面遷移監視	画面遷移
	API監視	API応答

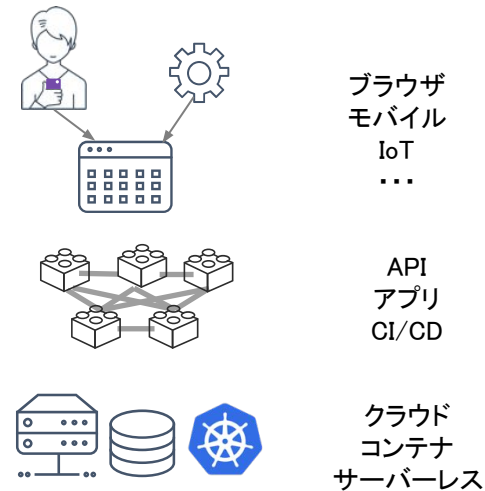
“システム”が問題なく動いているかを監視

# システムモニタリングのこれまでとこれから

これまで



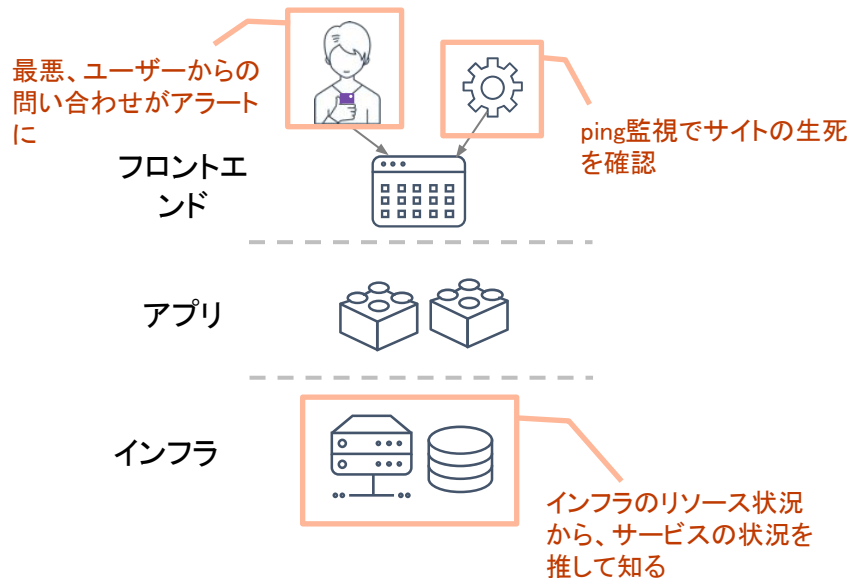
これから



“システム”が問題なく動いているかを監視

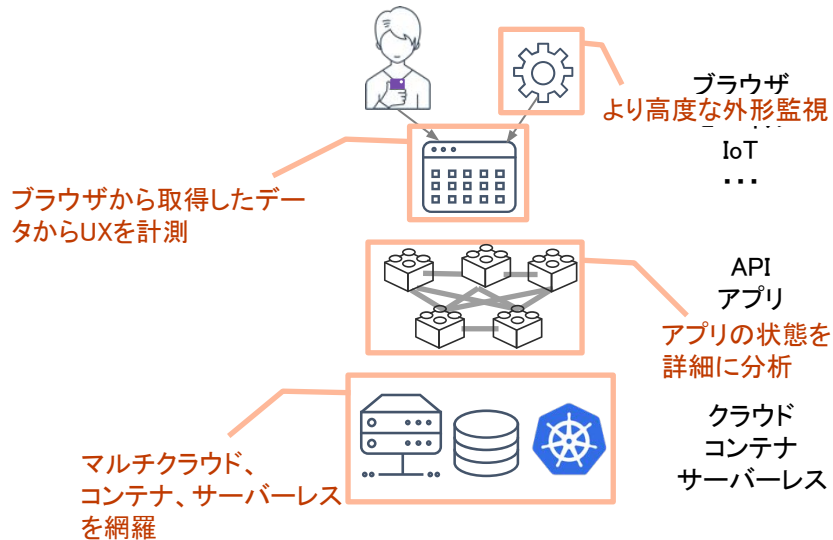
# システムモニタリングのこれまでとこれから

これまで



“システム”が問題なく動いているかを監視

これから



“サービス”が継続的に提供できているかを監視

# アラートのアンチパターンとデザインパターン

## アンチパターン: OSのメトリクスのアラート

” MySQLが継続的にCPU全部を使っていたとしても、レスポンスタイムが許容範囲に収まっていれば何も問題ありません。”

“OSのメトリクスは診断やパフォーマンス分析にとっては重要です。しかし99%の場合、これらのメトリクスは誰かを叩き起こすには値しません。”

出典: 入門監視 (Oreilly, 2019)



# アラートのアンチパターンとデザインパターン

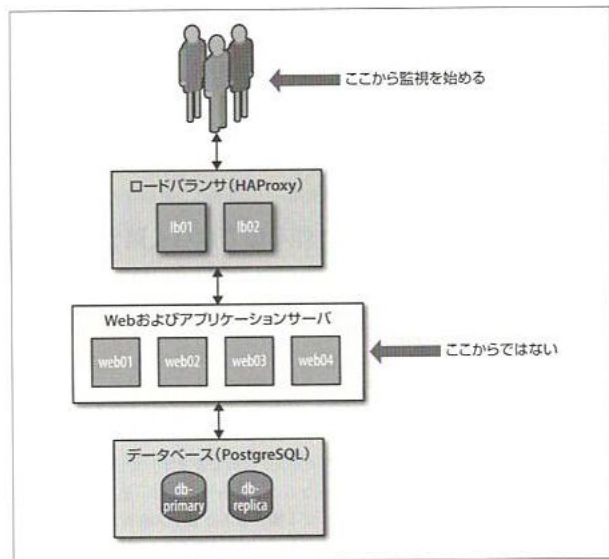


図2-1 できるだけユーザーに近いところから監視を始める

## デザインパターン:ユーザー視点の監視

”ユーザーが気にするのは、アプリケーションが動いているかどうかです。”

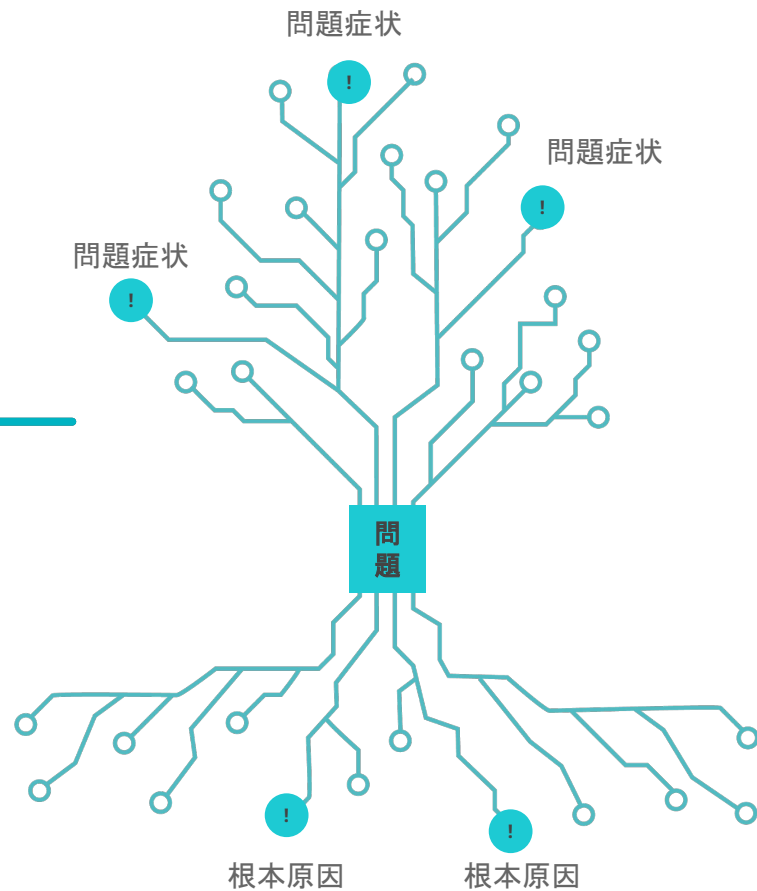
“ユーザー視点優先の監視によって、個別のノードを気にすることから解放されます。”

出典: 入門監視 (Oreilly, 2019)

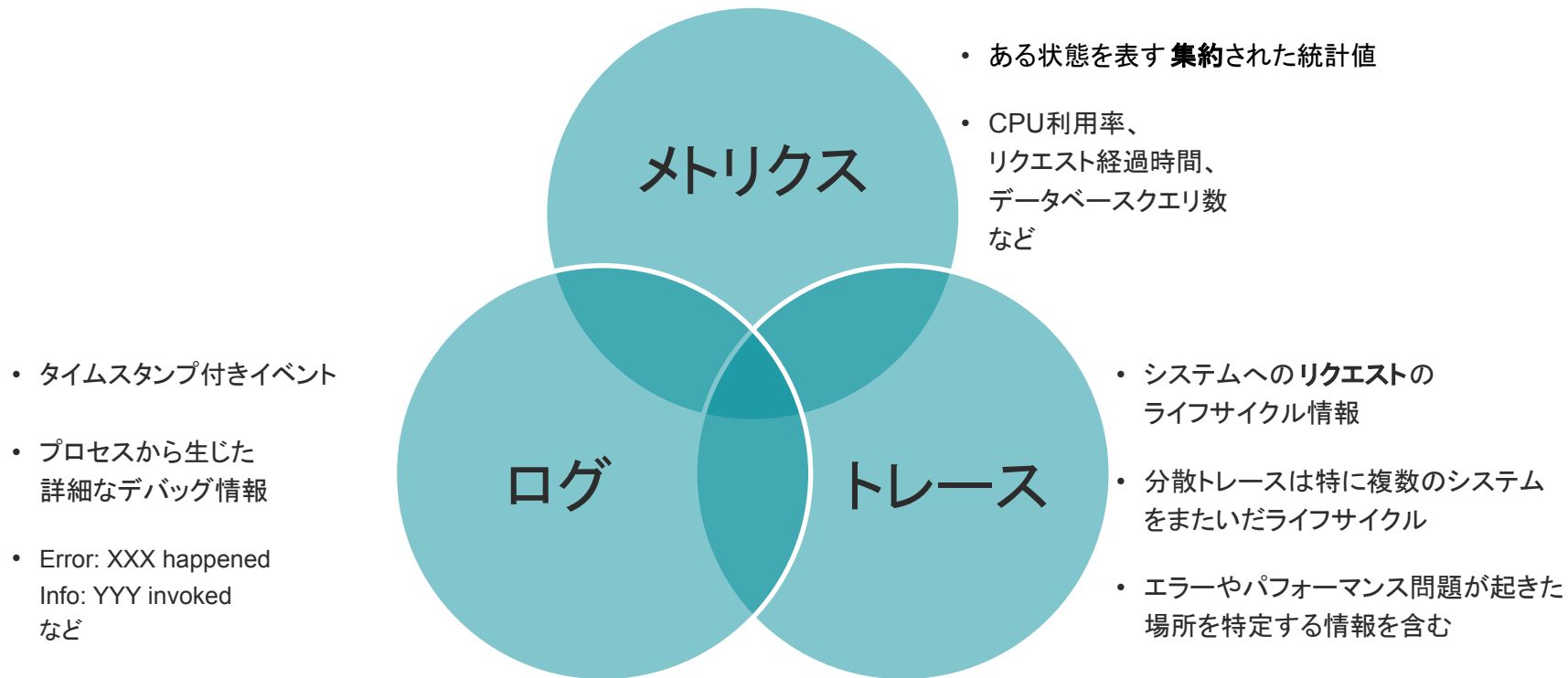
# オブザーバビリティ

Observability

システムのメトリクス・イベント・  
ログ・トレースのデータを  
リアルタイムに取得し続け、  
常にシステム全容の  
状態把握と改善ができる状態



# オブザーバビリティに必要なデータ



# 必要なのは根本原因の発見と継続的な改善

## 症状

CPU利用率90%以上

メトリクス

レスポンス遅延による  
パフォーマンス低下が  
発生している

メトリクス

## 問題

(問題なし)

