

STATE OF CLOUD NATIVE DEVELOPMENT IN JAPAN 2026

日本におけるクラウドネイティブ開発の現状 2026

日本についての洞察
for KubeCon + CloudNativeCon Japan



2026年7月

このレポートのデータを共有できますか？

1. 使用許諾

このレポートは、[Creative Commons Attribution-NoDerivatives Licence 4.0 \(International\)](#) の下でライセンスされています。簡単にいえば、本ライセンスの条件に従って、次のことが自由に行えます。

共有 — 営利・非営利を問わず、本レポートを複製したり、本レポートの一部を1つまたは複数の文書や出版物に組み込んだりすることができます。

ただし、以下の条件に従うものとします：

帰属 — 本レポートのスポンサーである SlashData™ および Cloud Native Computing Foundation (CNCF) のクレジットを適切に表示し、変更が加えられた場合はその旨を表示する必要があります。その場合、合理的方法であればどのような方法でも構いませんが、SlashData™ があなたやあなたの使用を支持していることを示唆するような方法で行うことはできません。

改変禁止 — レポートの内容を編集したり、作り替えたりすることはできません。変更されたコンテンツを配布することも禁止されています。

2. 責任の制限

SlashData™ は、本出版物に含まれる記述が信頼できると考えられる情報に基づいていると考えていますが、それが正確または完全であることを表明するものではなく、そのようなものとして依拠すべきではありません。表明された意見は、本出版物に掲載された日付現在の意見であり、ここに含まれる意見を含む情報は、予告なしに変更されることがあります。どのような目的であれ、第三者による本書の使用は、当該第三者が本書の内容を確認する上で十分な注意を払うことを免責するものではなく、また免責されるものでもありません。SlashData™ は、商品性または特定目的への適合性を含むがこれに限定されない、すべての黙示的保証を放棄します。

SlashData™、その関連会社、および代理人は、この出版物に基づいて行われた、または行われなかった決定、または行われた、または行われなかったアクションの結果として、第三者が被った直接的、付随的、特別、または結果的な損害または逸失利益について、もしあれば一切責任を負いません。

[The analyst of the developer economy | formerly known as VisionMobile](#)
SlashData © Copyright 2026 | Some rights reserved

この日本語文書は、英語版を機械翻訳し、State of Cloud Native Development in Japan 2026 の参考訳として、The Linux Foundation Japan が便宜上提供するものです。
作成協力：佐藤駿



主要な知見	4	5	はじめに
クラウドネイティブ開発の トレンド	9	29	クラウドネイティブ成熟度 向上への道
付録	45	48	調査方法



主要な知見

- 2026年第1四半期時点で、日本には約95万人のクラウドネイティブ開発者がおり、世界全体では1,990万人となっています。→
- バックエンド開発者の71%が、クラウドネイティブコンピューティングに関連する技術や手法を少なくとも1つ利用しています。→
- 世界的に見て、ハイブリッドクラウドの利用率は過去最高を記録し、開発者の34%がハイブリッドクラウドへのデプロイを行っています。これは、データ主権の要件や規制遵守が背景にあると考えられます。→
- 日本では、オンプレミス型サーバーが依然として主要な導入インフラとして主流であり、日本の開発者の47%がこれを利用しています。→
- 他の地域に比べてクラウドの利用率は低いものの、クラウドネイティブ化の割合は世界平均と同水準です(日本:41%、世界平均:39%)。→
- バックエンド開発者の88%が、何らかの形のDevOps標準化に取り組んでいます。→
- 開発者がクラウドネイティブの成熟度を高めていくにつれ、機能フラグは、主流のインフラストラクチャ手法とより高度な手法との間をつなぐ重要なゲートウェイ技術としての役割を果たします。→
- 先進的な実践手法(カオスエンジニアリング、サービスメッシュ、不変インフラストラクチャの実践、マルチクラスタ管理)は、偶然に期待される水準よりも高い相互利用率を示す、緊密な「エリート・クラスター」を形成しています。→
- AI開発者は異なる成熟度の道筋をたどっており、不変インフラの実践は、最終目標というよりは、より高度な実践への入り口となっています。→

はじめに

01

クラウド導入はソフトウェア開発に欠かせない要素となっており、現在では開発者の大多数が何らかの形でクラウドインフラに依存しています。クラウド導入はほぼ普遍的なものとなりましたが、クラウドネイティブ技術やアーキテクチャ手法をいかに効果的に活用しているかについては、組織によって大きな差があります。クラウドインフラの普及が進んでいるにもかかわらず、多くの開発者や組織は、クラウドの潜在能力を最大限に引き出すクラウドネイティブ技術やアーキテクチャパターンを、まだ十分に活用できていません。

本レポートは、Cloud Native Computing Foundation (CNCF) との提携により作成されたもので、2026年第2四半期のState of Cloud Native Development (クラウドネイティブ開発の現状) に関する最新の知見を紹介しています。本レポートでは、2025年12月から2026年1月にかけて実施されたSlashDataのDeveloper Nation調査第31回版のデータに基づき、クラウドネイティブエコシステムを形作る主要なトレンドを分析しています。同調査では、世界95カ国から12,500人以上の回答者を対象に実施されました。

「クラウドネイティブ」という用語は開発者によって解釈が異なるため、本レポートでは、コンテナ、コンテナオーケストレーションおよび管理ツール、サービスメッシュ、Kubernetes、クラウドファンクションまたはサーバーレスコンピューティング、イベント駆動型アーキテクチャ、オブザーバビリティツール、不変インフラストラクチャの実践、カオスエンジニアリング、マルチクラスター管理、ストリーミングおよびメッセージングサービスといった、2つ以上の特定の技術¹の利用を基準として、開発者を「クラウドネイティブ」と定義しています。

本レポートは、「拡大された」クラウドネイティブのユーザー層（従来のバックエンドサービス分野の外側に存在するコミュニティを含む）を対象としたシリーズの第2弾です。そのため、本レポートでは引き続き、幅広い開発分野におけるクラウドネイティブ開発者の割合に関する推計値、およびこれがクラウドネイティブコミュニティの将来や、こうした新たな開発者に向けた技術開発にどのような影響を与える可能性があるかについて考察します。さらに、こうした多様な開発領域における各種クラウド技術の人気度についても検証します。拡大されたクラウドネイティブコミュニティの分析を通じて、クラウドネイティブ技術が現在のニーズに答えようとする中で理解すべき重要なグループである、人工知能（AI）コミュニティにも特に注目しています。

¹ 調査では、バックエンド開発に携わる回答者には、より多くのクラウドネイティブ技術が提示されています。そのため、バックエンド開発に携わる回答者は少なくとも 3つのクラウドネイティブ技術を選択する必要がありますが、それ以外の回答者は少なくとも 2つを選択する必要があります。

本レポートの新たなセクションでは、クラウドネイティブ分野におけるテクノロジーの導入経路と成熟度の推移について検証しています。関連性分析を用いて、どのクラウドネイティブ技術が一般的に組み合わせて採用されているかを特定し、基礎的なツールから主流の実践手法、さらには高度な信頼性およびスケーラビリティ管理機能に至るまで、明確な進展段階を明らかにしています。この分析により、テクノロジー開発者やプラットフォームチームは、自然な導入順序やアーキテクチャ上の親和性に関する洞察を得ることができます。



地域スポットライト：本レポートでは、横浜で開催される「KubeCon + CloudNativeCon Japan 2026」と連携し、日本におけるクラウドネイティブの導入動向やトレンドについて取り上げています。

クラウドネイティブ開発の トレンド

Q2

クラウドネイティブ開発者コミュニティは絶対数においても拡大を続けており、2025年第3四半期の1,560万人から2026年第1四半期にはおよそ約1,990万人に増加しています。これは全世界の開発者総数の約39%を占めており、クラウドネイティブの実践が、従来のバックエンドインフラストラクチャという枠を超えて、あらゆる分野における主流のソフトウェア開発へと広がっていることを反映しています。



1,990万人

2026年第1四半期時点の
クラウドネイティブ開発者数

2. クラウドネイティブ開発のトレンド

特にバックエンド開発者（従来からクラウドネイティブコミュニティの中核を成してきた層）のうち、2026年第1四半期時点で「クラウドネイティブ」に分類される割合は52%であり、2025年第3四半期の58%からは減少したものの、2025年第1四半期の49%よりは依然として高い水準にあります。四半期ごとの減少幅は顕著ですが、これは通常の変動を反映している可能性も、2025年第3四半期の58%という数値が過大評価されていたことを示している可能性もあります。2026年第3四半期の測定結果により、通常のベースラインが52%、58%、あるいはその中間値のいずれに近いのかが明らかになるでしょう。前年同期比の伸びは引き続きプラスとなっています。



