

クラウドアプリやインフラストラクチャを監視するための7つの要件

クラウドコンピューティングによりすべてが変化していく中、アプリケーションとインフラストラクチャの監視もその例外ではありません。[Amazon Web Services \(AWS\)](#)、[Microsoft Azure](#)、[Google Cloud Platform](#) などの最新鋭のクラウドプラットフォームや、[Pivotal](#) などのツールが提供するフレキシビリティ、スケール、サービス、従量制の料金体系によって、企業におけるアプリケーションの構築方法や、それらのアプリケーションを対応インフラにデプロイする方法は一変しました。

単一のデータベースとだけ通信する単体のアプリをベアメタルや仮想マシンのホストに何か月もかけてデプロイするのではなく、何種類ものマイクロサービスやコンテナをオンデマンドでデプロイする方法を選ぶ企業が次第に増えてきています。また、以前は静的なサーバーであったコンピューティングインスタンスは、オンデマンドで(使用量に基づいて)デプロイされ、シームレスに負荷分散される動的なインスタンスへと変化してきています。企業は、1つの静的なコンテンツ配信ネットワーク(CDN)に依存するのではなく、さまざまなリージョンを受け持つ複数のCDNを日常的に扱うようになりました。ウェブデベロッパーは、クライアント側(エンドユーザーのブラウザ)にかかる負荷を増やして、飽きのこないユーザーエクスペリエンスの実現に役立てています。変化の例は他にもまだまだあります。

こうしたすべての変化に対応し、クラウドベースのアプリケーションや動的なインフラストラクチャを効率的に監視するには、監視プラットフォームも変化する必要があります。従来のオンプレミス(シングルテナント)型のマネージドホスティング(別名「疑似SaaS」)による監視ソリューションで、動的なインフラストラクチャや、クラウドでホストされるアプリケーションを監視しようとするのは、ゴーカートで高速道路を走ろうとするようなものです。それではとても追いつけません。

ここでは最新鋭のクラウドベースアプリケーションやインフラストラクチャを適切に監視するために必須の、機能や性能7つをご紹介します。

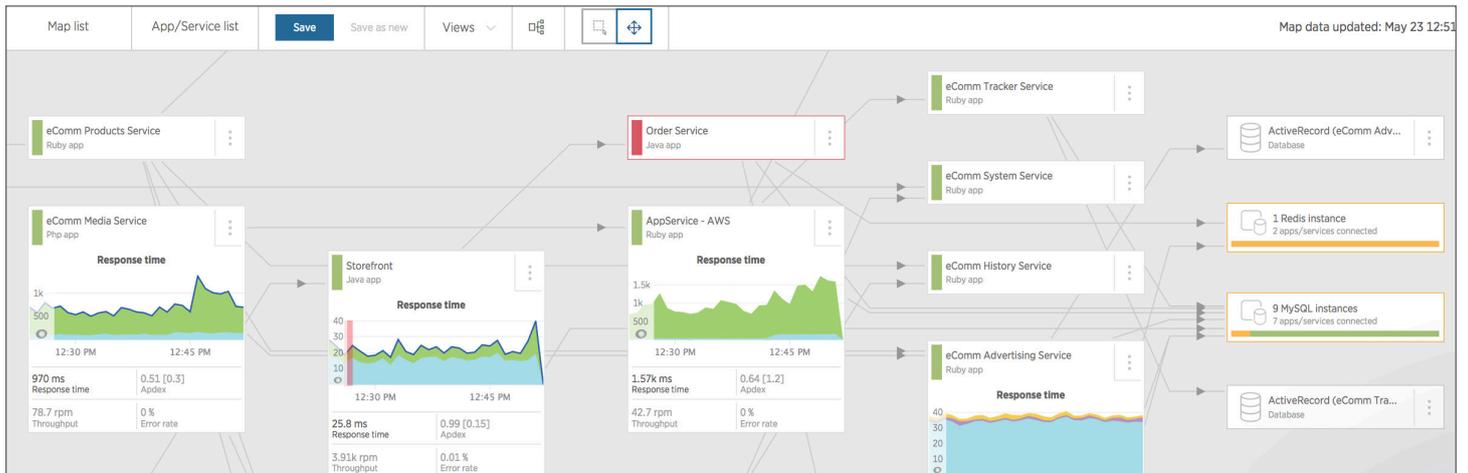
1. イベントをキャプチャしてクエリを行い、データを集計するだけでなくトレースも行う機能