全体のトランザクションの  
うち、データベースから  
の取得処理を行っている  
サービスで時間がかかっ  
ている

トレース

## 根本原因

CPU利用率に加えて別のデータを計測し  
問題ないことを把握できるようにする

インデックスが貼られて  
おらず、フルスキャンが  
走っている

ログ

## 改善 アクション

インデックス設計の見直し

# New Relicで実現できること

---

# New Relic One オブザーバビリティプラットフォーム

より良いソフトウェアの開発と実行



## New Relic **BROWSER**

ブラウザ体験モニタリング  
ユーザー目線でページロードやエラーを把握

## New Relic **MOBILE**

モバイル環境をモニタリング  
iOSとAndroidアプリに対応

## New Relic **SYNTHETICS**

外形モニタリング  
世界複数拠点からの外形監視



Back-End

## New Relic **APM**

アプリケーション性能モニタリング  
8言語と70を超えるフレームワークに対応

## New Relic **INFRASTRUCTURE**

あらゆるインフラ環境をモニタリング  
パブリッククラウドとオンプレミス

## New Relic **LOGS**

ログ収集と高速検索  
MELTを高速収集し検索可能に

## New Relic **NPM**

SNMP、NetFlow、VPCなどのネットワークを  
容易に収集可視化



Analysis

## New Relic **AIOps**

AIを利用した検知とインシデント対応の効率化

## New Relic **ONE**

ダッシュボード開発チャートビルダーで分析を  
超高速化し、あらゆるテレメトリデータの可視化  
を実現

## New Relic **Programability**

データ分析をツールの自作



Perfect  
Software

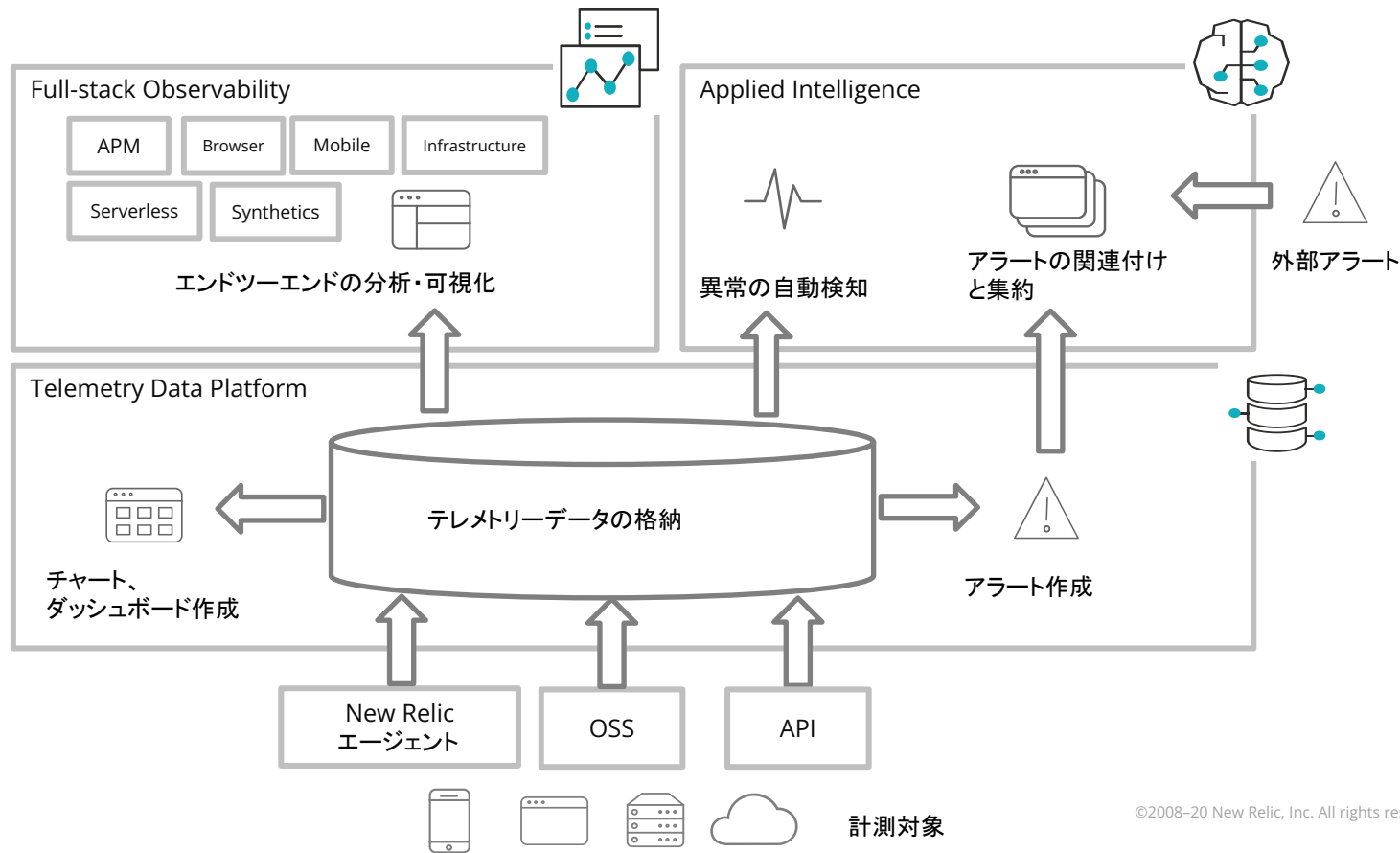
顧客体験の改善