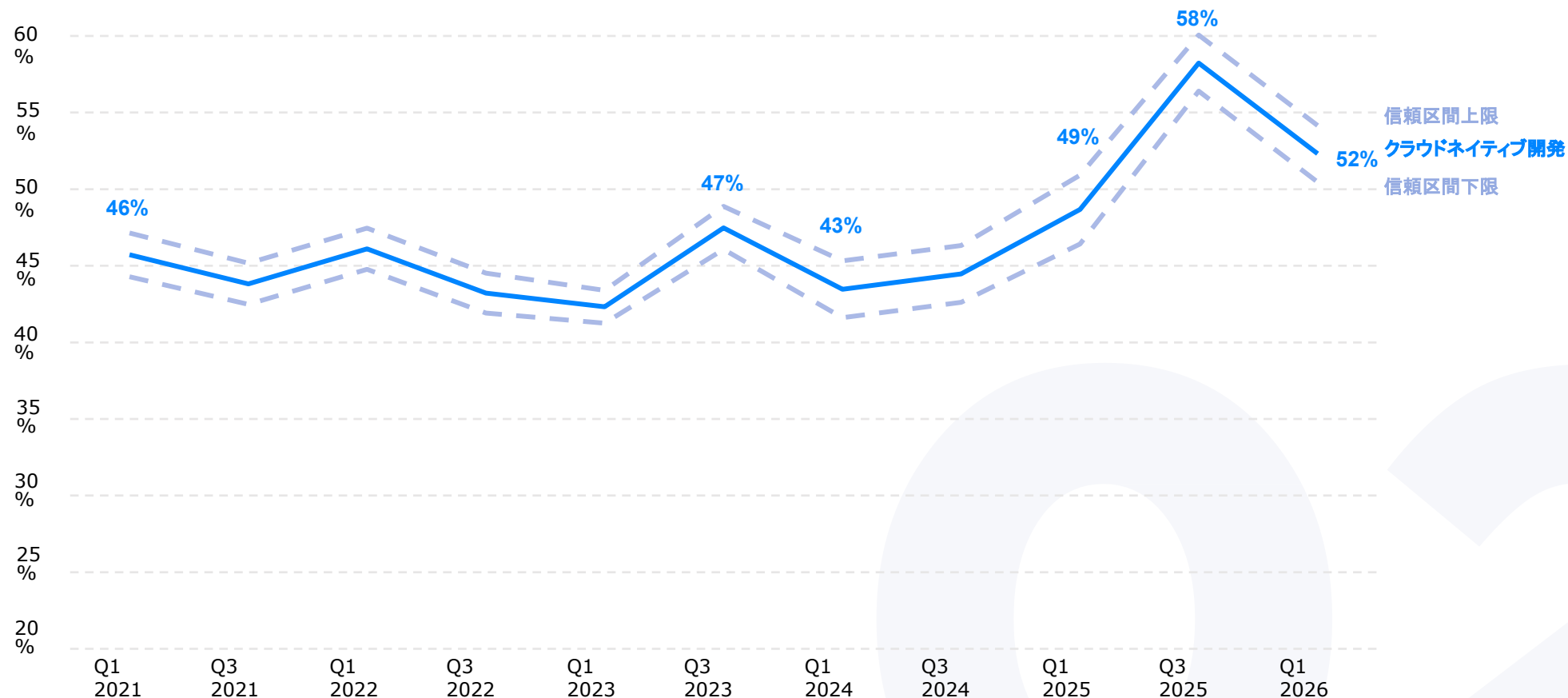
2026年第1四半期時点で、日本には約95万人のクラウドネイティブ開発者がいます。



日本では、開発者の41%がクラウドネイティブです（2026年第1四半期）。これは、世界平均の39%をわずかに上回っています。2025年第3四半期の47%からは減少しています。ただし、本調査における日本の開発者のサンプル数が少ないことを考慮すると、これらの数値は互いに95%信頼区間内に収まっています。さらに重要な点として、これは2024年第1四半期の24%から上昇しており、クラウドネイティブ技術に対する認知度と導入が確実に高まっていることを示唆しています。

世界におけるクラウドネイティブ・バックエンド開発者の割合

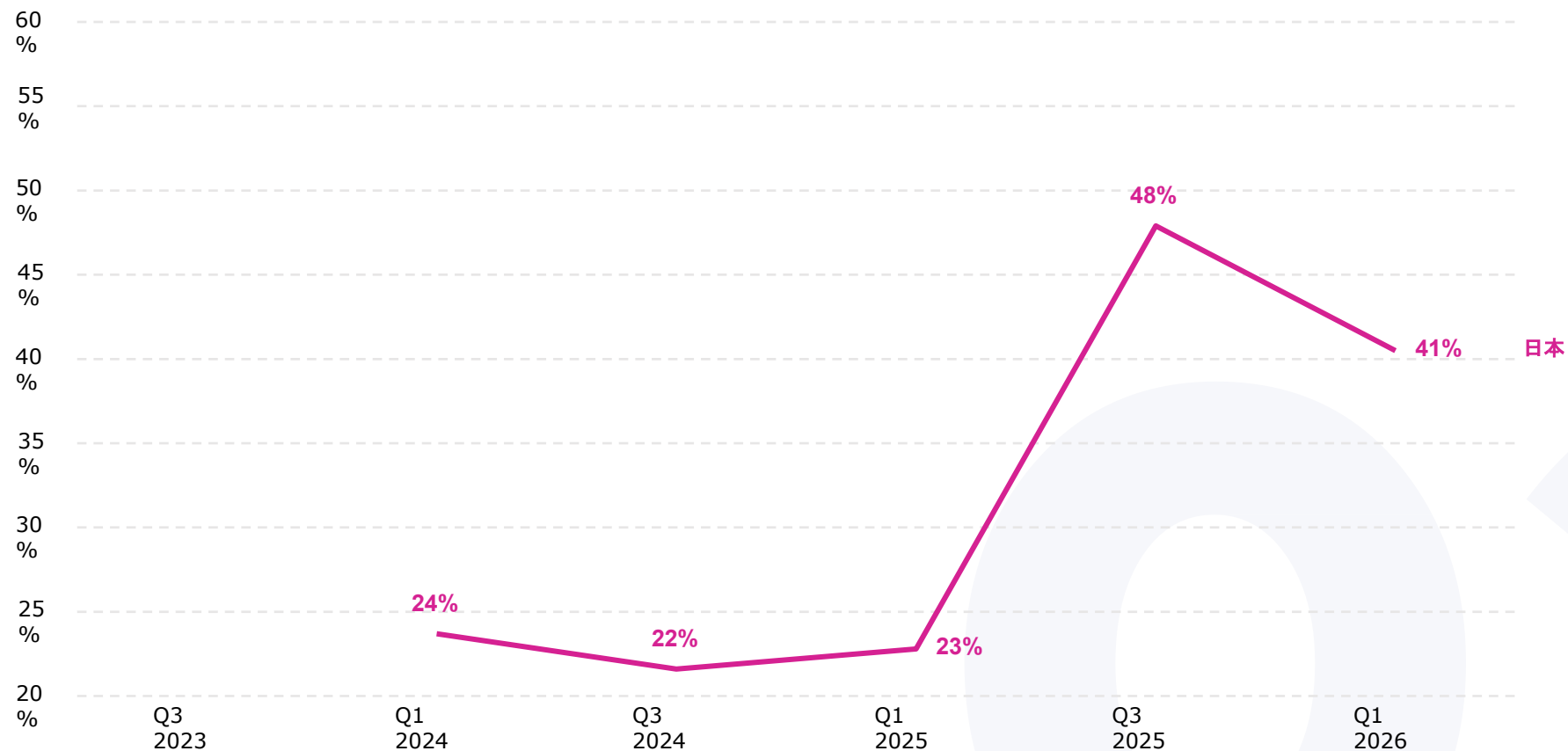
バックエンド開発者の割合 (%) Q1 2021 (n=4,668) | Q3 2021 (n=5,260) | Q1 2022 (n=5,261) | Q3 2022 (n=5,619) | Q1 2023 (n=8,182) | Q3 2023 (n=4,847) | Q1 2024 (n=2,743) | Q3 2024 (n=2,754) | Q1 2025 (n=1,903) | Q3 2025 (n=2,791) | Q1 2026 (n=2,793)



質問文: 過去12か月間に、バックエンドサービスの開発において、以下の技術のうちどれを使用しましたか？

日本におけるクラウドネイティブ開発者の割合

日本における割合 (%) Q1 2024 (n=93) | Q3 2024 (n=118) | Q1 2025 (n=71) | Q3 2025 (n=74) | Q1 2026 (n=92)



質問文: 過去12か月間に、バックエンドサービスの開発において、以下の技術のうちどれを使用しましたか？

クラウドネイティブを採用する開発者の増加は、特定の分野に集中するのではなく、幅広い分野で見られました。

例えば、プロフェッショナルなゲーム開発者のうち、クラウドネイティブを採用する割合は、2025年第3四半期の30%から2026年第1四半期には39%に増加しました。一方、産業用IoT開発者では、同期間に38%から42%へと上昇しました。こうした増加は、クラウドネイティブ技術が、専門的なインフラ業務だけでなく、多様な開発環境において標準的なツールとなりつつあることを反映しています。