ユーザーがオンラインで何かを購入するときに行われる処理では、大量のHTTPリクエストが生成されます。エンドツーエンドのクラウド監視を正しく行うには、ユーザーが購入を完了するまでに作成されるすべてのHTTPリクエスト(個々の商品の詳細ページ、カートに追加のPOSTリクエスト、カートの確認ページ、支払いと配送のPOSTリクエスト、最後の「注文を送信」ページなど)を正確に把握する必要があります。クラウドへのデプロイでは、リクエストはアプリケーションや、トランザクションを完了するためのマイクロサービスやキューに達するまでに、多数の動的なコンポーネント(ブラウザ、モバイル端末、CDN、ロードバランサなど)で処理されるため、未処理のトランザクションデータをキャプチャする機能が特に重要となります。

ボトルネックをすばやく特定し、クラウドネイティブシステム内のさまざまなコンポーネント間の関係を把握できることは、監視ソリューションの大きな差別化要因となります。New Relicを使用すると、トランザクションに関連する指標を集計するだけでなく、それぞれのコンポーネントの正確な応答時間を確認することができます。これにより、購入手続きで発生したエラーについてユーザーからサポート担当に問い合わせがあった際に、デベロッパー(またはサポート担当やDevOpsチーム)は、アプリケーションスタックのNew Relicで管理している階層すべての中から、エラー発生の原因となっているトランザクションのセットを正確に見つけ出すことができます。

New Relicでは、エラートレースやカスタム属性などの不可欠なメタデータを使用して、トレースやイベントのデータを強化できます。

ユーザーに固有の属性や企業に固有の属性(顧客のユーザー名など)から取得したデータを分割することで、改善すべき点の優先順位をつけやすくなり、顧客向けの最適化プランを加速することができます。New Relicの数多くの顧客は、専用に構築されたマルチテ



New Relic サービスマップには、相互に呼び出し合うアプリケーションとサーバー間の接続や依存性が表示され、各ノードの詳細なパフォーマンスデータを確認することができます。

ナントの分析データベース (別名「NRDB」) に格納されたこうした収集データに対し、SQL と似た **New Relic Query Language (NRQL)** を利用してクエリを実行して、ダッシュボードやアラートを作成しています。(詳しくは「**NRQL Alerts が New Relic データの利用方法を変える**」をご覧ください)。

2. サイト、アプリ、インフラストラクチャ間の接続を監視して可視化する機能

企業がクラウドに移行するという事は、大規模な単体のアプリをデータベースサーバーと通信させるということだけにとどまりません。それだけではなく、ロードバランサ、キュー、サーバーレスアーキテクチャ、アプリ、マイクロサービスなど、数が増え続けるコンポーネント間の複雑化した接続から発生する問題についても気を配らなければなりません。

クラウドの監視において重要なのは、トランザクションをトレースしてすべてのコンポーネント間の接続を確認できることです。また、個々のコンポーネントに不可欠で有用なデータを配信するための内部アプリケーションコードを深く理解できる、詳細な言語エージェントの存在も重要です。

約10年間に渡る、最新鋭のクラウド監視に実績を持つ New Relic では、7つの主要プログラミング言語に対応しています。個別の設定やコード変更は必要ありません。さらに、**New Relic APM サービスマップ**ではコンポーネント間の接続を視覚的に確認できるため、すべての接点におけるパフォーマンスの問題を追跡することができます。問題の特定後にトレースデータを使用すれば、問題を引き起こしている関連コードを見つけることができるため、迅速に問題を修正できます。

また、New Relic の革新的な新しい**ヘルスマップ**では、業界最先端の APM やインフラ監視データを統合して可視化できるため、運用チームや DevOps チームはアプリケーションの問題をすばやく把握してトラブルシューティングすることができます。