複雑かつ大規模システムの管理

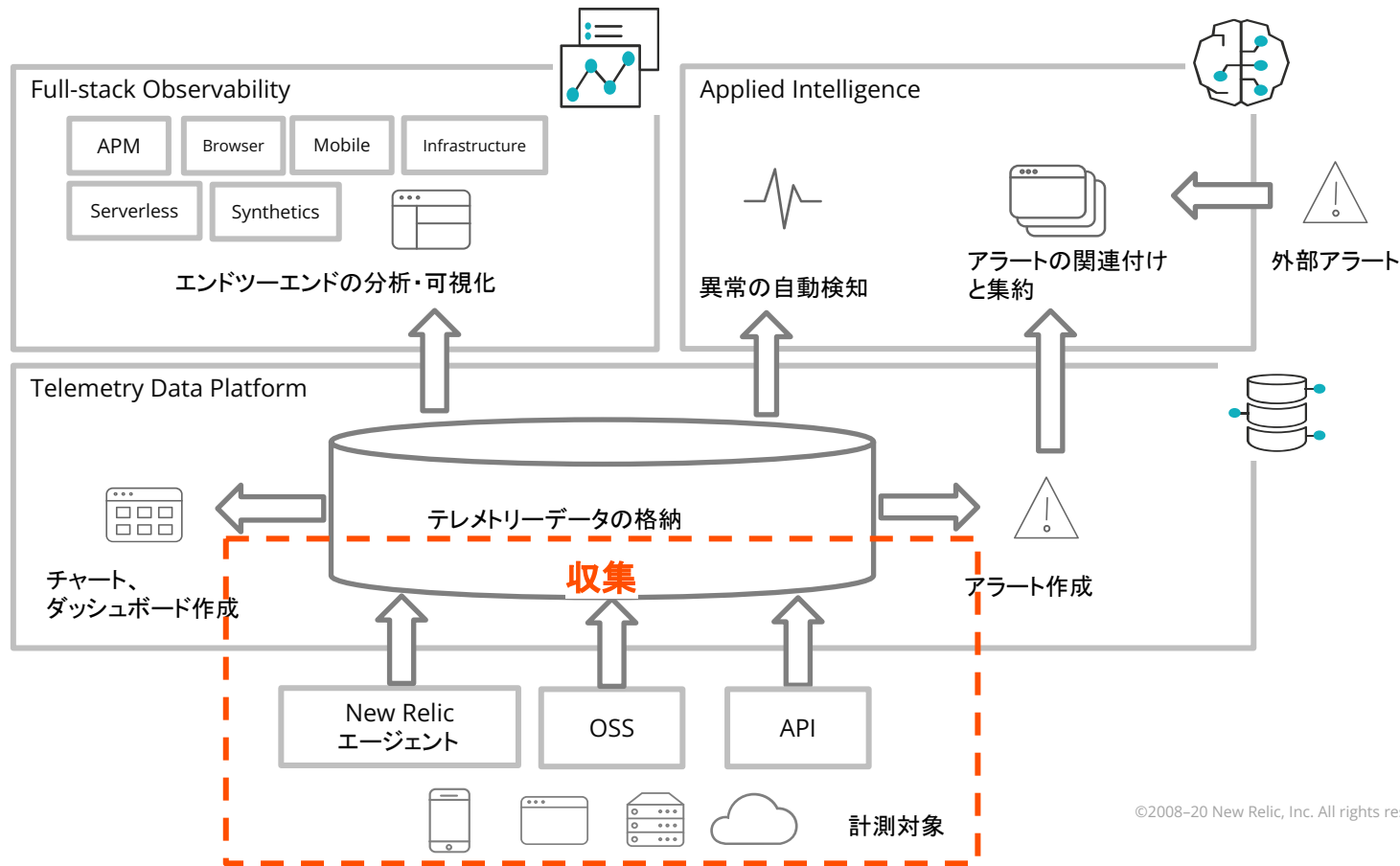
## NRDB

世界最速のデータ収集と検索

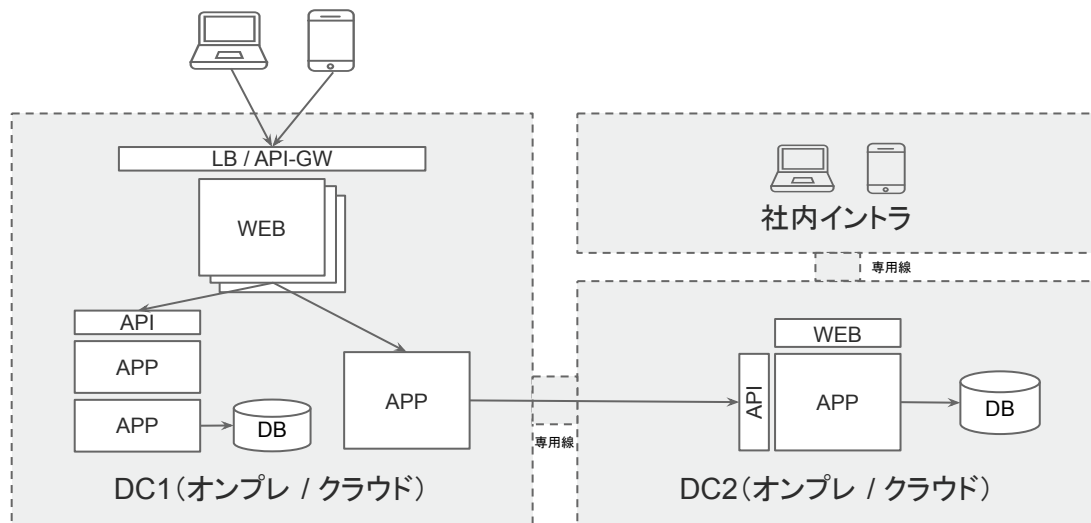
# New Relic One の全体像と関係性



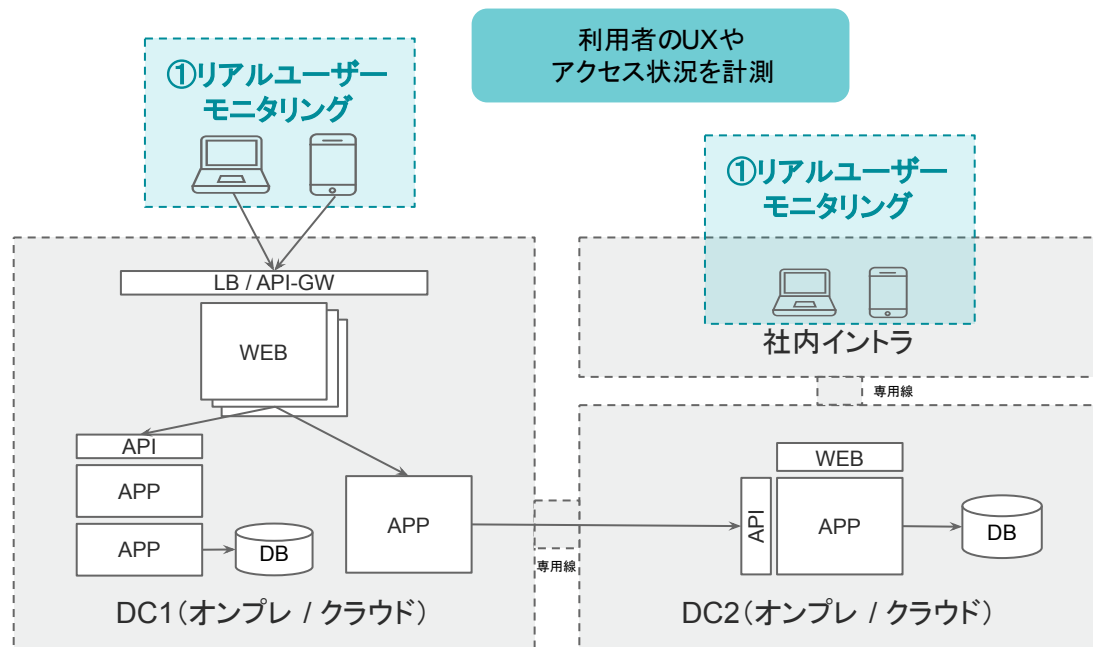
# New Relic One の全体像と関係性



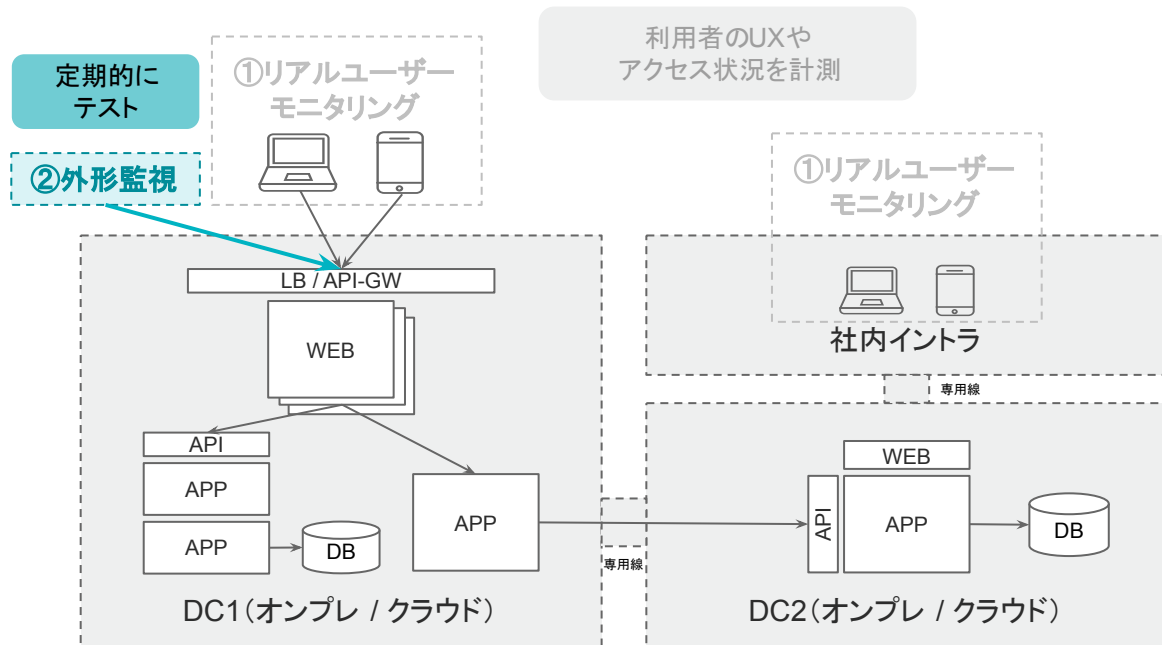
# すべてを観測しシステム全体を分析・可視化



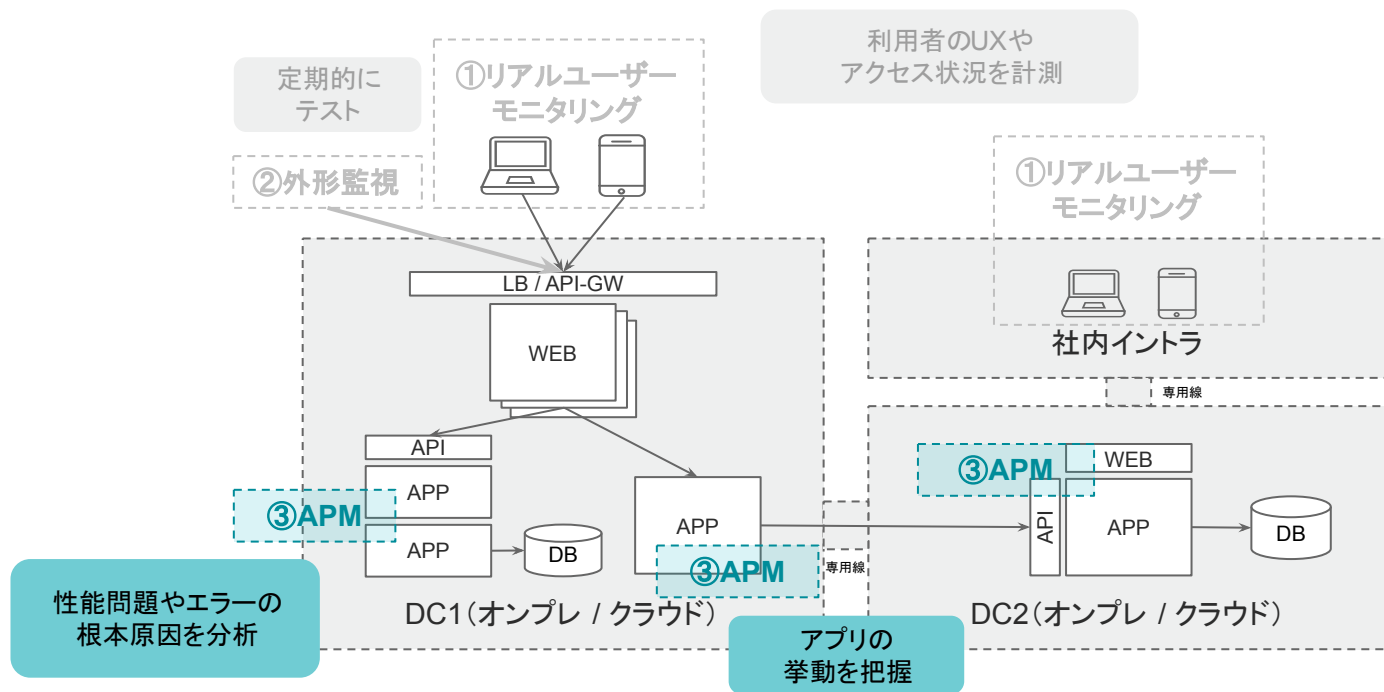
# すべてを観測しシステム全体を分析・可視化



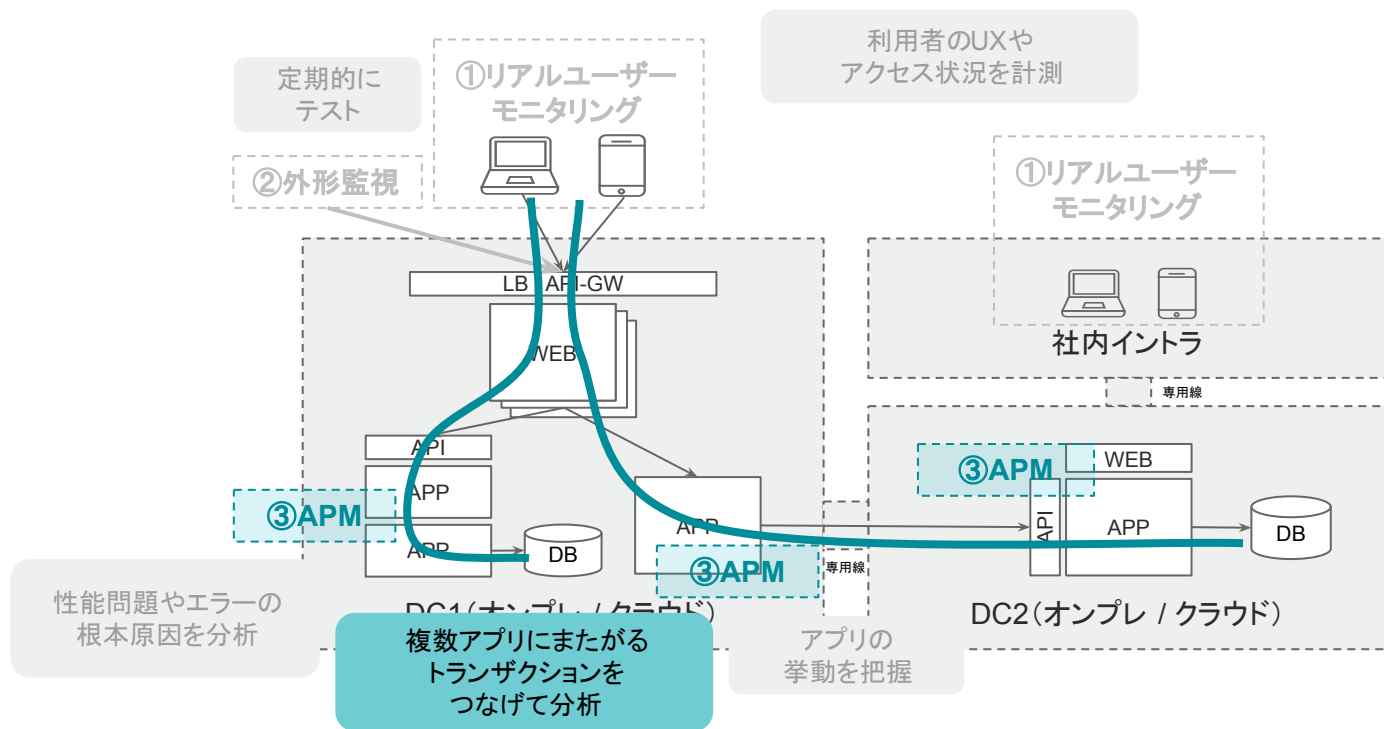
# すべてを観測しシステム全体を分析・可視化



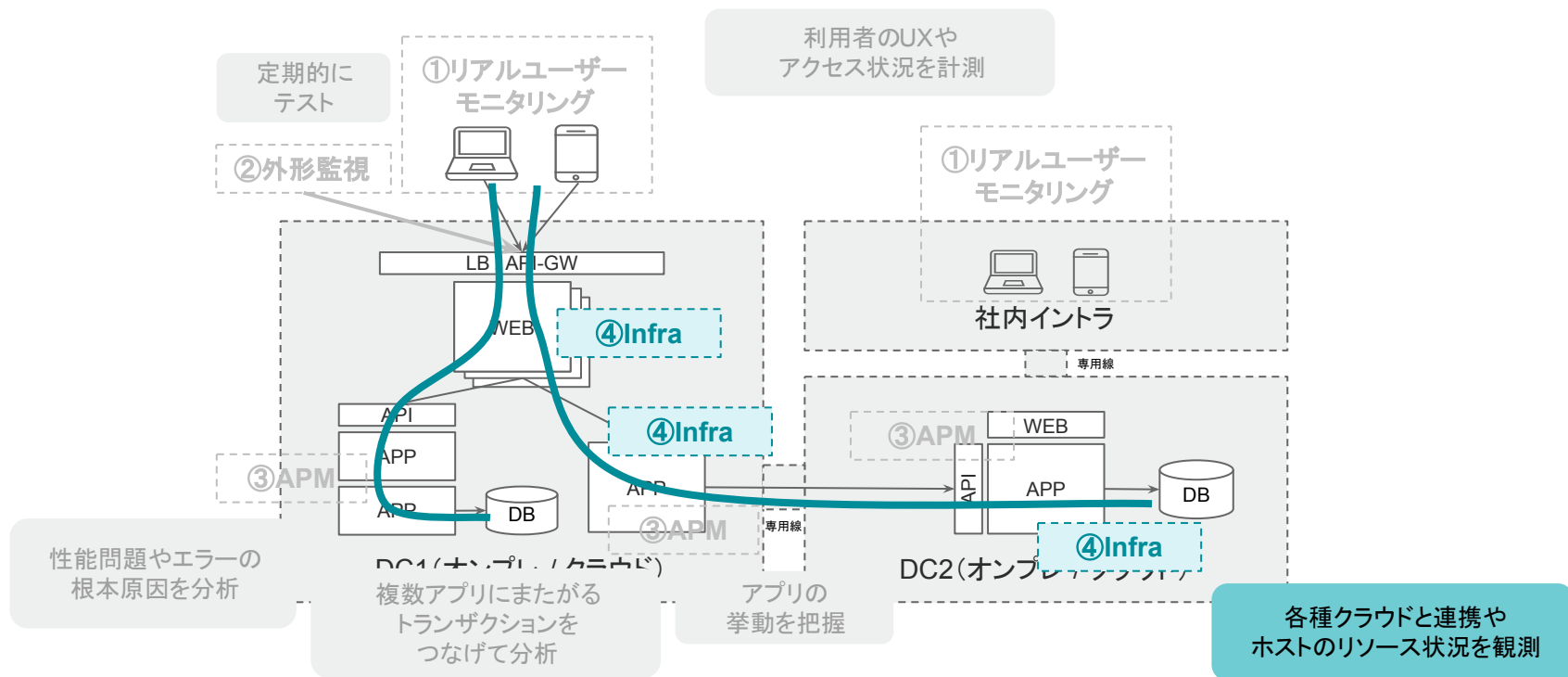
# すべてを観測しシステム全体を分析・可視化



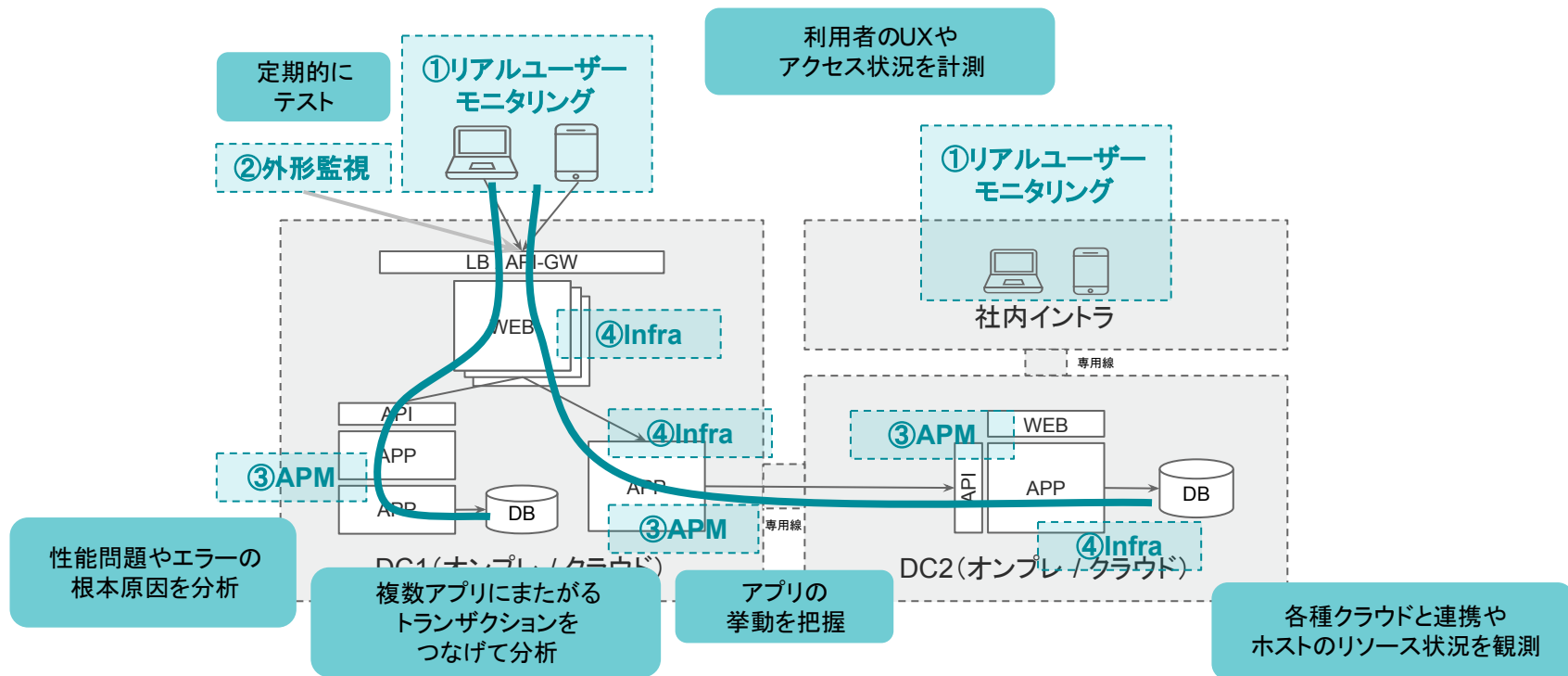
# すべてを観測しシステム全体を分析・可視化



# すべてを観測しシステム全体を分析・可視化



# すべてを観測しシステム全体を分析・可視化



# 豊富な連携先

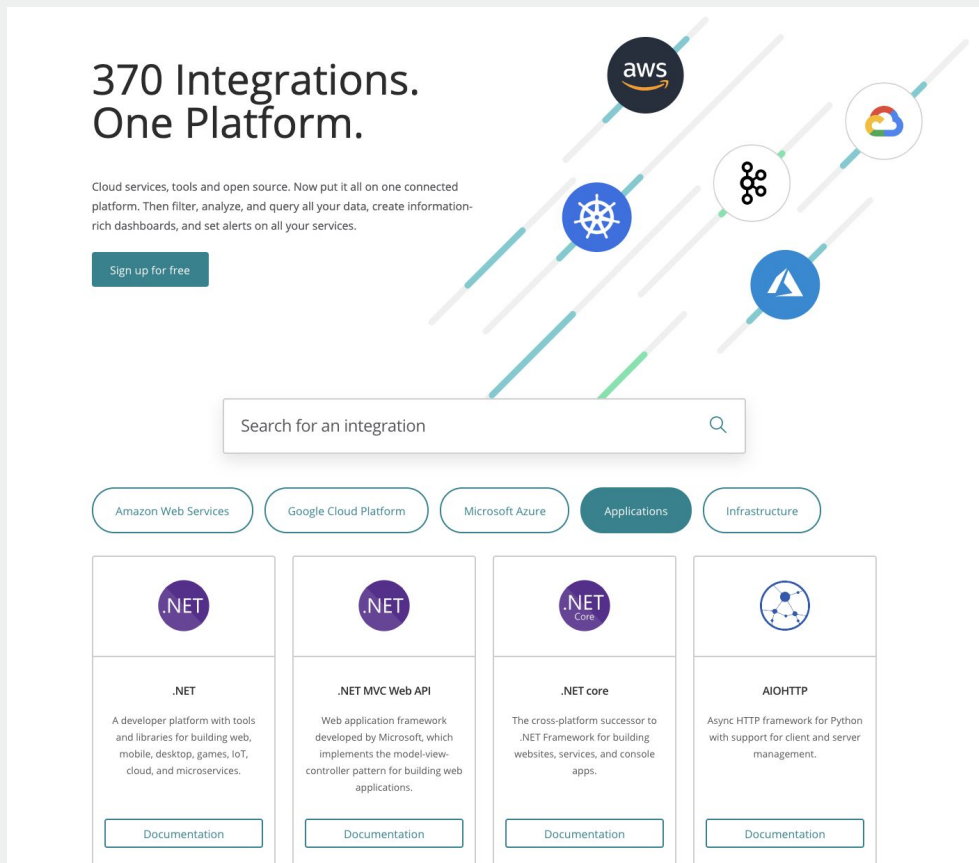
## ① クラウドとの連携による収集



## ② サービス・ミドルウェアから収集



## ③ OSS ツールを活かして収集







370 Integrations.  
One Platform.

Cloud services, tools and open source. Now put it all on one connected platform. Then filter, analyze, and query all your data, create information-rich dashboards, and set alerts on all your services.