AI開発者コミュニティも同様の成長を見せましたが、その伸びはより緩やかであり、2026年第1四半期のクラウドネイティブAI開発者数は、2025年第3四半期の710万人から約730万人に増加しました。クラウドネイティブ技術によって、スケーラブルなAI運用、分散トレーニング、本番環境へのモデル展開がますます可能になっていることから、この分野は引き続き戦略的に重要な位置を占めています。

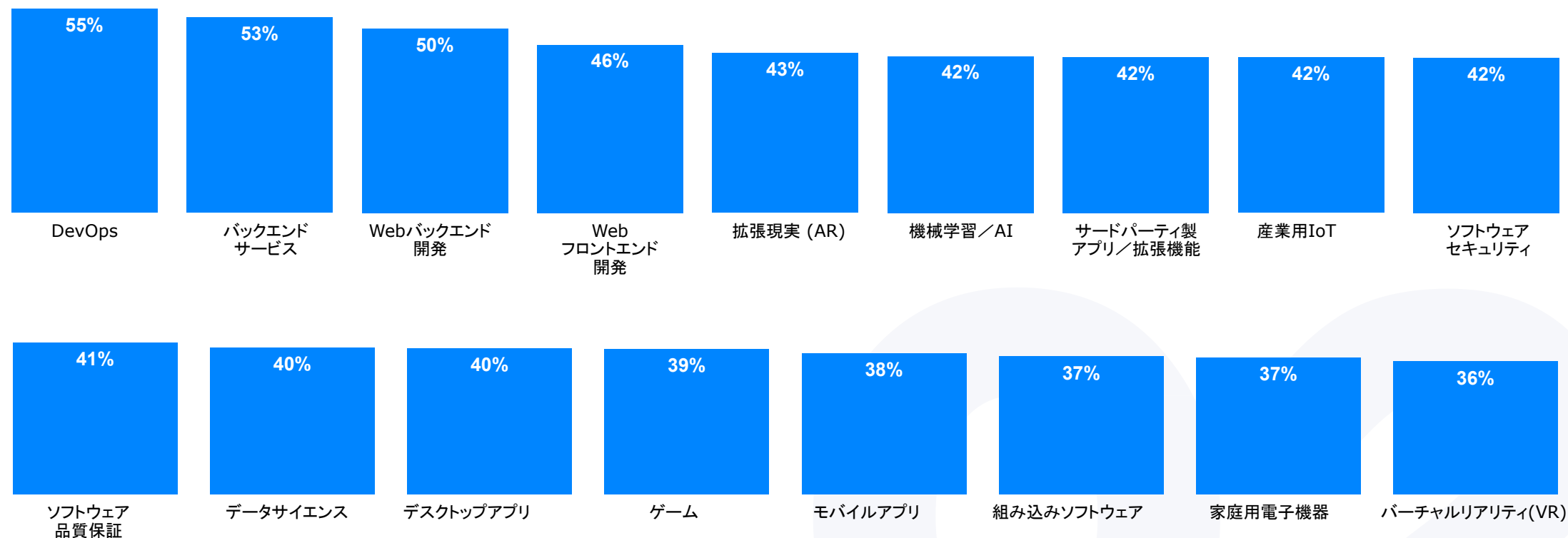


730万人

AI開発者はクラウドネイティブであり、
そのうち約10万人が日本に拠点を置いている

各開発分野における、クラウドネイティブの プロフェッショナル開発者の割合

各開発分野におけるプロの開発者の割合 (%) (Q1 2026: 全世界 n=9,082)



質問文: 以下の開発やプロジェクトに、どのような形で関わっていますか？ 過去12か月間に、バックエンドやクラウドサービスの開発において、以下の技術のうちどれを使用しましたか？

クラウド導入のトレンド

クラウドネイティブ開発者の動向に加え、クラウド導入の選択肢が時間とともにどのように変化してきたかについても検証しています。開発者コミュニティ全体を見ると、過去5年間でハイブリッドクラウドの採用が大幅に増加していることがわかります。2021年第1四半期には開発者の22%がハイブリッドクラウドにデプロイしていましたが、2026年第1四半期には34%に増加しました。ハイブリッドクラウドは、中堅企業や、金融・医療など規制の厳しい業界において非常に人気のある選択肢として台頭しています。これは、多くの人々にとってますます魅力的あるいは必要不可欠となっている、セキュリティやコンプライアンス上の追加メリットを提供するためです。欧州のGDPRや米国のクラウド法など、データ主権に関する要件やプライバシー規制の進化により、クラウドのスケーラビリティとオンプレミスでのデータ保管のバランスを取る必要がある組織にとって、ハイブリッド展開は魅力的な選択肢となっています。



の開発者が2026年第1四半期にハイブリッドクラウドへのデプロイを行っており、これは過去最高の割合となっている

日本における過去2年間だけを見ると、状況は異なってきます。オンプレミスサーバーへのコードやサービスの展開が依然として主流であり、調査対象となった開発者のほぼ半数(47%)がこの選択肢を選んでいきます。この地域では、ハイブリッドクラウドは依然として少数派の選択肢にとどまっており、このインフラストラクチャ手法を利用していると回答した開発者はわずか16%にとどまっています。このことから、2つの明確かつ興味深い点が浮かび上がります: すなわち、日本は世界の他の地域とは構造的に異なるインフラストラクチャ戦略をとっていること、そして日本は異なるメカニズムを通じて、他地域と同等のレベルのクラウドネイティブ化を実現しているということです。

日本では、システムインテグレーター(SI)が、組織や開発者のニーズに応えるため、オンプレミスサーバー向けにカスタマイズされた専用システムを構築してきた長い歴史があります。その結果、こうしたシステムの多くは、クラウドサービスへの移行が困難なほど複雑になっています。さらに、その高性能さゆえに、クラウドサービスのメリットは、オンプレミスサーバーが提供するデータ主権やリスク管理の制御機能と競合せざるを得ない状況にあります。

重要な背景情報として、この結果はインフラストラクチャの選択の多様性を測定したものであり、各インフラストラクチャの種類ごとにどれだけの演算処理が行われているかを必ずしも示しているわけではない点に留意してください。例えば、ある組織では、組織内のほとんどのチームがオンプレミスサーバーを使用している一方で、AIチームはハイブリッドクラウド構成を利用して、クラウドプラットフォーム上でAI推論ワークロードを実行している場合があります。現在、日本のデータセンターおよびAIインフラ分野では著しい成長と投資が進んでおり、組織全体のインフラストラクチャ全体ではオンプレミスサーバーが依然として主流である一方で、特定の高計算負荷ワークロードを対象としたこのようなハイブリッド利用が、今後ますます一般的になっていく可能性があります。

インフラ戦略におけるこの構造的な違いは、日本が異なる経路を通じてクラウドネイティブの同等水準を達成した理由として、最も説得力のある説明でもあります。他の地域では、クラウドネイティブ性のレベルが高いほど、より広範なクラウド利用率の高さと並行して現れる傾向があります。しかし、日本では、このパターンが見られないにもかかわらず、開発者層の間で同等のレベルのクラウドネイティブ化が達成されています。その一因として考えられるのは、SI企業が、自社が管理するオンプレミス型サービスの中にクラウドネイティブ技術やアプローチをますます統合していることです。この傾向は、日本のクラウドネイティブエコシステムにおいて、富士通や日立といったシステムインテグレーターが重要な役割を果たしていることとも一致しています。つまり、日本においては、他の地域とは多少異なる文脈でクラウドネイティブ性を考慮する必要があります。その対象となるユーザー層が、クラウドネイティブの実践に取り組む従来のクラウドユーザーと、提供サービスを最大化するためにクラウドネイティブのアプローチを統合したオンプレミスサーバーをSIを通じて利用しているユーザーとに二分されているからです。