特に重要なのは、New Relic が監視するのはアプリケーションサーバーとインフラストラクチャ間の接続だけではないということです。クライアント側のモバイルやブラウザのアプリケーションのパフォーマンス全体を監視し、それがどのようにユーザーエクスペリエンスに直接影響しているかを確認することもできます。先述のとおり、ウェブデベロッパーはエンドユーザー側のクライアントにかかる負荷を増やしてきているため、バックエンドを監視するだけでは全体的な状況を把握できないことがあります。アプリケーションサーバー側が正常に見えていても、実際はクライアント側でフェイルオーバーやエラーが発生しているといった場合もあります。**New Relic Browser**、**New Relic Mobile**、**New Relic Synthetics**などを備えた **New Relic デジタルインテリジェンスプラットフォーム**は、エンドツーエンドのパフォーマンス全体をリアルタイムで実際にキャプチャできる数少ないソリューションです。

3. 主要クラウドプロバイダに対応したネイティブインテグレーション機能

クラウドベースのアプリやインフラストラクチャを扱う際は、多岐にわたるクラウドプロバイダが提供するサービスとのインテグレーションを1つの場所で監視できる機能が重要です。さまざまなプラグインを AWS (またはその他のクラウドプロバイダ) サービスごとにダウンロードしてインストールするのは非常に煩雑で望ましくありません。**New Relic Infrastructure**を使用すると、**サポートされる AWS 統合**をすべて1か所で監視することができます。これが、New Relic が「本物の SaaS 監視プラットフォーム」と呼ばれる理由です。New Relic Infrastructure には、Elastic Load Balancer (ELB)、EC2、ElastiCache など数々の主要なサービスに対する AWS インテグレーションのサポートが組み込まれているため、設定を1行変更するだけで、サポートされる AWS サービスの状況を監視できます。そして、弊社ではクラウド統合開発・支援を担当する専門チームを配置していることから、制作、保持、拡充といったクラウド統合向けのコミュニティ制作型プラグインに頼る必要がありません。



INTEGRATION NAME ^	DASHBOARDS	CREATE ALERT	EXPLORE DATA
Billing	Billing (Costs) Billing (Budgets)	API	
CloudFront		API	
DynamoDB		API	
EBS		API	
EC2			
ElastiCache	ElastiCache (Redis) ElastiCache (Memcached)	API	
ELB		API	

New Relic Infrastructure Pro で Lambda, DynamoDB, RDS, Cloudfront 等の普及しているサービスに AWS のネイティブ統合を行えるようになりました。

こうしたインテグレーションサービスは、New Relic 完全独自のサービスです。さらに、こうしたクラウド統合を介して使用可能なデータはすべて、**New Relic Insights** でキャプチャされ、ボタンを1クリックするだけでユーザーが簡単に分析できるようになっています。

なぜこのことが重要なのでしょうか? クラウドプロバイダの何千というサービスを利用していく中で、アプリケーションやインフラストラクチャの利用状況を理解するためだけにこうしたサービスとのインテグレーションを構築する作業に膨大な手間をかけることは、極めて非効率と言えるからです。

4. クラウド志向の価格

オンプレミス用に開発された監視ソリューションのホスト単位での課金体は、サーバーやコンテナが行き交うダイナミックなクラウドアーキテクチャの現実に適切に対応しているとは言い難いでしょう。常にダイナミックに変化するクラウド環境において、実際のインフラの稼働量に適切に対応した課金モデルが必要です。

だからこそ、New Relic は弊社顧客がシンプル、柔軟かつ透明性のある方法で **New Relic APM** および **New Relic Infrastructure** を購入できるよう設計された **クラウド課金モデル** を提供しています。シンプルさを特長に、クラウド課金モデルではインスタンスのサイズ、インスタンスの数、そして実行総時間を基に **New Relic APM** および **Infrastructure** の支払いが計算されます。単一ホストで週に60分間スピンアップする場合、単一ホストに対し60分間分の費用が請求されます。

5. 最も重要な時に対応可能な監視能力

クラウド上では、増加するインフラ規模に対応して、監視ソリューションのキャパシティを拡充する時間的余裕はありません。従来型のインフラオペレーションマネージャーがサービスやロードバランサー、その他要素にシリアル番号のスプレッドシートを管理しているも、ダイナミックなクラウド環境では機能しません。

オンプレミス型コントローラーを設定・調整して、予測不可能な回数で発生する大幅に増加した負荷に対応することはわずらわしくコストも膨大です。ダイナミッククラウドインフラストラクチャにおいては、特定の単一インスタンスの稼働状況のみを注視するのではなく、何千というホストを管理し、必要に応じて個別に該当ホストの状況を迅速に確認することができる環境を整えることが重要と言えるでしょう。