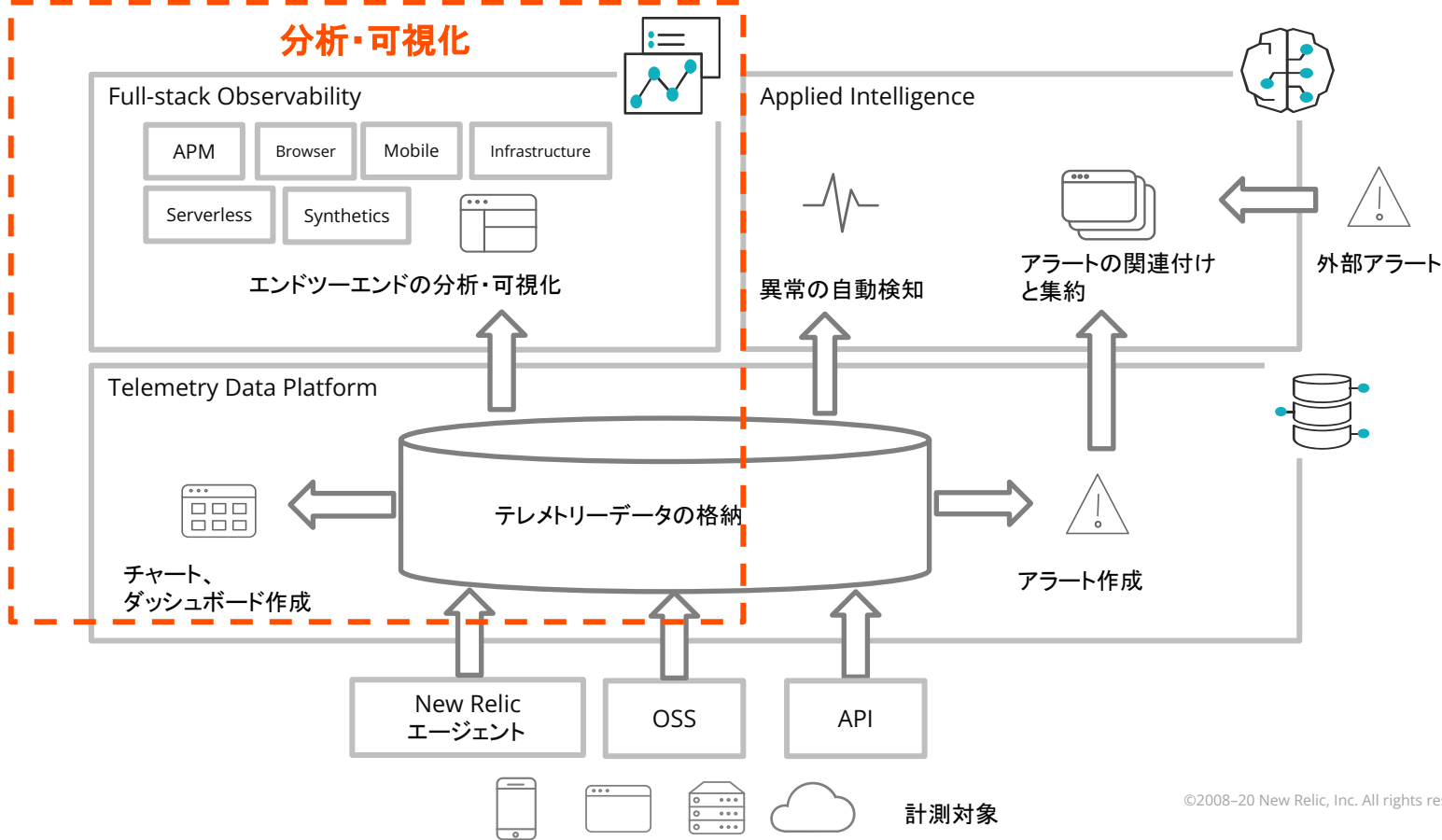
[Sign up for free](#)

Search for an integration

Amazon Web Services | Google Cloud Platform | Microsoft Azure | **Applications** | Infrastructure

			
<b>.NET</b>	<b>.NET MVC Web API</b>	<b>.NET core</b>	<b>AIOHTTP</b>
A developer platform with tools and libraries for building web, mobile, desktop, games, IoT, cloud, and microservices.	Web application framework developed by Microsoft, which implements the model-view-controller pattern for building web applications.	The cross-platform successor to .NET Framework for building websites, services, and console apps.	Async HTTP framework for Python with support for client and server management.
<a href="#">Documentation</a>	<a href="#">Documentation</a>	<a href="#">Documentation</a>	<a href="#">Documentation</a>

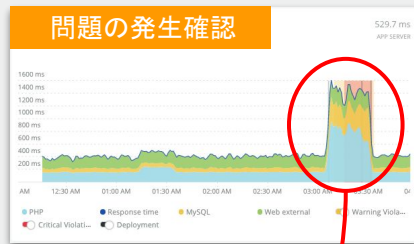
# New Relic One の全体像と関係性



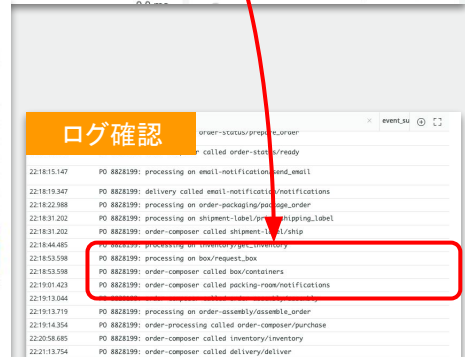
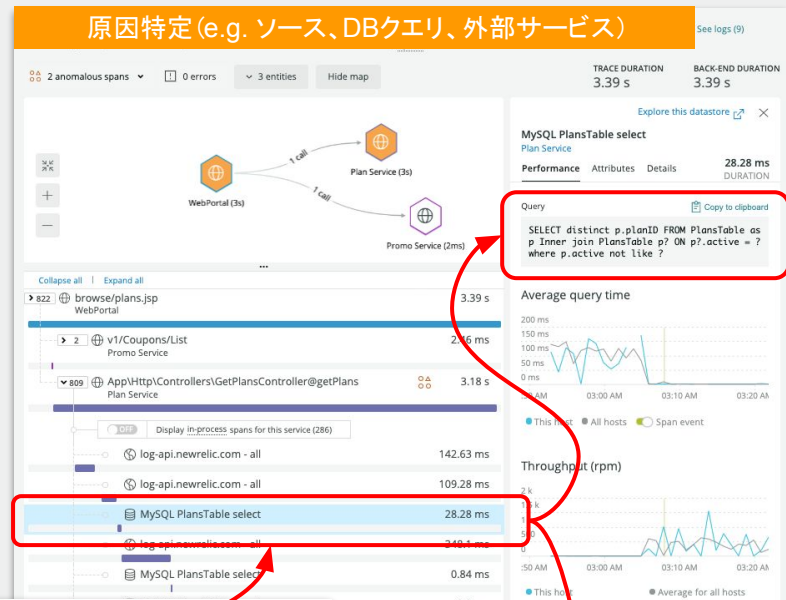
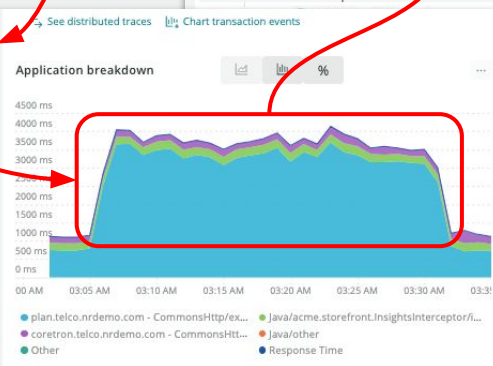
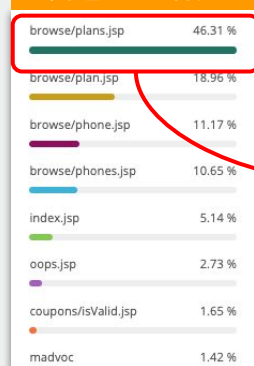
## 問題解決の迅速化

インフラからクライアントサイドまで、システムのスタック全体に渡って一つのプラットフォームで迅速に問題解決ができます

ソースコードやDBクエリの詳細なレベルまで原因特定ができ、根本解決が可能になります



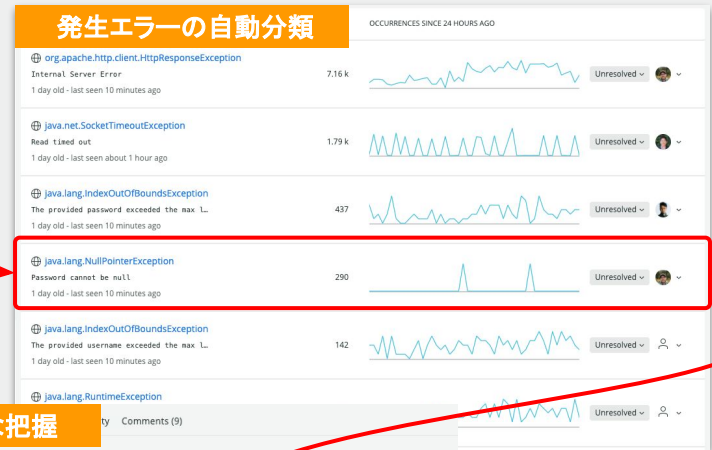
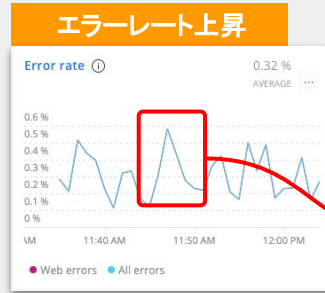
### 問題のAPI特定



## 問題解決 の迅速化

フロントエンドからバックエンドのシステム全体のエラーを捕捉し、コンテキスト(スタックトレースやエラーログ)の分析をシームレスに実施し、原因特定と解決を迅速化

チーム内でのコラボレーションを記録し、エラー分析にかかるコミュニケーションを効率化



### エラー発生コンテキストの迅速な把握

org.apache.http.client.HttpResponseException  
Internal Server Error

First seen 1 day ago  
Last seen 8 minutes ago

account: Demotron V2 accountid: 1606862 language: java

Occurrences 7.16 k OCCURRENCES

180  
160  
140  
120  
100  
80  
60  
40  
20  
0

4 1M Sep 14, 03:00 PM Sep 14, 09:00 PM Sep 15, 03:00 AM Sep 15, 09:00 AM

● Error count

Key Attributes

transactionName	WebTransaction/JP/loops.jsp	100.0%
transactionUIName	/oops.jsp	100.0%

Timestamp: Sep 15, 12:07:13 PM GMT+09:00  
URL host: webportal.telco.nrdemo.com  
URL path: /coupons/9f4a802a-e923-44ca-a4b1-35b8dc05...

Show only errors with stack trace

#### Stack Trace

Error message: Internal Server Error

```
...tractResponseHandler.handleResponse(AbstractResponseHandler.java:69)  
...apache.http.client.fluent.Response.handleResponse(Response.java:98)  
...g.apache.http.client.fluent.Response.returnContent(Response.java:98)  
...refront.serviceproxy.ServiceProxy.getResponse(ServiceProxy.java:98)  
...serviceproxy.CouponServiceProxy.isValid(CouponServiceProxy.java:98)  
...on.ValidateCouponAction.validateCoupon(ValidateCouponAction.java:98)  
sun.reflect.GeneratedMethodAccessor79.invoke(Unknown Source)  
...atInMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)  
java.lang.reflect.Method.invoke(Method.java:562)  
jodd.mvc.ActionRequest.invokeActionMethod(ActionRequest.java:562)
```

LOGS

Timestamp	Message
12:07:10.901	ValidateCouponAction.validateCoupon: Validating coupon...
12:07:11.092	Checking headers...
12:07:11.093	ValidateCouponAction.validateCoupon: Adding X-TELCO-USERID: 91070...

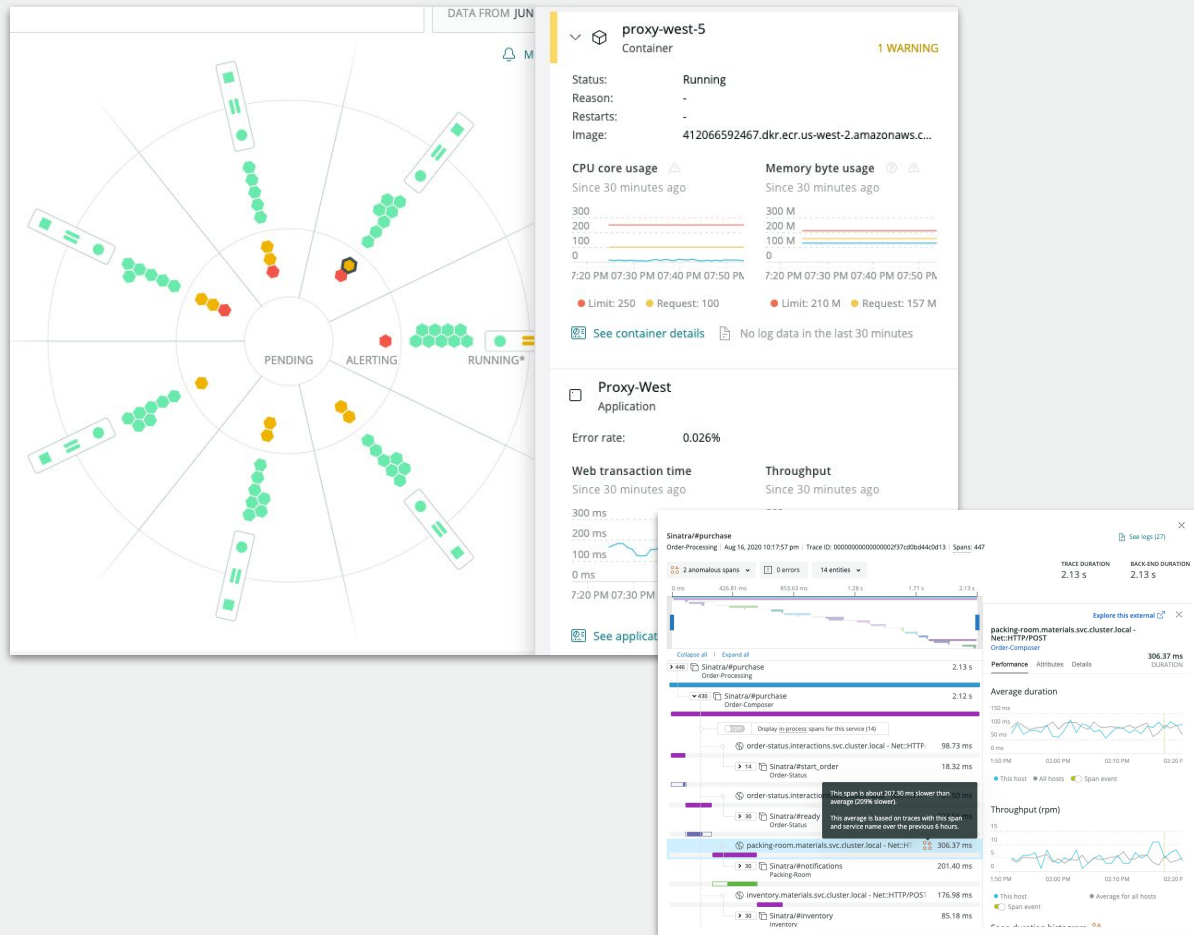
Demo User commented:  
Sep 8, 06:50 PM  
Lo he solucionado con un rollback  
Jul 19, 05:14 PM  
Comment deleted by user.  
Jun 24, 08:59 AM  
hey ab can you look at this - i think it is X  
Jun 24, 01:41 AM  
Comment deleted by user.  
Jun 24, 12:32 AM  
Comment deleted by user.