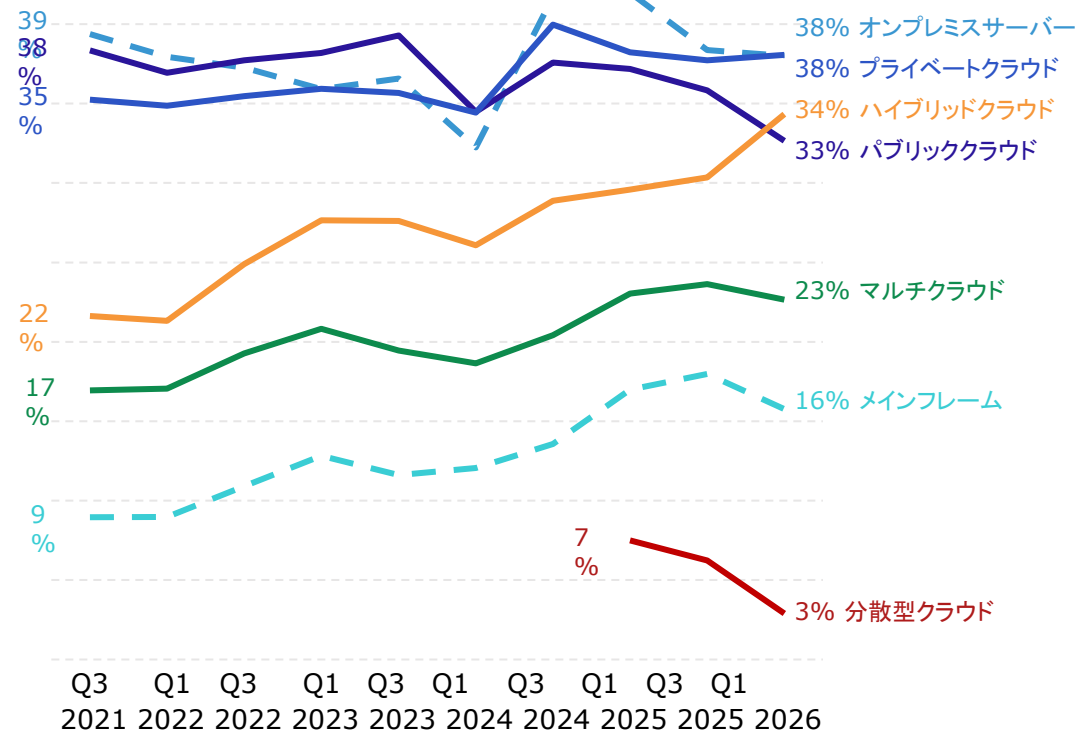
日本のSI主導型インフラモデルでは、クラウドネイティブツールの選定は通常、開発者レベルではなく組織レベルで行われます。これには調達面以外の影響もあります。つまり、自組織がすでに依存している技術を支えるオープンソースプロジェクトに、日本の開発者が直接関与する機会が少なくなってしまう可能性があるのです。これは、日本にとってだけでなく、オープンソースコミュニティにとっても重要な問題です。さまざまな地域、文化、業界、役割を持つ開発者が、自分たちや同僚のために役立つソリューションを構築することで、オープンソースは強化されていくのです。さらに、クラウドネイティブプロジェクトにより直接的に関与する日本の開発者コミュニティは、日本特有の課題、優先事項、ユースケースを貢献することができ、それらがなければ、より広範なコミュニティには決して知られ得なかったであろう情報を提供できるのです。

2. クラウドネイティブ開発のトレンド

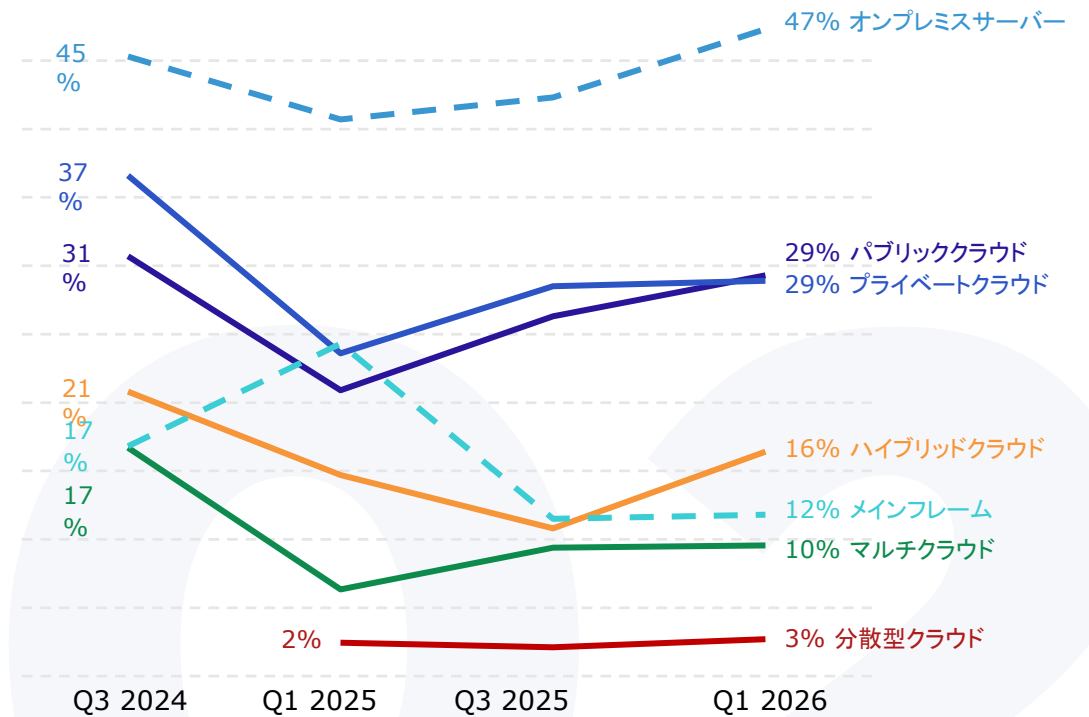
クラウド導入のトレンド

開発者の比率(%) | 日本における開発者の比率(%) (全世界: 21Q1 n=11,177 | 21Q3 n=12,132 | 22Q1 n=13,344 | 22Q3 n=17,820 | 23Q1 n=20,879 | 23Q3 n=10,521 | 24Q1 n=10,521 | 24Q3 n=9,886 | 25Q1 n=10,731 | 25Q3 n=12,021 | 26Q1 n=11,937 日本: 24Q3 n=224 | 25Q1 n=192 | 25Q3 n=232 | 26Q1 n=224)

全開発者の間でみられる
クラウド導入の傾向



日本の開発者の間でみられる
クラウド導入の傾向



質問文: 過去12か月間に、バックエンドサービスの開発において、以下の技術のうちどれを使用しましたか？

クラウド技術

クラウドネイティブ開発は、開発者が使用する技術の範囲にとどまりません。しかし、クラウドネイティブ開発に関連する技術の利用状況を追跡することは、この分野がどのように進化しているかを理解する上で重要な手がかりとなります。このセクションでは、バックエンド開発者の間におけるこれらの技術の採用状況について見ていきます。

最も顕著な変化は、2020年第3四半期以降一貫して60%を上回っていたコンテナの利用率が、40%へと急激に低下したことです。この減少は、コンテナの利用が劇的かつ予期せぬ形で落ち込んだように見えるかもしれませんが、開発業界での経験からすれば、そうではないことは明らかです。[本レポートの前号](#)でも述べたように、この減少は、バックエンド開発者コミュニティの定義方法における方法論的な変更を反映したものです。現在の定義には、従来のバックエンドインフラ開発者に加え、通常、基盤となるインフラを抽象化した高レベルのサービスを扱うWebバックエンド開発者が含まれています。この構成の変化により、各サブグループ内の採用率が比較的安定している場合でも、報告されるコンテナ利用率は必然的に減少することになります。¹

¹ 過去のバックエンド開発者を対象とした調査によると、コンテナの導入率は 58%となっています。

その他の技術については、そのほとんどでわずかな減少が見られますが、サービスメッシュは例外です。サービスメッシュは過去6か月間で8%から11%へと増加しましたが、対象範囲の変更が行われる前の過去最高値にはまだ達していません。他の技術に影響を与えている構成の変化にもかかわらず、サービスメッシュの導入がこのような増加していることは、導入が大幅に拡大していることを示唆しており、その背景には、サービスメッシュのメリットが導入コストを上回る規模や複雑さに達した組織が増えていることが要因として考えられます。

バックエンドの定義範囲が拡大したことに伴う構成の変化の後も、自己申告による技術導入率が低下し続けていることは、行動的な要因が影響している可能性を示唆しています。プラットフォームエンジニアリングの実践が拡大するにつれ、開発者はコンテナ、Kubernetes、サービスメッシュといった技術を直接設定するのではなく、抽象化レイヤーを介してインフラストラクチャとやり取りすることが増えてきています。これにより、開発者はアプリケーションのロジックやビジネス上の課題に集中でき、プラットフォームチームが基盤となるインフラストラクチャを管理できるようになります。



正式なプラットフォームや DevOpsの実践がないバックエンド開発者の割合は、20%から12%へと急減しており、インフラの標準化と抽象化が進んでいることを示しています