クラウド監視プラットフォームは停電や高い負荷がかかっている時期などの外部的な状況を問わず、常に利用可能な状態にあり、データを収集し続けて、何が起きているのかを常に把握できる状態であることが、非常に重要です。つまり、クラウド監視システムは他の重要なシステム同様に失敗が許されないということです。

New Relic であれば、突発的に訪れる、一年のうちで最も重要な日に対応するために、監視ソリューションのスケールを気にする必要はありません。New Relic はスタックの拡張や変更に関係なく、全体でリアルタイムにダイナミックな環境の変更を可視化可能な設計になっています。例えば、スプレッドシートを使用してウェブサイトの製品ページがSLAにどの程度対応しているか週報用に手作業で更新する代わりに、一度ダッシュボードを作成してしまえば必要なときにいつでもリアルタイムの情報を得られるようになります。

6. マルチテナント型 SaaS アーキテクチャ

本物のマルチテナント型 Software-as-a-Service (SaaS) アーキテクチャが誇る利点は、監視規模の拡張のために新たにサーバーを購入する必要がないというだけではありません。New Relic のような本物の SaaS ソリューションは、アプリを監視するためだけに専用のインフラストラクチャを維持、更新、管理する不安や手間、費用を抑えられるように設計されています。シングルテナント型ソリューションのようにパッチやアップグレードのインストールを強制する代わりに、ユーザーは常に**弊社の最新機能・イノベーションにアクセスすることができるのです。**



さらに、NRDB はリアルタイム データストアとして、生データをスキームやインデックスなしに検索可能です。ElasticSearch クラスタが継承されたこのようなシステムがスケールならびに効果的な実行には長年の経験が必須であり、簡単に実現可能なものではありません。

New Relic は本物の SaaS であることから、生データのクエリに関係ないデータベースやサービスのスケールアウトを心配する必要がありません。New Relic にデータを送信していただいたら、その後の処理は全て New Relic 側で対応いたします。

New Relic は本物の SaaS であることから、生データのクエリに関係ないデータベースやサービスのスケールアウトを心配する必要がありません。New Relic にデータを送信していただいたら、弊社側で対応いたします。

7. 将来に備えたソリューション

オンプレミス型の監視ソリューションは高度なサーバー側の変更やデータの予測分析にはあまり適していません。厳密に言えば、マルチテナントではないからです。マルチテナント型の環境では、データをサイロに取り出す代わりに何千という顧客の比較・ベンチマークについて様々な観点を生み出すことができます。ただし、シングルテナント型オンプレミスまたは管理ホスト環境でとなると、これは実用的ではありません。同様に、オンプレミスのソリューションプロバイダはデータについて通常何もすることができません。そのため、リアルタイムでアルゴリズムや予測分析を調整および改善することは至難の業です。

100% SaaS ベンダーである New Relic は、そのユニークな特性を活かして顧客から回収するデータを有効に活用しています。例えば **Project Seymour** における弊社 **プラットフォーム デモ** では、AI を活用して積極的にウェブサイトやアプリ上のエラーをより積極的に検出する様子をお見せしています。このようなクラウド上で AI やデータ分析を駆使し、データ量を最大限に活用して NRDB で容量を演算することが可能なプラットフォームは、New Relic の大きな差別化要因になっています。

結論

結局のところ、開発チームに監視ソリューションの管理運用ではなく、より良いソフトウェア開発にフォーカスして欲しいのであれば、オンプレミス型のインフラストラクチャとデータコレクタの更新について心配する必要のないソリューションが必要です。それらを気にせずとも、リアルタイムで有意義な情報を提供してくれる、エンドツーエンドの監視ソリューションが必要なのです。

New Relic Platform は、クラウドで展開するダイナミック インフラストラクチャとアプリケーションのオートスケールに対応するために構築されています。ネイティブクラウドインテグレーション、(集計メトリクスに加えて) 生データをキャプチャする能力、7言語に対応したエージェント、クラウド志向の課金モデルを備えた本物のマルチテナント SaaS ソリューションとして、New Relic はデジタルビジネスに対応したフルパッケージでのクラウド パフォーマンス監視プラットフォームを提供いたします。