Show more (5)

## モダンアーキテクチャの運用効率化

Kubernetesをはじめとするコンテナ、サーバーレスなどモダンアーキテクチャも効率的に運用できます

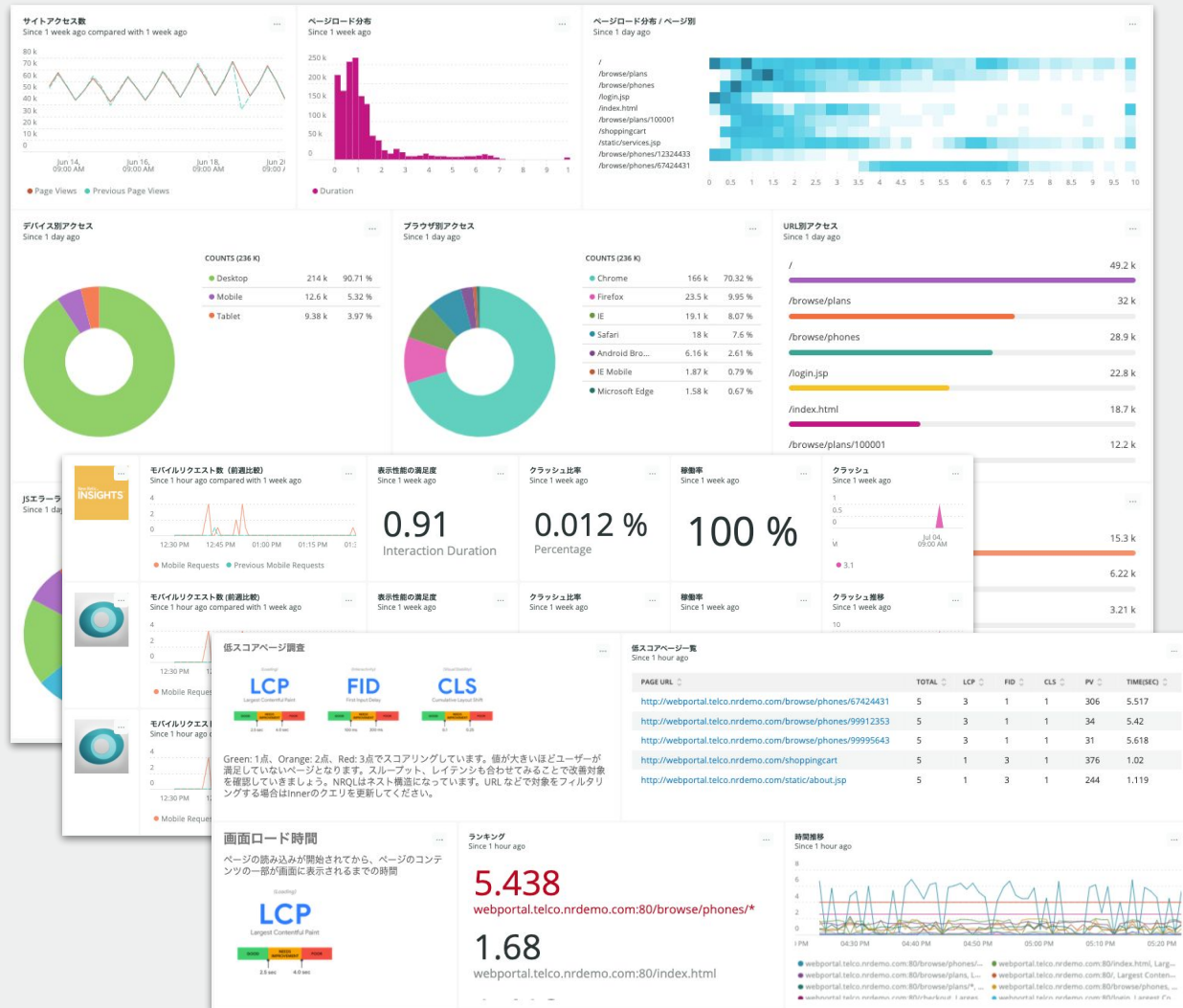
成熟していない領域に対する心理的安全性を確保して、モダンテクノロジーによる効率向上を最大化



## 顧客体験の把握

ヒアリング頼みだったユーザーサイドの問題を把握し、解決することができます

ユーザーの利用状況・環境を踏まえた改善、投資判断を行うことができます

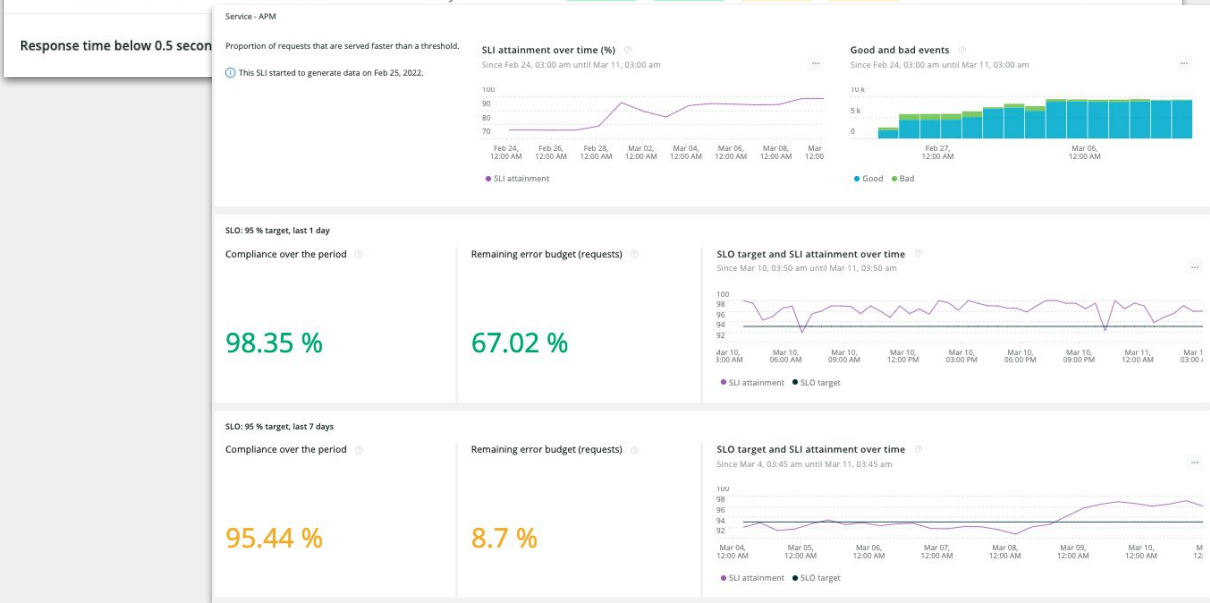


## サービスレベルの把握

ユーザーから見たサービス品質をリアルタイムに把握し、ユーザーに甚大な影響が出る前に問題(の兆候)にプロアクティブに対応できます

サービスレベルと目標値に基づき、開発と品質確保のバランスを定量的・建設的に判断できます

Name	Associated entity	SLO	Last 2 hours...	Last 24 hours...	Last 7 days...	Last 28 days...	Last 24 hours...	Last 7 days...	Last 28 days...
Cumulative layout shift	WebPortal Browser application	98.5 % 7 days	98.7 %	98.7 %	98.7 %	98.69 %	13 %	14 %	13 %
First input delay below 300ms	WebPortal Browser application	95 % 7 days	95.68 %	96.05 %	96.47 %	96.24 %	21 %	29 %	25 %
Largest contentful paint is less t...	WebPortal Browser application	95 % 7 days	93.94 %	94.17 %	94.22 %	94.03 %	0 %	0 %	0 %
Latency	Billing Service Service - APM	95 % 7 days	99.72 %	99.33 %	99.35 %	99.29 %	87 %	87 %	86 %
Response time below 0.1 sec an...	Login Service Service - APM	99.99 % 7 days	100 %	100 %	100 %	100 %	100 %	98 %	99 %
Response time below 0.5 seconds	WebPortal Service - APM	94 % 7 days	96.09 %	95.66 %	95.33 %	95.41 %	28 %	22 %	24 %



ドキュメント:

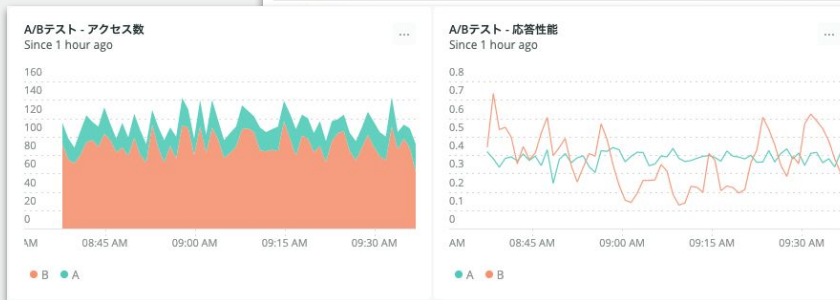
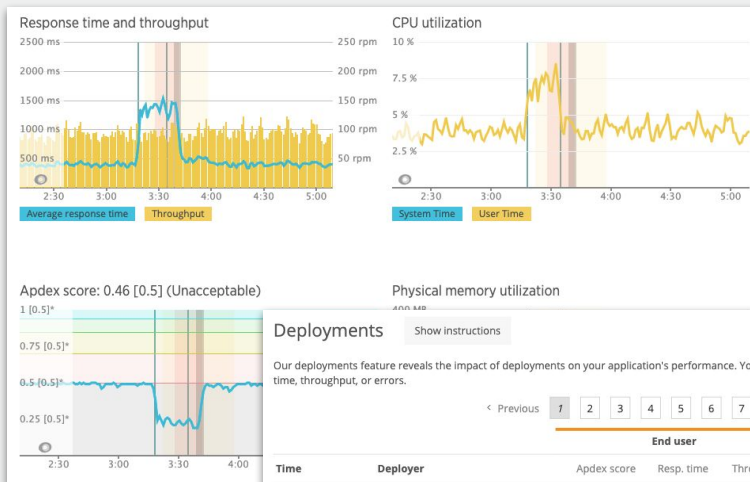
<https://docs.newrelic.com/jp/docs/service-level-management/intro-slm/>

## リリースの迅速化 と安定化の両立

リリースの影響を即座に把握できるため、アグレッシブな変更を実現できます

リリース毎の品質を管理可能になるため、開発プロセスの健全性を把握できます

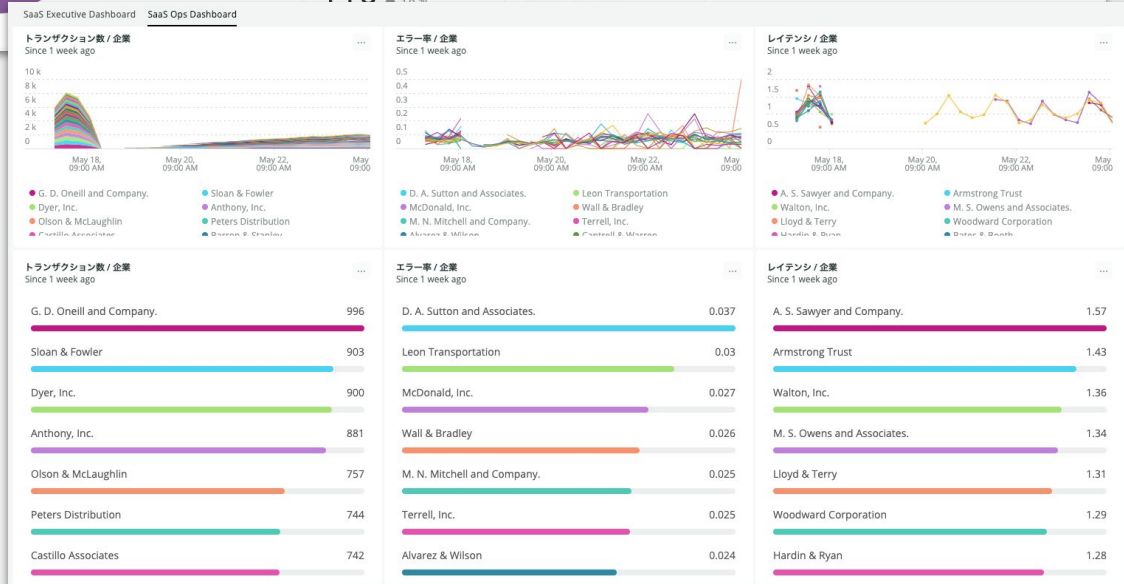
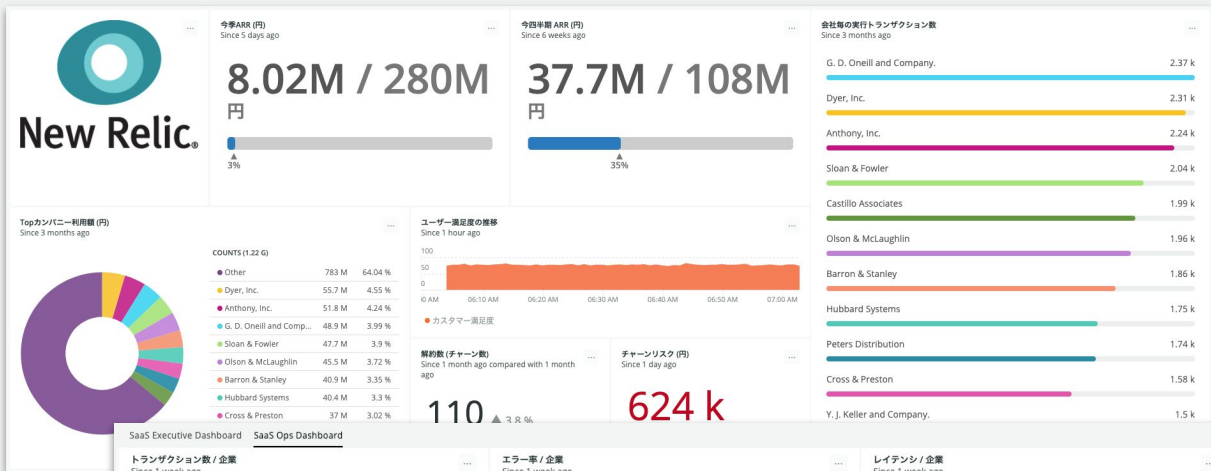
開発・テスト段階からの採用(シフトレフト)により手戻り防止、リリース品質を向上