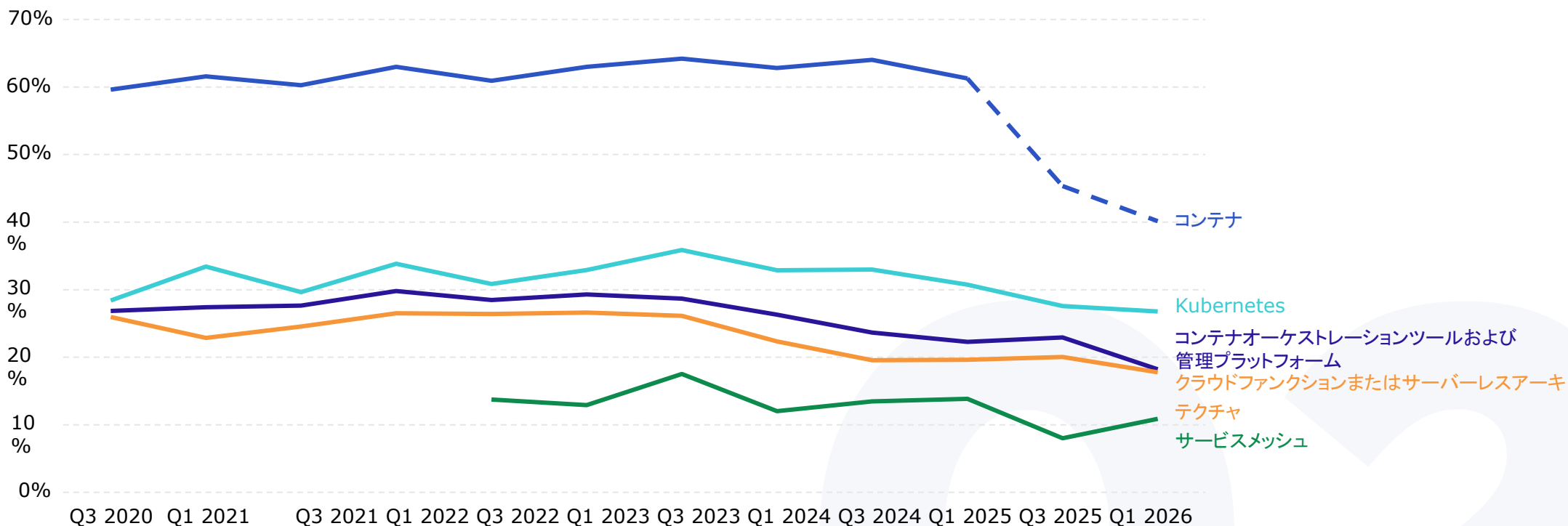
個別のプラットフォームエンジニアリング手法は依然として少数派ですが、その組み合わせは臨界点に達しています。バックエンド開発者の88%が、現在少なくとも1つの形態のインフラストラクチャ標準化を採用しており、これは6か月前の80%から増加しています。具体的には、デベロッパーエクスペリエンス(DX)チームを擁する開発者の割合が38%から42%に、統一されたDevOpsシステムの導入率が36%から41%にそれぞれ増加しました。インフラストラクチャの標準化を一切行わずに開発を行っている開発者の割合が8ポイント減少したのと同時に、コンテナの利用率が6ポイント低下し、その他のクラウドネイティブ技術においても同様の小幅な減少が見られました。

以前、[「DevOpsの2側面」](#)に関する調査で論じたように、成功したプラットフォームエンジニアリングでは、インフラストラクチャ運用担当者(技術を管理する者)とアプリケーション開発者(簡素化されたインターフェースを通じてそれらを利用する者)との間に役割分担が確立されます。これにより、測定上の疑問が生じます。開発者がDockerfileを作成したりレジストリを設定したりすることなく、IDPを通じてコンテナ化されたアプリケーションをデプロイする場合、彼らは自らを「コンテナを使用している」と報告すべきなのでしょうか？

標準化された環境の普及と、報告される技術利用状況の減少が同時に見られることは、たとえ基盤となるインフラストラクチャに変化がなくても、インフラストラクチャの抽象化が、開発者が技術の利用状況をどのように認識し、報告するかに影響を与えていることを示唆しています。

クラウドネイティブ開発に関連する技術の導入動向

バックエンド開発者の比率 (%) Q1 2021 (n=4,668) | Q3 2021 (n=5,260) | Q1 2022 (n=5,261) | Q3 2022 (n=5,619) | Q1 2023 (n=8,182) | Q3 2023 (n=4,847) | Q1 2024 (n=2,743) | Q3 2024 (n=2,754) | Q1 2025 (n=1,903) | Q3 2025 (n=2,666) | Q1 2026 (n=2,792)



注意: バックエンドサービス開発の性質が変化していること、および当社の調査におけるその方法論的な捉え方の変化により、コンテナの利用数が大幅に減少したのは、開発者の構成に変化が生じたことを反映しています。過去のグループにおける傾向の推定値については、『2025年第3四半期のクラウドネイティブの現状』をご参照ください。

質問文: 過去12か月間に、バックエンドまたはクラウドサービスの開発において、以下の技術のうちどれを利用しましたか？

サーバー技術とアプローチ

このセクションの最後の部分では、バックエンド開発に焦点を当て、そのインフラストラクチャに使用されている技術やアプローチについて、より詳細に掘り下げていきます。このデータにより、それらの相対的な人気度を初めてより明確に把握することが可能になります。さらに、クラウドネイティブに分類される基準を満たしていない開発者であっても、クラウドネイティブ技術やアプローチの採用がどれほど広まっているかを確認できます。こうした開発者の多くは、依然として個別のクラウドネイティブ技術を利用しています。クラウドネイティブの分類基準(3つ以上の技術)を満たしているのは52%に過ぎませんが、バックエンド開発者の71%が少なくとも1つのクラウドネイティブ技術を利用しています。この19ポイントの差は、多くの開発者がクラウドネイティブの導入初期段階にあり、1つまたは2つの技術を使用しているものの、成熟したクラウドネイティブの実践に特徴的なマルチテクノロジースタックをまだ採用していないことを示しています。



39%がマイクロサービスを利用している一方で、カオスエンジニアリングやイミュータブルインフラストラクチャといった高度な手法を採用しているのはわずか6~7%にとどまっています

APIゲートウェイは最も普及している技術であり、バックエンド開発者の47%が利用しています。一方、マイクロサービスは主要なアプローチとして、バックエンド開発者の39%に採用されています。これは、APIゲートウェイとマイクロサービスの両方が、他の技術に不可欠であるか、あるいは現代のサービススタックの機能に不可欠であることが多いという点で、その基盤的な性質を反映しています。注目すべきは、APIゲートウェイとマイクロサービスは通常Kubernetesインフラ上で動作するアーキテクチャパターンであるにもかかわらず、Kubernetesを使用していると明示的に回答したのはわずか27%にとどまっている点です。これは、開発者がKubernetes以外の環境でマイクロサービスをデプロイしている可能性を示唆しているほか、前述したように、プラットフォームの抽象化によって、たとえデプロイメントの基盤にKubernetesが使用されていたとしても、開発者からはKubernetesが「見えない」状態になっている可能性も考えられます。

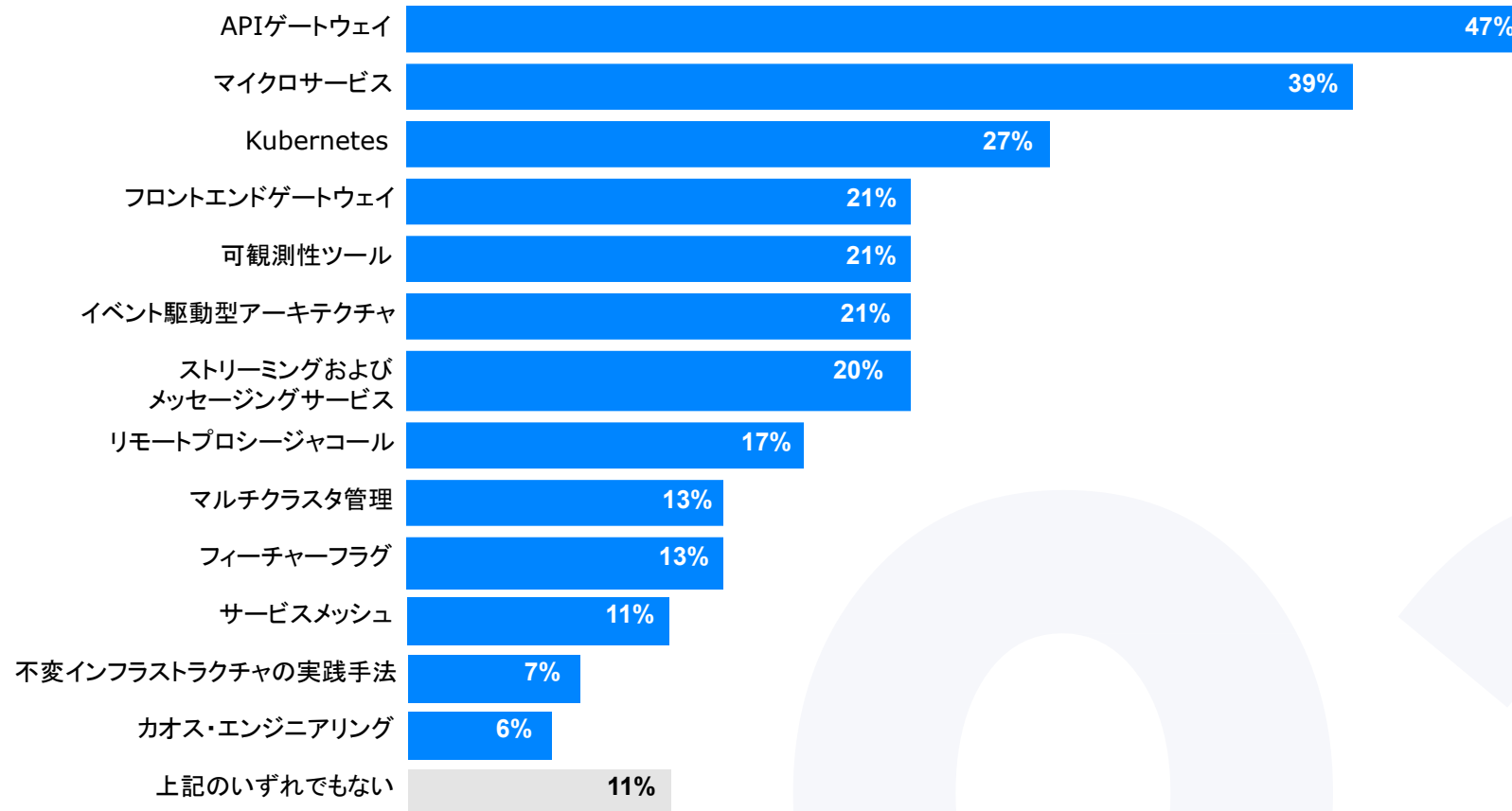
前述したように、IDPの台頭やDevOpsプロセスの集中化が、テクノロジーの利用状況に関する自己申告の減少の一因となっている可能性が高いです。この場合、Kubernetesに依存する技術を頻繁に利用しているにもかかわらず、自分自身をKubernetesのユーザーとは考えていない開発者がいる可能性があります。IDPの利用拡大やDevOpsプロセスの集中化は、組織のプラットフォームエンジニアリングが成熟度を高めていることを如実に示す印象的な動きではありますが、組織としては、開発者の生産性を支える基盤プラットフォームを管理、トラブルシューティング、最適化するためのインフラに関する専門知識がチーム内に確実に存在するようにすべきです。

Kubernetes(27%)に次いで、最も一般的な技術やアプローチとしては、フロントエンドゲートウェイ(21%)、オブザーバビリティツール(21%)、イベント駆動型アーキテクチャ(21%)、ストリーミングおよびメッセージングサービス(20%)が挙げられます。これらはAPIゲートウェイやマイクロサービスほど普及してはいませんが、より高度なアーキテクチャアプローチを構成しています。この統計は、バックエンド開発者のコミュニティのかなりの割合が、より成熟したインフラストラクチャのアプローチに取り組んでいることを示しています。

Kubernetes やコンテナはデフォルトで不変インフラストラクチャを提供していますが、開発者に対しては、不変インフラストラクチャの実践を軸とした開発哲学を反映した取り組みや行動を行っているかどうかを尋ねました。その結果、不変インフラストラクチャの実践を行っていると回答した割合(7%)は低いだけでなく、カオスエンジニアリング(6%)についても同様の傾向が見られました。これは、複数の要因が反映されていると考えられます。多くの組織では、これらのプラクティスが明確なROIをもたらすような規模や複雑さにまだ達していないこと、実装には技術的な能力を超えた組織的・文化的な成熟度が求められること、そして現在のツールは、一般的なチーム規模においてそのメリットに比べて導入の障壁となっている可能性があることなどが挙げられます。プラットフォームエンジニアリングが成熟し続け、ツールが改善されるにつれて、これらのプラクティスは主流の開発チームにとってもより利用しやすくなるでしょう。

バックエンド技術やインフラストラクチャのアプローチの採用

バックエンド開発者中の比率 (%) (Q1 2026 n=2,792)



質問文: 過去12か月間に、バックエンドサービスにおいて以下のインフラストラクチャのアプローチや技術のいずれかを利用しましたか？ もし利用した場合は、どれですか？

クラウドネイティブ
成熟度
向上への道

03

クラウドネイティブ技術は開発者コミュニティ全体で拡大を続けており、2026年初頭にはクラウドネイティブ開発者の数は2,000万人近くに達すると見込まれています。しかし、クラウドネイティブの普及状況を把握するには、「広がり」（クラウドネイティブの実践を採用している開発者の数）と「深さ」（その実装がいかに洗練されているか）という2つの側面を検討する必要があります。「広がり」は、クラウドネイティブ技術を使用している開発者の数を数えるだけで測定が比較的容易ですが、「深さ」はより複雑です。なぜなら、我々が注目しているのは、より高度な技術の使用状況だけでなく、洗練されたアーキテクチャパターンの兆候でもあるからです。

すべての開発者が高度なクラウドネイティブの実践を必要とするわけではありません。基礎的なツールで十分なプロジェクトに携わる開発者もいます。しかし、複雑で大規模なシステムを管理する開発者にとっては、クラウドネイティブの成熟度を高めるための道筋を理解しておくことが、戦略的な技術的判断を下す上で参考になります。

この問題を探求するため、本レポートでは、開発者がKubernetesやマイクロサービスといった主流のクラウドネイティブ手法から、不変インフラやカオスエンジニアリングといった高度な手法へと移行する過程において、技術導入の道筋を明らかにしようと試みます。そのために、技術間の相対的な併用状況を検証するとともに、さらに重要な点として、技術間の「リフト」を分析します。リフトとは、2つの技術が、偶然に予想される頻度よりも頻繁に併用されているかどうかを測定する指標です（「[調査方法](#)」参照）。つまり、リフトが1を超える場合は正の相関関係を示し、1未満の場合は負の相関関係を示します。

この分析により、2つの相互に補完し合う結果が得られます：

リフト行列：すべての技術ペア間の関連性の強さを示した包括的な表であり、どの組み合わせが一般的に併用されているかを明らかにします

開発者のジャーニーマップ：成熟度の道筋を視覚的に表現したもので、組織がクラウドネイティブの基礎的な実践から高度な運用パターンへと移行する過程で、技術が一般的にどのような順序で導入されるかを明らかにするものです。

こうした導入経路を理解することは、複数のステークホルダーにとって有益です。**技術開発者**は、どの技術が自社のツールへの入り口となるかを特定できます。**プラットフォームチーム**は、自然な進展経路に沿った導入戦略を策定できます。また、**組織**は、恣意的な技術導入を試みるのではなく、自社の成熟度を評価し、論理的な次のステップを特定することができます。

¹ 基礎となるネットワークグラフの可視化図は、[付録](#)に掲載されています。

開発者の歩み

一般的な開発者にとって、技術的成熟度を高める道筋として浮かび上がってくるのは、「機能フラグ」と「可観測性ツール」という2つの重要なプロセスと技術を経由するものです。

APIゲートウェイは最も一般的に使用されている技術ですが、多くの技術においてリフト値が約1.0を示しており、特定のアーキテクチャパターンに関連付けられているというよりは、成熟度のレベルを問わず広く利用されていることを示しています。注目すべき例外として、不変インフラストラクチャの実践（リフト値：0.79）やカオスエンジニアリング（リフト値：0.53）との間に負の相関が見られます。「エリート実践クラスター」に属する組織は、より洗練された代替的なトラフィック管理手法を好む傾向があり、これはエリート実践者が、より主流なクラウドネイティブ導入企業とはアーキテクチャの面で異なることを示唆しています。

その代わりに、開発者の間でまず目につくのは、Kubernetesとマイクロサービスの併用です。この組み合わせは、Kubernetesの本来の設計目的、すなわちコンテナ化されたマイクロサービスアーキテクチャのオーケストレーションを反映しています。マイクロサービスを導入する組織は、コンテナのオーケストレーション、サービスディスカバリ、ロードバランシングのために自然とKubernetesを採用するようになります。これらは、マイクロサービスの数が手動管理の限界を超えて増加するにつれて、不可欠となる機能だからです。

Kubernetes ユーザーの間では、オブザーバビリティツール(リフト: 1.41)とストリーミングおよびメッセージングサービス(リフト: 1.33)の両方の採用が拡大していることが確認されています。この傾向は運用上の必要性を反映しています。Kubernetesのデプロイメントが複雑化(複数のサービス、ネームスペース、クラスターの管理)するにつれ、分散システムを監視するために可観測性が不可欠となる一方で、ストリーミングおよびメッセージングサービスは、マイクロサービスアーキテクチャが疎結合と耐障害性を実現するために必要とする非同期通信インフラストラクチャを提供します。

ただし、ストリーミングおよびメッセージングサービスについては、イベント駆動型アーキテクチャにおいても採用率の上昇(上昇率: 1.45)が見られますが、これは予想通りの結果です。ストリーミングプラットフォームは、イベント駆動型パターンを大規模で実用的にするインフラ基盤を提供します。イベント駆動型アーキテクチャはアーキテクチャ的なアプローチを表す一方で、ストリーミングおよびメッセージングサービスは技術的な実装を提供します。