# システム特性に応じた 深い分析

テナントIDやユーザーIDなど  
システムの特性に応じた属性を軸に  
精緻な分析が可能

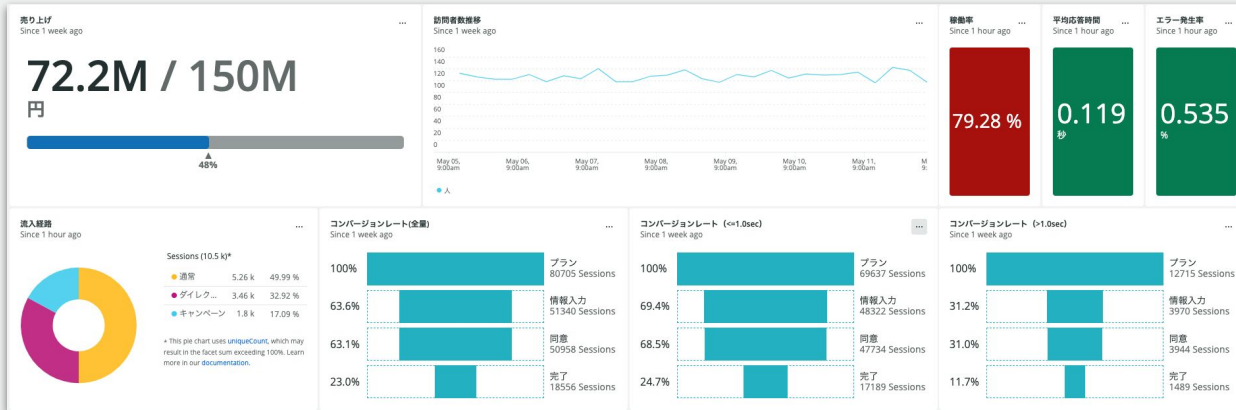
特定条件や特定顧客で発生している  
問題を見逃すことなく把握



## ビジネスKPI の計測と改善

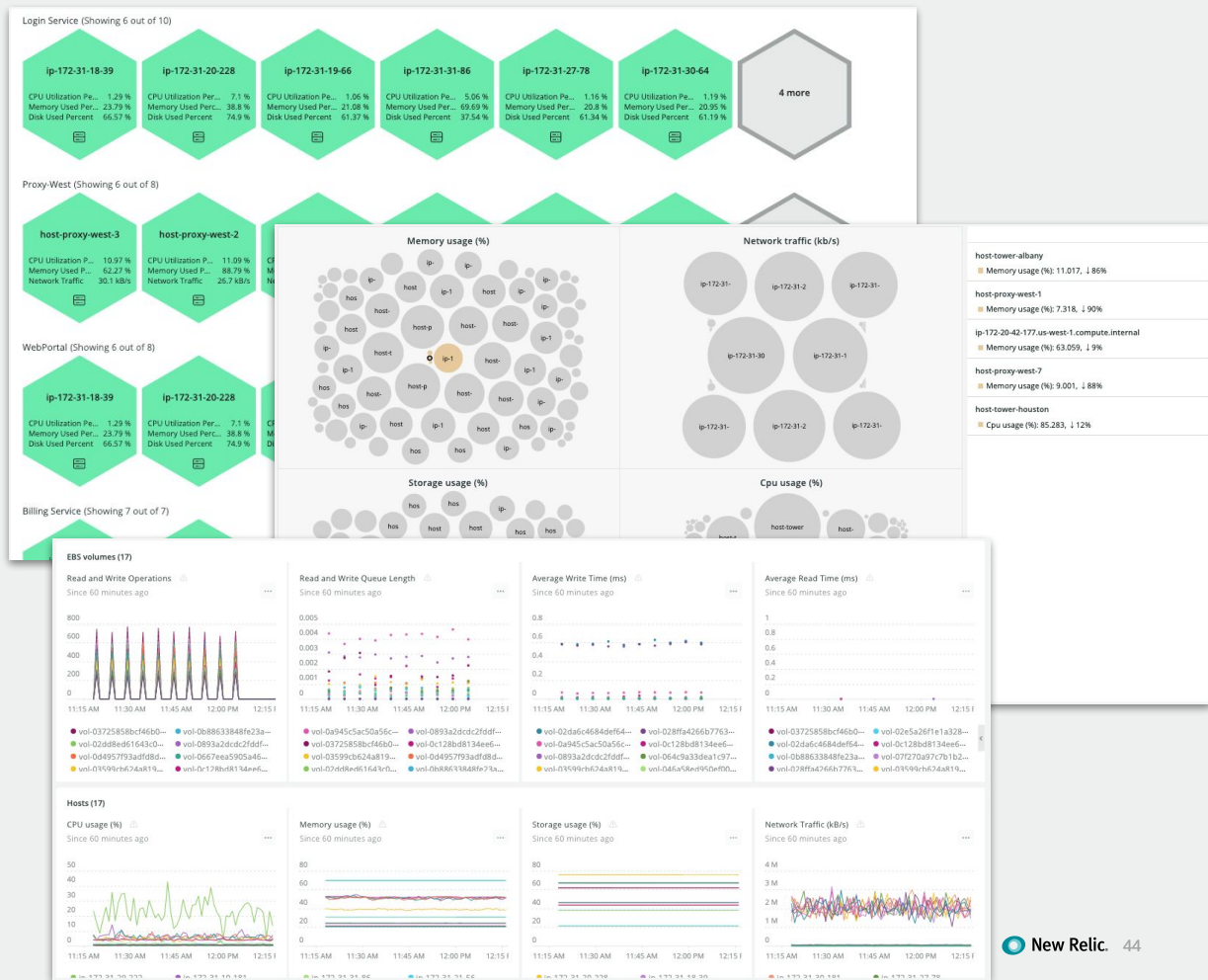
パフォーマンス低下や障害などシステムの問題の影響を、ビジネス視点から把握できます

システムの変更や改善によるビジネスへの効果を明らかにし、システムの継続的改善による収益向上につなげられます



# インフラ全体の 状態俯瞰

インフラの全体の稼働情報を俯瞰し、アプリケーションのダウンにつながる問題やその傾向をプロアクティブに把握



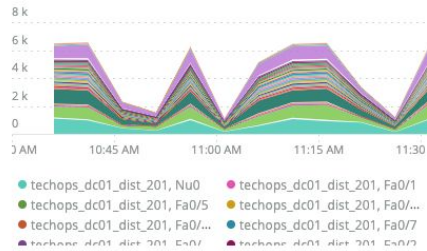
## ネットワークの問題切り分けの迅速化

システム全体の障害につながる **ネットワーク機器の致命的な問題をいち早く検出**

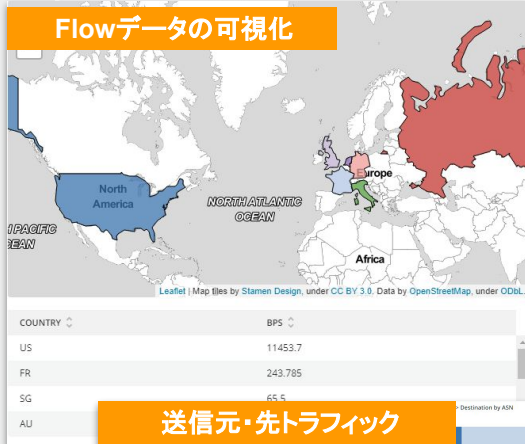
プロトコルや送信元・先ごとのトラフィック把握により、**アプリケーション視点でネットワークの問題を特定**。ネットワーク管理者への問い合わせ前に切り分け可能

### ネットワーク機器の障害、通信量

DEVICE NAME	INTERFACE NAME	INTERFACE STATUS
techops_branch01_edge	Et0/0	up
techops_branch01_edge	Se0/0	up
techops_branch01_edge	Se0/0.1	up
techops_branch02_edge	Fa0/0	up
techops_branch02_edge	Fa0/1	up
techops_branch02_edge	Se0/0	up