次に重要な関連性は、フィーチャーフラグgingとオブザーバビリティツール(1.53)、ストリーミングおよびメッセージングサービス(1.31)、イベント駆動型アーキテクチャ(1.45)との相乗効果です。さらに、イミュータブルインフラストラクチャの実践(1.33)、マルチクラスター管理(1.40)、カオスエンジニアリング(1.44)との組み合わせにおいても、利用率の向上が見られます。したがって、フィーチャーフラグは、クラウドネイティブサービスの成熟度を高めたいと考える開発者にとって鍵となる、極めて重要なゲートウェイ技術としての役割を果たします。フィーチャーフラグは、2つの重要な運用機能——コードの変更をユーザーに即座に公開することなくデプロイする能力(デプロイとリリースの分離)と、トラフィックの一部を用いて変更を段階的にテストする能力——を導入することで、主流のプラクティスと高度なプラクティスを橋渡しする役割を果たします。これらの機能により、組織はプログレッシブ・デリバリー・パターンや制御された実験という考え方を学ぶことができます。これらは、カオスエンジニアリング(制御された障害の注入)やイミュータブル・インフラストラクチャ(デプロイメントを使い捨ての実験として扱う)といった高度なプラクティスに直接つながる思考モデルです。フィーチャーフラグは、そうでなければ難しそうに思える高度な運用手法への、リスクの低い入り口としての役割を果たします。

高度な実践手法へのもう一つの道筋は、オブザーバビリティ・ツールを通るものであり、これはイミュータブル・インフラストラクチャとの関連性が示されています(リフト: 1.33)。このルートは、包括的なオブザーバビリティに投資し、その結果、イミュータブル・インフラストラクチャを自信を持って導入するために必要な可視性を確立している組織の実態を反映していると考えられます。オブザーバビリティは、「破棄して再デプロイする」という戦略を現実的なものにするセーフティネットの役割を果たします。

開発者が高度なプラクティスを採用すると、技術間の相関値が著しく高くなることが確認されます。具体的には、不変インフラとカオスエンジニアリング(2.22)、サービスメッシュとカオスエンジニアリング(1.76)、そして不変インフラとサービスメッシュ(1.75)です。これらの値は、基盤となる技術間の相関よりも大幅に高く、組織が段階的ではなく同時に複数の高度なプラクティスを採用する「エリート・クラスター」の存在を示唆しています。この極めて高い相関値(2.22—データセット中で最も強い値)は、技術的な依存関係(失敗した実験を迅速に置き換えられるため、不変インフラストラクチャと組み合わせることでカオスエンジニアリングの安全性が向上する)と、哲学的な整合性(両方のプラクティスがインフラストラクチャを使い捨てと見なし、失敗を学習の手段として受け入れる)の両方を反映していると考えられます。一方を採用している組織は、自然と他方にも傾倒していく傾向があるでしょう。

一般的な開発者については、このデータから3段階の成熟度モデルが明らかになっています：

基礎： Kubernetes + マイクロサービス(エントリーポイント)

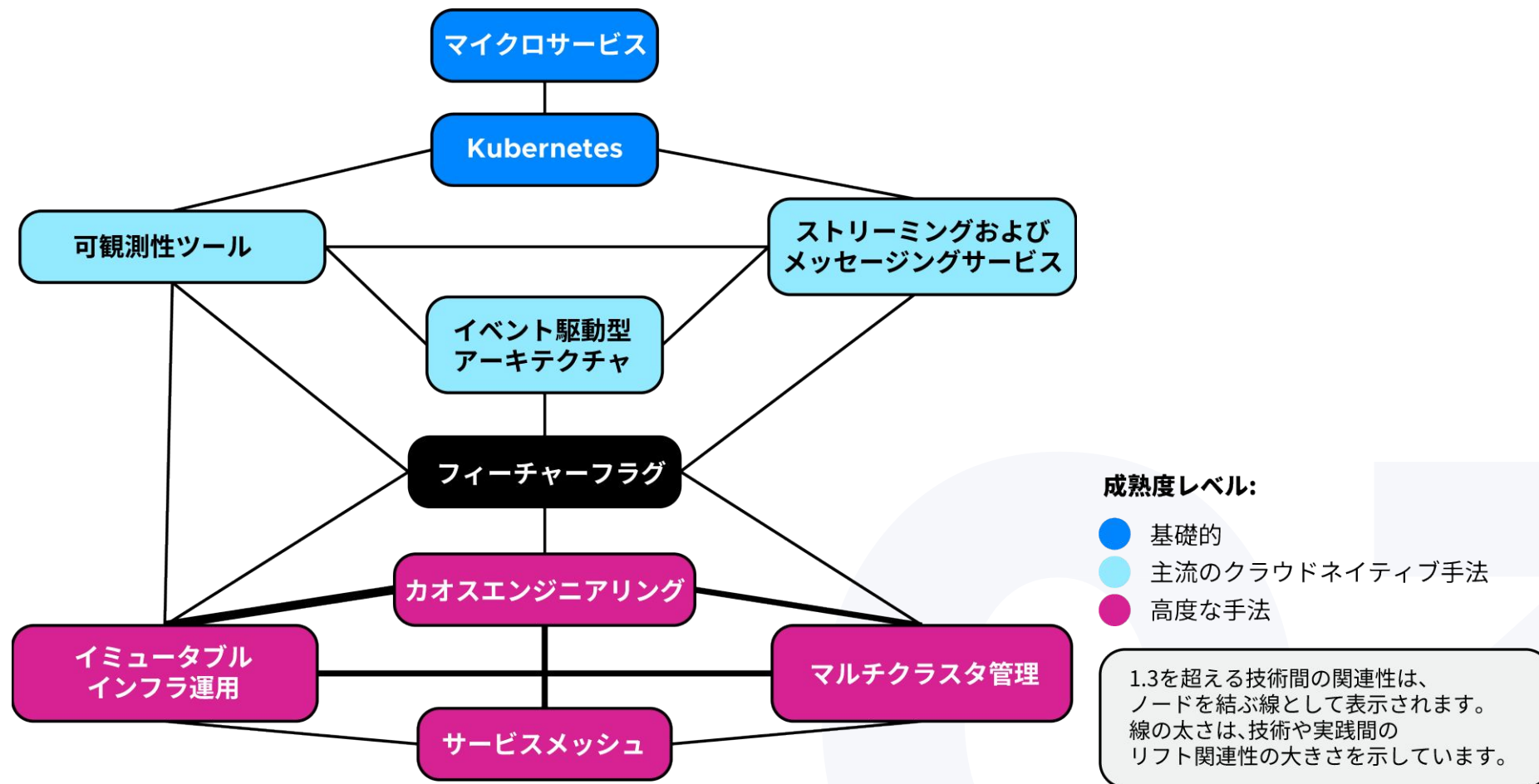
クラウドネイティブの主流： 可観測性ツール、ストリーミング／メッセージング、イベント駆動型アーキテクチャ(運用面の高度化)

高度な実践手法： 不変インフラ、カオスエンジニアリング、サービスメッシュ(最高レベルの運用成熟度)

機能フラグ設定は、ティア2とティア3をつなぐ重要な架け橋としての役割を果たしており、2つのアプローチが考えられます。組織は、実験の成熟度(機能フラグ設定)または運用可視性(オブザーバビリティツール)のいずれかを通じて、高度な実践手法に取り組むことができます。組織がティア3に到達すると、通常、高度な実践手法を段階的にではなく同時に複数導入する傾向にあり、これは、これらの実践手法が個別の機能というよりも、一貫した運用哲学を形成していることを示唆しています。

開発者のクラウドネイティブへの道のり

バックエンド技術間の関連性



質問文: 過去12か月間に、バックエンドサービスにおいて以下のインフラストラクチャのアプローチや技術のいずれかを利用しましたか？もし利用した場合は、どれですか？

バックエンド技術間の関連性を明確化

バックエンド技術間の関連性の明確化 (Q1 2026 n=2,792)

マイクロサービス	1.39											
サービスメッシュ	1.04	0.87										
APIゲートウェイ	1.08	1.13	0.77									
イベント駆動型アーキテクチャ	1.27	1.17	1.24	1.08								
可観測性ツール	1.41	1.11	1.28	0.98	1.27							
不変インフラストラクチャの実践手法	1.12	0.83	1.75	0.79	1.15	1.33						
カオス・エンジニアリング	0.66	0.68	1.76	0.53	0.77	0.96	2.22					
フィーチャーフラグ	1.19	1.01	1.27	0.97	1.45	1.53	1.33	1.44				
リモートプロシージャコール	0.90	0.87	0.96	1.00	0.95	0.89	0.98	0.88	1.16			
フロントエンドゲートウェイ	1.04	0.92	1.09	1.17	0.89	0.97	1.14	0.82	1.19	1.03		
マルチクラスタ管理	0.96	0.84	1.42	0.86	1.08	0.91	1.68	1.64	1.40	1.28	1.04	
ストリーミングおよびメッセージングサービス	1.33	1.15	1.26	0.97	1.45	1.40	1.11	0.95	1.31	1.15	1.06	1.08
	Kubernetes	マイクロサービス	サービスメッシュ	APIゲートウェイ	イベント駆動型アーキテクチャ	可観測性ツール	不変インフラストラクチャの実践手法	カオス・エンジニアリング	フィーチャーフラグ	リモートプロシージャコール	フロントエンドゲートウェイ	マルチクラスタ管理