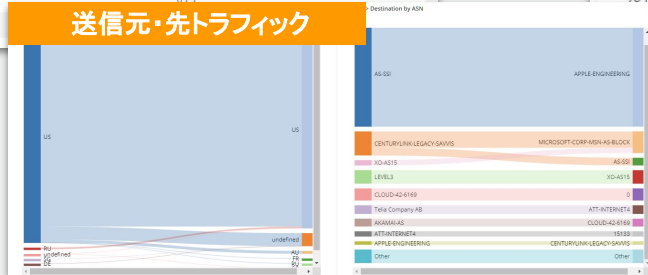
### Flowデータの可視化



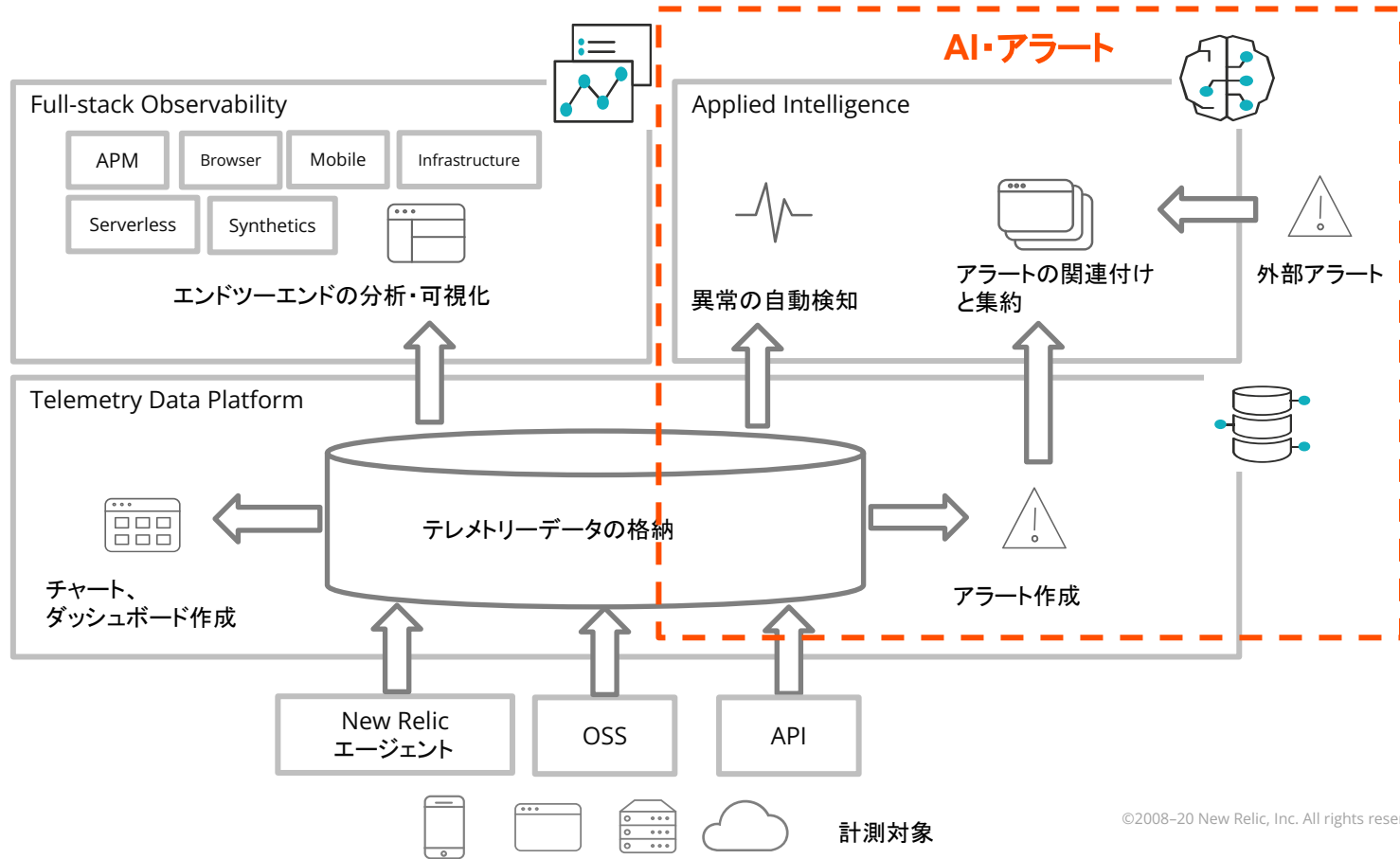
### プロトコル別通信量



### 送信元・先トラフィック



# New Relic One の全体像と関係性





## その他ユースケース

---

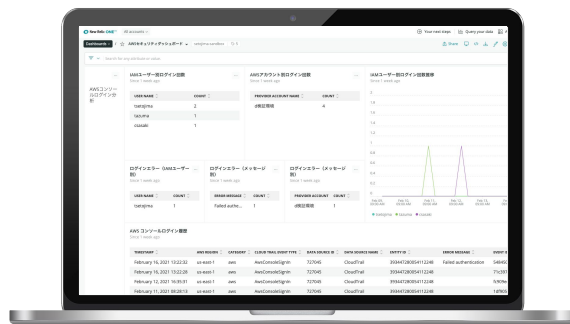
# AWS大規模運用時特有の課題解決にも利用可能

## AWS/オンプレ 全システム管理統合



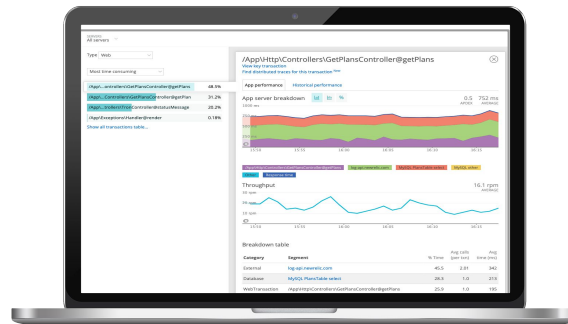
インフラ管理運用を効率化  
全サービス横串で評価可能に

## AWS統制を可能にする マルチアカウント管理



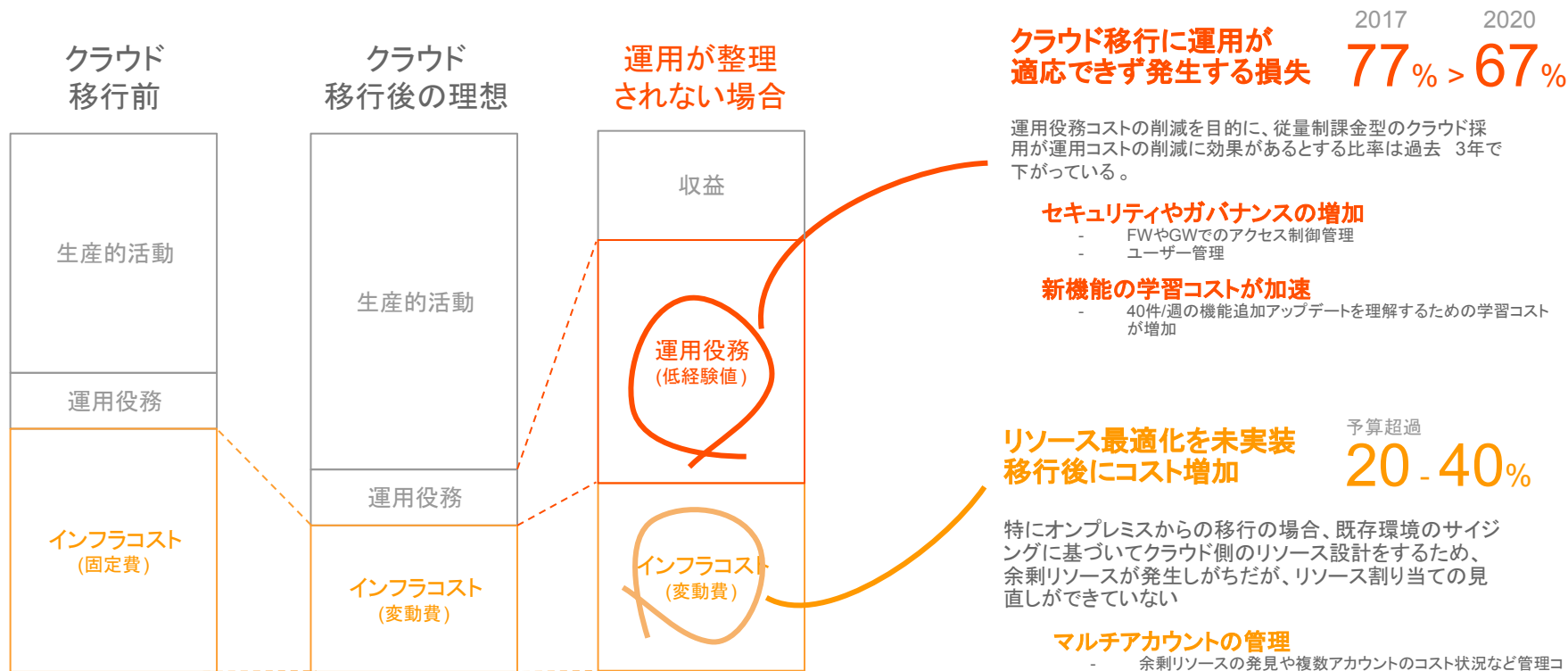
“守り”に割く工数を抑えて  
生産的活動に工数を割り振る

## システム全体の 問題対応を高速化

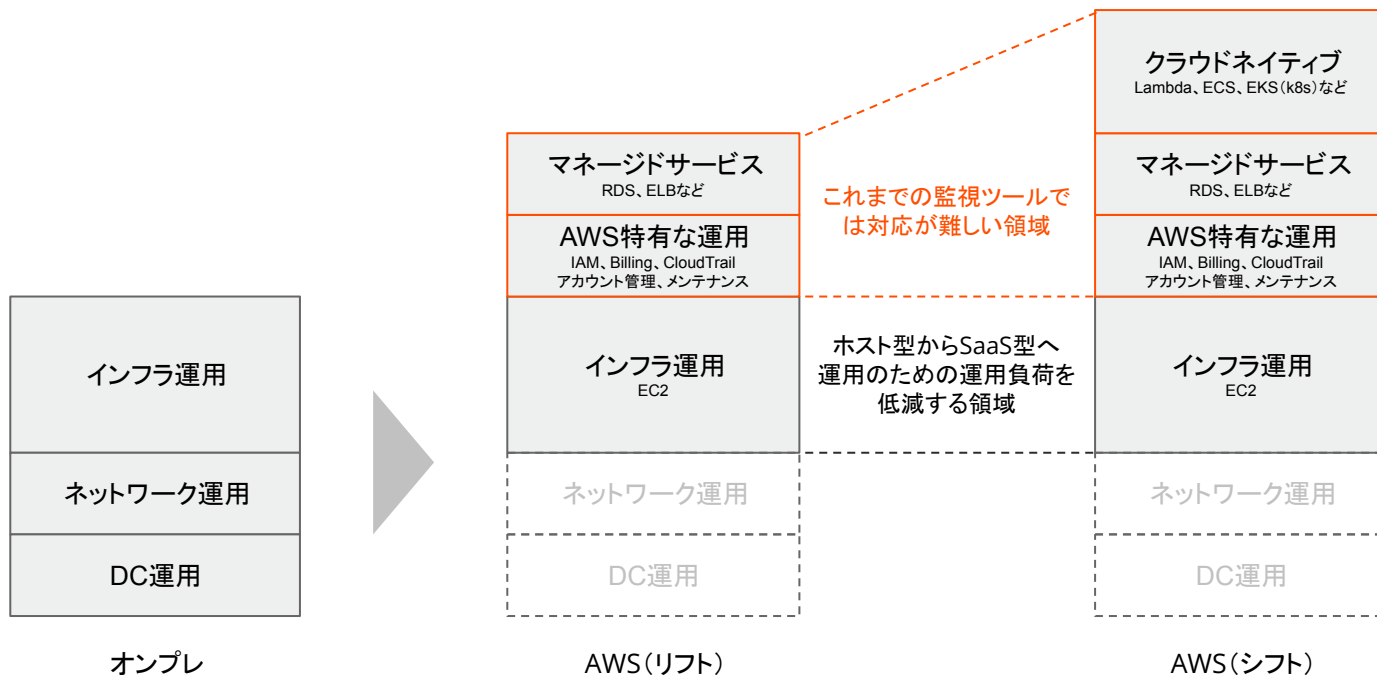


アプリのパフォーマンスを見ながら  
安全にリソース最適化を実現

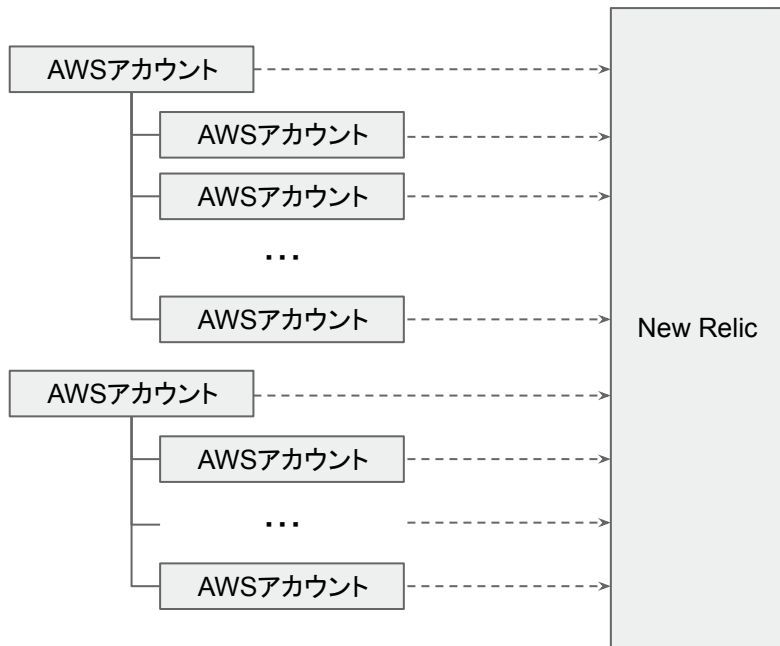
# クラウド管理には運用変革も必要



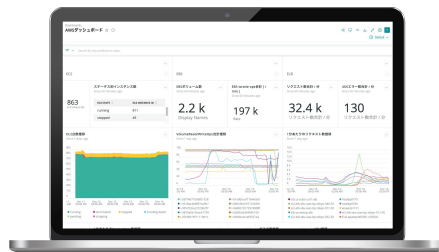
# クラウドマイグレーションに合わせたインフラ運用



# AWSと連携しAWSアカウント横断で分析・可視化

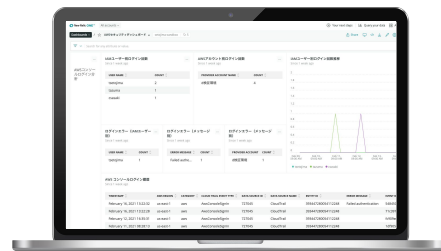


**AWS/オンプレ  
全システム管理統合**



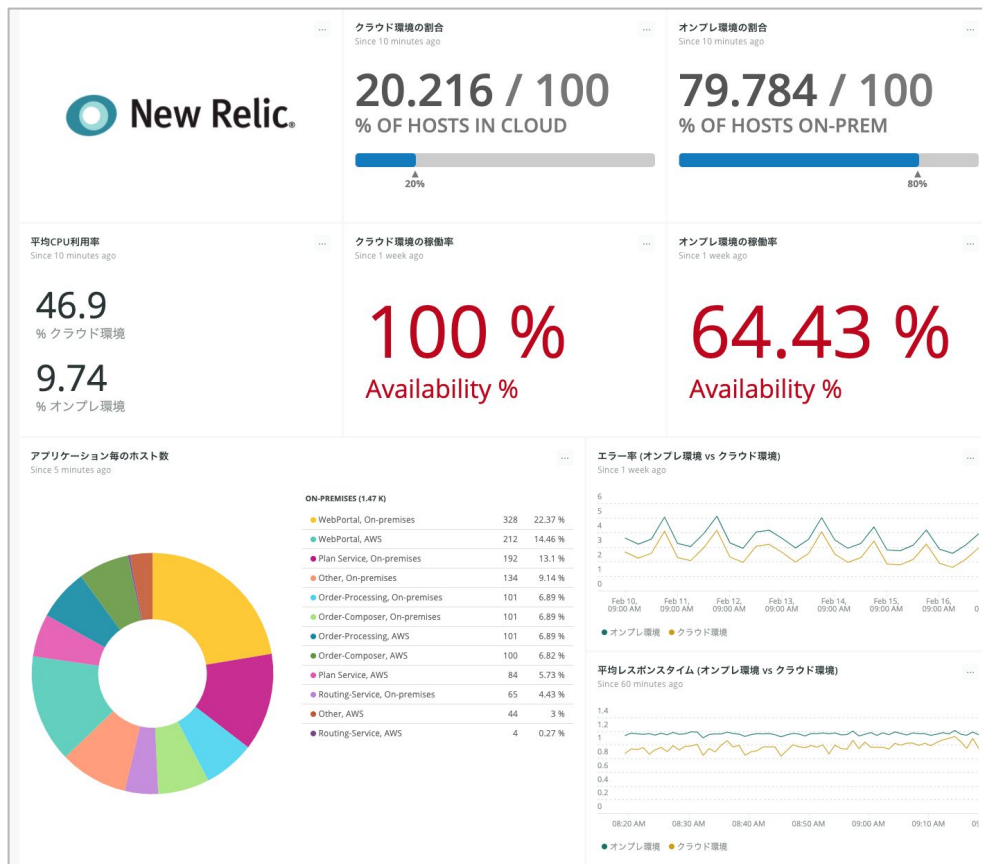
インフラ管理運用を効率化  
全サービス横串で評価可能に

**AWS統制を可能にする  
マルチアカウント管理**



“守り”に割く工数を抑えて  
生産的活動に工数を割り振る

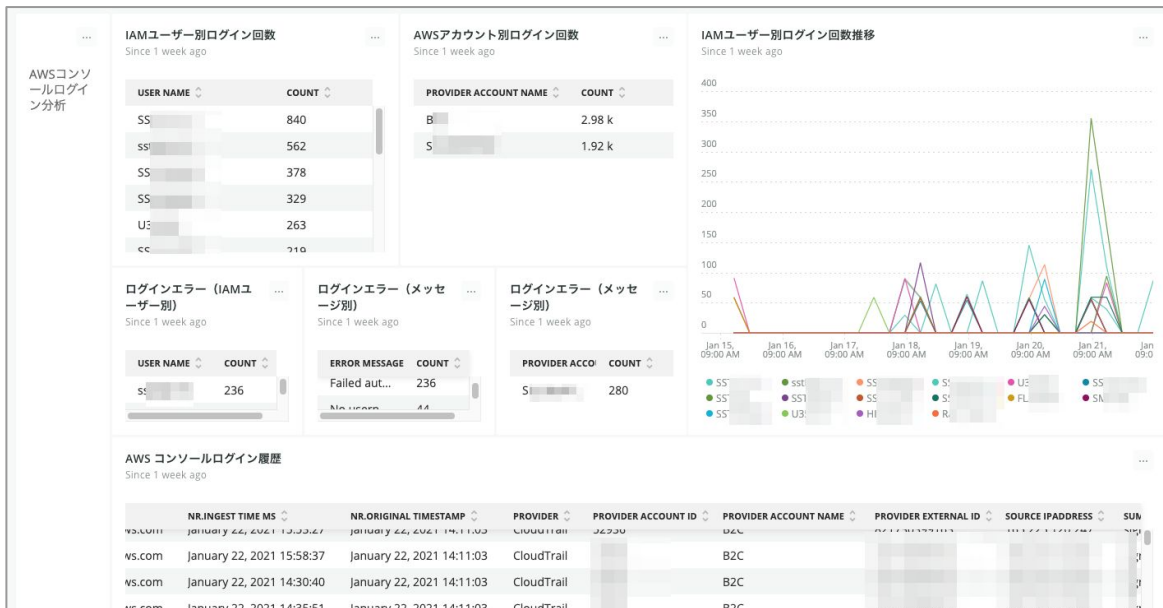
# AWS/オンプレ 全システム管理統合



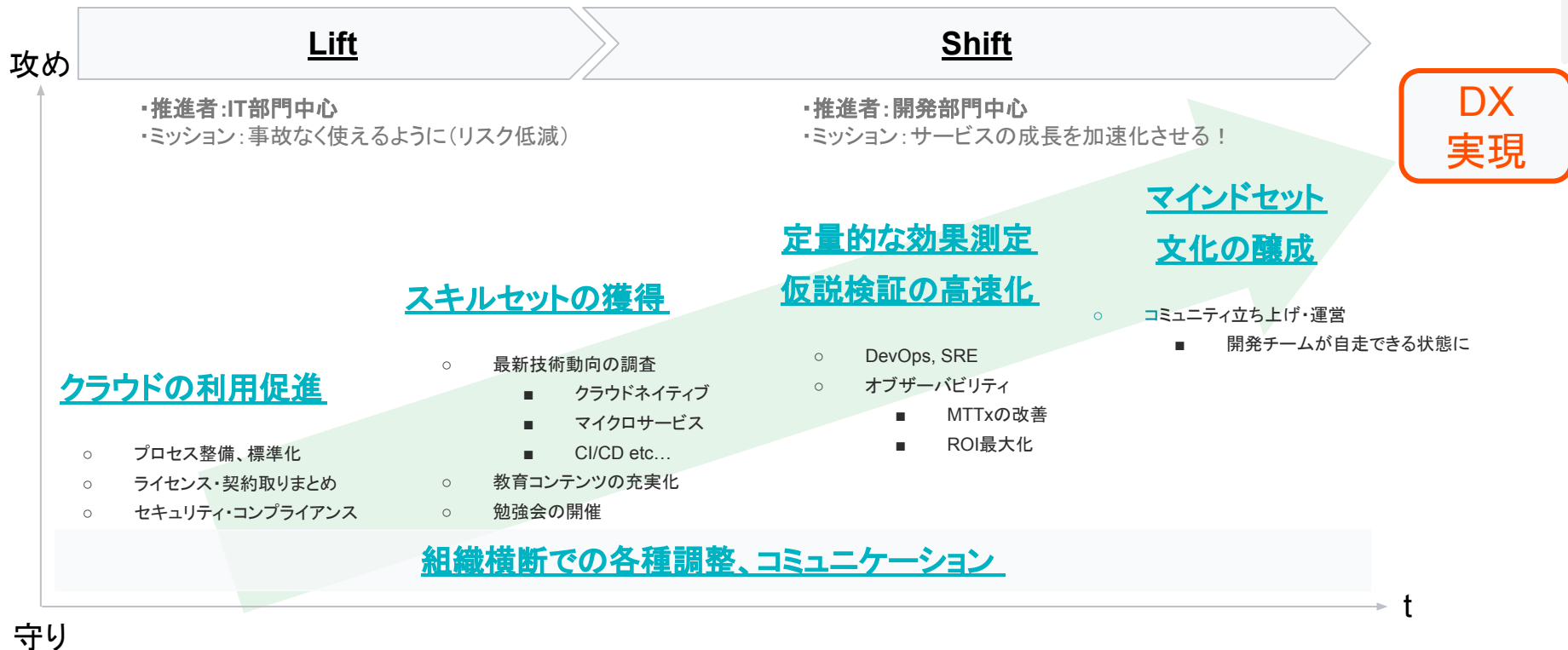
# 例) ④すべてAWSアカウント横断で利用状況可視化



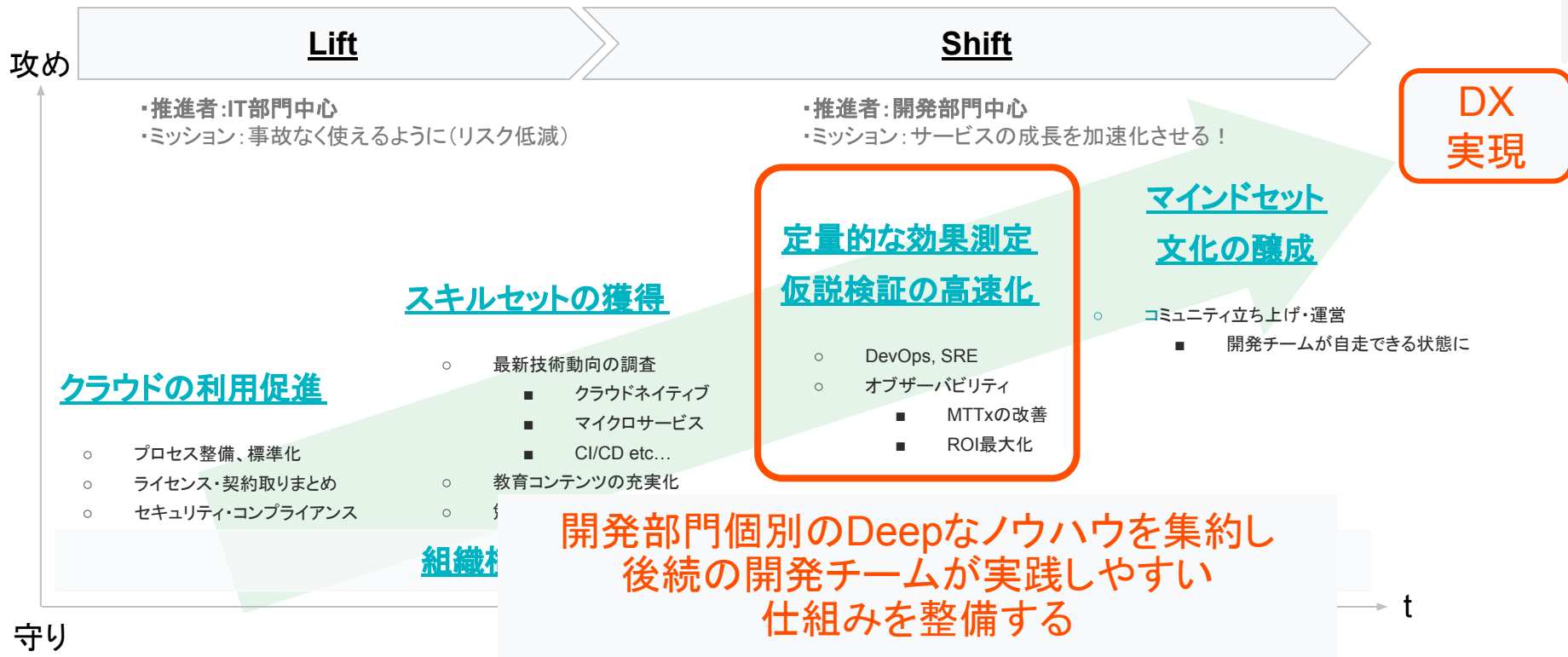
# 例) ⑤ガバナンス関連情報の可視化・分析



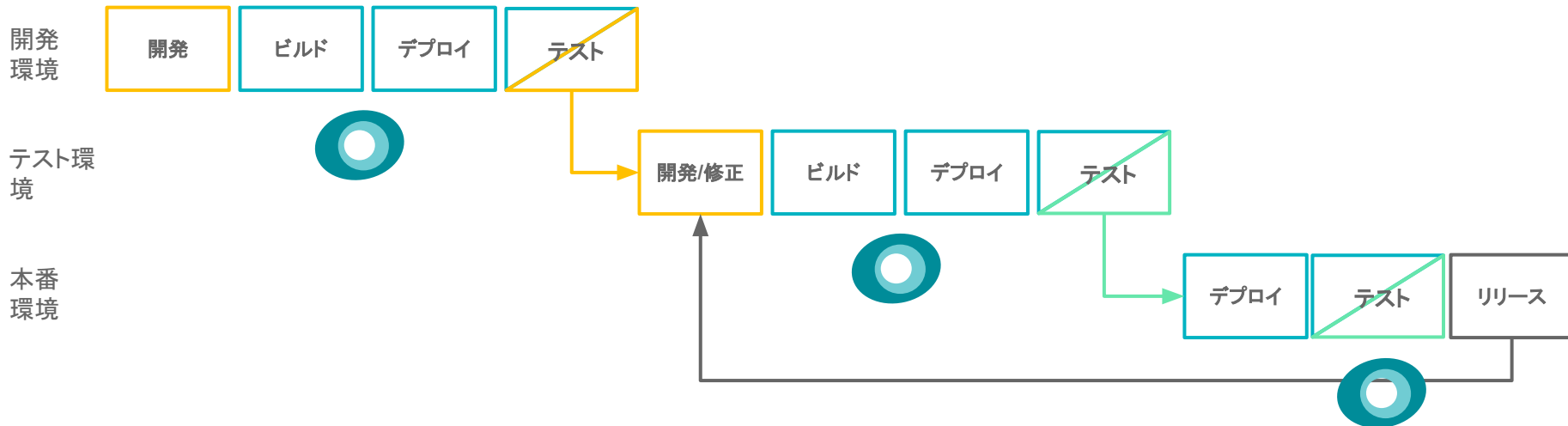
# CCoEジャーニー：攻めのCCoEへのチャレンジ



# CCoEジャーニー：攻めのCCoEへのチャレンジ



# 開発・テストフェーズでのNew Relic活用



- ... 開発者
- ... CI/CD
- ... 品質評価

テストフェーズやリリース後の  
問題早期発見で手戻り少なく、  
さらにデプロイごとのアプリ性能を観測し信頼性向上

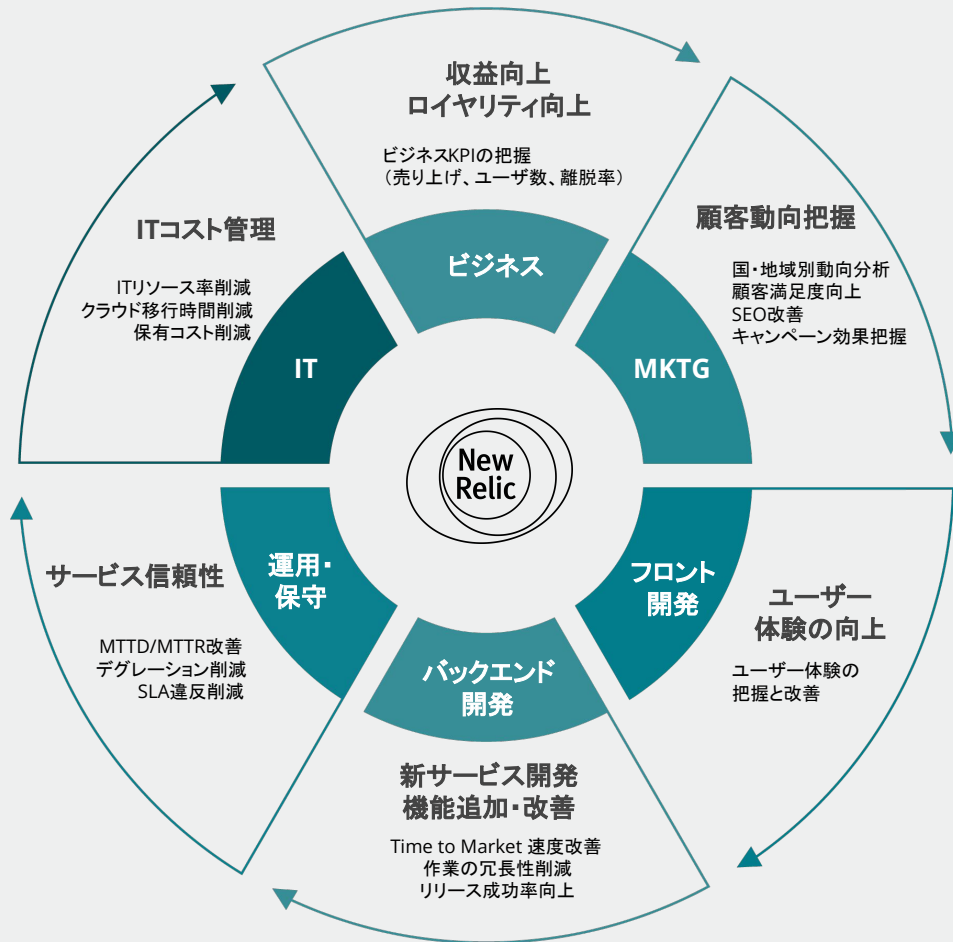
# テストフェーズにおけるNew Relic活用

- **非機能面モニタリング**
  - テスト初期フェーズから継続的に非機能面(アプリ内部挙動、性能、エラー原因など)特定可能
  - スループットとレスポンス、DBアクセス、CPUの相関を確認
- **テストの証跡確認**
  - メンバーやオフショア・ベンダーのテスト状況(エラー発生状況等)を確認可能
- **性能テスト時の活用**
  - 負荷テスト時のアプリ挙動、その際のボトルネックを未然に検知
- **バージョンごとの比較**
  - CustomAttributeにバージョンやその他情報を入れることで、バージョンごとの性能やエラー率などの比較が可能
- **リリース判断**
  - 性能やエラー率を元にしたリリース判断

# アジャイルな 組織・文化への変革

データやプラットフォームを共通言語として、組織全体として改善活動を行う文化やプロセスを醸成

組織内のコミュニケーションオーバーヘッドを減らし、社員の生産性を向上



# Why New Relic?

---

# New Relicのシンプル・親切的な課金体系



## Full-Stack Observability

ユーザ数課金

有償ユーザは全機能を利用可能  
Dashboardのみの利用は無償

ビジネスメトリクスと関連性のない請求から脱却。  
管理が容易な価格体系

**¥38,390** Per 有償 user / month



## Telemetry Data Platform

送信データ量課金

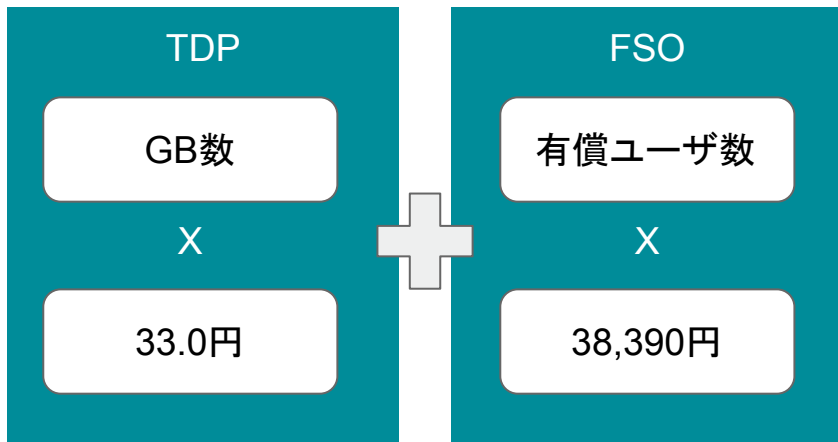
Metric/Event/Log/Trace  
すべて一律のGB単位課金

テレメトリーデータのストレージ費用は業界最安値に設定 ( ¥27.5 /GB\*month )  
固定費を限界まで圧縮する課金体系

**100 GB Free | ¥33.0** / GB Ingested \* month

# 一般的な費用の算出方法

<計算式>



<価格シミュレーション例>

購入製品	数量	月換算	年額
TDP	2100GB	66,000円	792,000円
FSO	5 users	197,450円	2,369,400円
合計:		263,450円	3,1614,00円

\*TDP: 100GBまで無料。101GBから課金対象

\*FSO: 最低1 Userからのご契約でOK

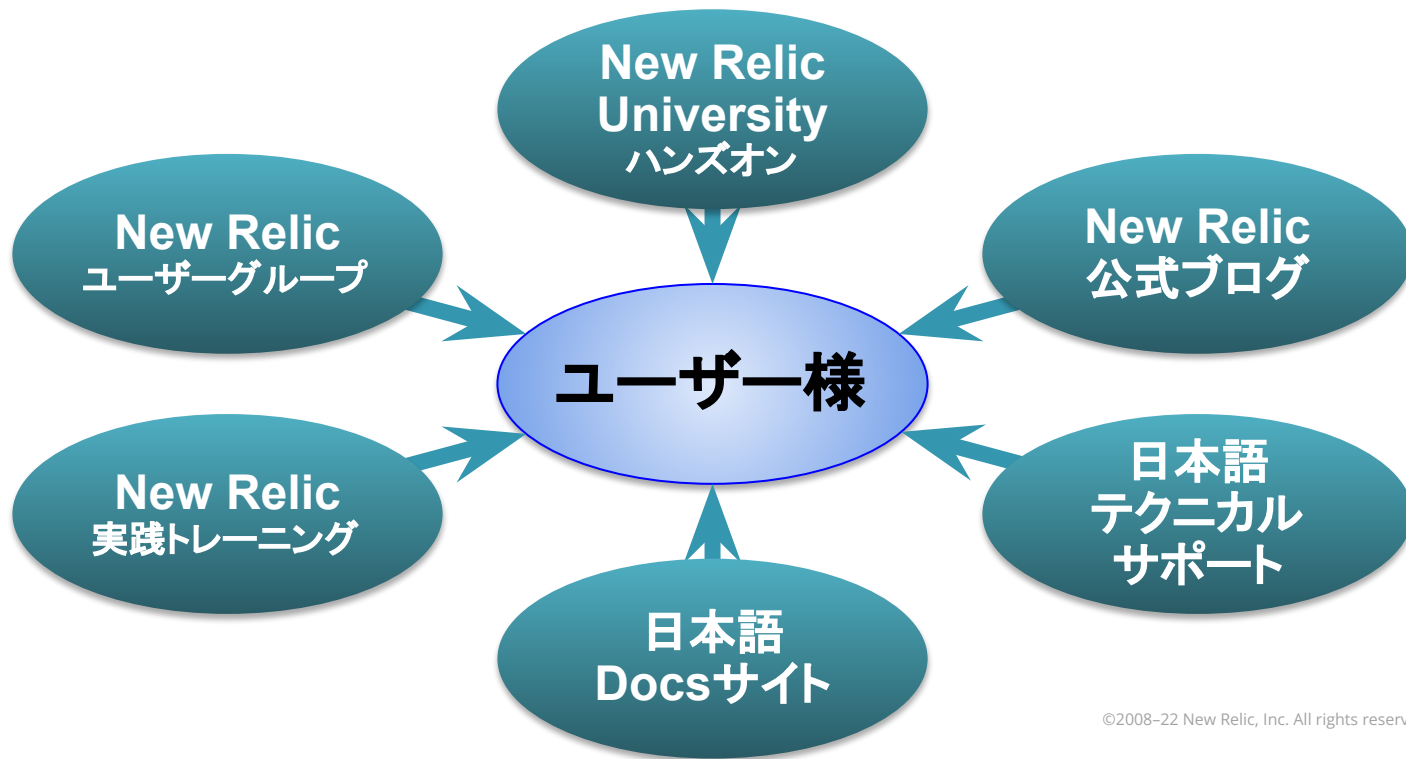
\*表示は税抜き価格



**年間一括請求**  
超過分自動請求なし（ご利用増時はお客様と協議の上、適切な契約内容をご相談）

# 全方位でサポートする豊富な技術支援体制

お客様のパートナーとして、ワンチームとして New Relic をフル活用頂き、ビジネスへ貢献できるような多彩な技術サポートメニューを提供させていただきます。



# お客様を支援する安心のサポートチーム

ソリューション  
コンサルタント(SC)



コンサ

New Relic  
ユニバーシティ



テクニカルサポート



カスタマサクセスマネジャ  
(CSM)



## PoC支援

ベストプラクティス提供  
課題と検証項目整理  
テクニカルサポート  
・Agentインストール  
・データ可視化  
・サービスマップ  
・各種QA

検証結果検討

## オンボーディング支援

キックオフ  
必要データ収集とトレーニング  
実践的な知識習得  
実業務プロセスへの落とし込み

導入効果確認と改善

## 各種トレーニング

ハンズオントレーニング  
各種ウェビナー

## 専門のテクニカルサポート

チケットによるQAやトラブル対応  
各種ドキュメント提供

## 確実な運用定着と成熟度向上を支援

サクセスプラン策定とQBR  
最新ベストプラクティス提供  
独自トレーニング実施  
Slack等の独自テクニカル支援  
成熟度を継続的に向上

# まとめ

---

# まとめ

- マネージドサービス、オートスケール、コンテナ、k8s、サーバーレス、、、  
**パブリッククラウドが主流になり**  
**従来の“監視方法”が適用できなくて困っている**  
→ **New Relic**を利用し、**AWS/GCP/Azure/オンプレのハイブリッド環境をまとめてモニタリング**
- マイクロサービス化、API化、React/Angular/Vue、モバイル、、、  
**アプリケーションがより複雑になり、障害原因特定が難しくなった**  
→ **オブザーバビリティを取り込み、常にシステム全体を観測**
- Why New Relic?  
**全てに入れると費用爆発。組織への浸透やビジネス改善に通ずるノウハウがない**  
→ **New Relicはエンジニア数課金**  
→ **万全のサポート体制**

# NRUG

躍動!!

Observability for all.



## New Relic User Group Vol.3

テーマ: New Relic初心者向け

日時: 2022年6月15日(水) 16:00 - 18:00

会場: オンライン(Zoom)

参加申し込み: <https://nrug.connpass.com/event/247057/>

# New Relic 実践入門

希望者に無償提供中

ついに発売された New Relic の全てを理解できる 330 ページにわたる技術書籍。オブザーバビリティの基本から New Relic One の基本機能、さらには 16 のオブザーバビリティ実装パターンまで含めた、初心者から応用を理解したい上級者まで対象にした New Relic のパーフェクトガイドブック。

無償提供希望は以下のGoogle Form から

<https://forms.gle/jqiYmWRYt8Hf1nHk8>

[紹介元情報には partner-leaningと入力ください](#)



Thank You