質問文: 過去12か月間に、バックエンドサービスにおいて以下のインフラストラクチャのアプローチや技術のいずれかを利用しましたか？もし利用した場合は、どれですか？

クラウドネイティブ機械学習への道のり

開発者の道のりにおいて、主な焦点は、開発者エコシステム全体がどのようにしてクラウドネイティブの成熟度を高めていくかを理解することですが、私たちは機械学習開発者にも注目しています。現在、機械学習や人工知能、とりわけクラウドネイティブがこれらのワークロードに取り組む組織をどのように支援できるかが大きな注目を集めている中、組織がクラウドネイティブ機能の拡充に注力する上で、この道筋を強調することは重要です。

全体として、その道筋や成熟度は類似していますが、非常に異なるイデオロギー的枠組みや技術的動機を示す、具体的かつ顕著な違いが存在します。AI開発者にとって、成熟度の道筋は、純粋なクラウドネイティブの高度化というよりも、AIサービスのライフサイクル段階を表しています。一般的なバックエンド開発者が基礎レベルから主流、そしてエリートレベルのクラウドネイティブ段階へと進むのに対し、AIチームはコンテナ化を経てデータパイプライン運用、さらにトレーニング／実験運用へと進み、その後、本番環境運用に向けたスキルアップを行います。トレーニング／実験段階で採用されるフィーチャーフラグやイミュータブルインフラストラクチャは、本番環境運用の成熟度を達成するための2つの前提条件となります。これは、イミュータブルインフラストラクチャ自体がエリートレベルの採用を意味する一般的な開発とは、明らかに異なるパターンです。

最も顕著な2つの違いは、不変インフラストラクチャがブリッジング技術へと移行したことと、AI開発者の間でリモートプロシージャコール(RPC)の重要性が高まったことです。AI開発者がクラウドネイティブスタックを成熟させるにつれ、不変インフラストラクチャがAI開発者にとってより重要な要件へと位置づけ直されたことは、機械学習における再現性の重要性を反映しています。一般的な開発において、不変インフラはカオスエンジニアリングとの関連性が最も強く(リフト値:2.22)、これは最高水準の信頼性確保の実践を示しています。一方、AI分野では、不変インフラはオブザーバビリティツールとの関連性がより強く(1.58 対 一般の1.33)、カオスエンジニアリングとの関連性はより弱くなっています(1.28 対 2.22)。

AI開発者の間では、不変インフラストラクチャは2つの目的を果たしていると考えられます。それは、再現性のあるトレーニング環境を確保することと、本番環境でのサービス提供に向けた一貫性のあるデプロイメント成果物を確保することです。AIチームは、クラウドネイティブの実践を探究する過程で、一般的な開発者よりも早い段階で不変インフラストラクチャを採用する傾向があり、これがカオスエンジニアリングとの関連性が比較的弱い理由の一つである可能性があります。多くのAIチームは、再現性が重要である一方、カオステストは重要ではない実験段階にとどまっています。

一般的なバックエンド開発ではほとんど目立たないリモートプロシージャコール(RPC)のパターンは、AIワークフローにおいては極めて重要であることが明らかになっています。RPCは、2番目に強い相関関係を示しています。RPCと機能フラグ設定の相関係数は2.07(一般の場合は1.16)です。

これは、モデル推論API、分散トレーニング、およびフィーチャーストアへのアクセスにおいて、RPCが中心的な役割を果たしていることを反映している可能性を示しています。

フィーチャーフラグとこの特異な関連性から、AIチームはモデルのバージョン切り替えなどの処理において、フィーチャーフラグへの依存度が高い可能性があることが示唆されます。RPCが数ある通信パターンのひとつに過ぎない一般的なバックエンド開発とは異なり、AIモデルの提供においては同期型のリクエスト・レスポンスがより重視されるため、RPCの導入は本番環境のAIチームの技術スタックにおいて重要な要素となっています。

これらを総合的に見ると、AI開発者の成熟度の道のりを別の視点から捉えることができます。クラウドネイティブの領域、すなわちKubernetesやマイクロサービスを活用する段階に入ると、彼らはデータパイプラインの運用を改善する方法を模索するようになります。これは、イベント駆動型アーキテクチャ(Kubernetes導入で1.31の向上)、ストリーミングサービス(イベント駆動型で1.50の向上)、およびオブザーバビリティツール(それぞれ1.25および1.34の向上)に見られます。これらはAI特有のニーズから生じています。イベント駆動型はパイプラインの再トレーニングをトリガーし、ストリーミングは継続的な特徴量エンジニアリングを処理し、可観測性はデータ品質とモデルのパフォーマンスを監視します。特に注目すべきは、可観測性がカオスエンジニアリングと負の相関(0.50)を示している点であり、これはインフラの耐障害性よりもデータやモデルのメトリクスに焦点を当てたAIの可観測性を裏付けています。

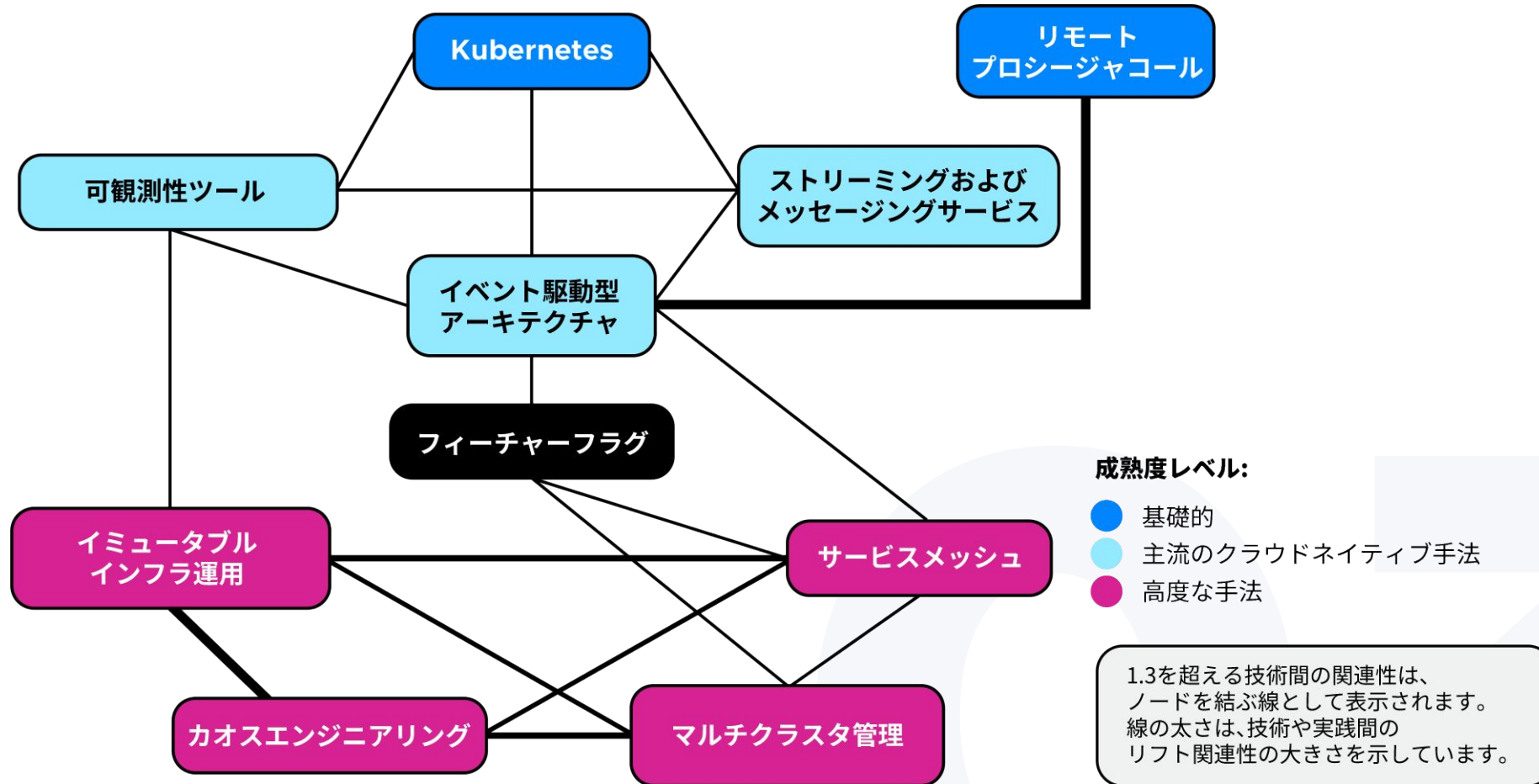
次の段階では、トレーニングや実験の運用改善に重点が置かれますが、フィーチャーフラグ、RPC、不変インフラストラクチャは、多くのAIチームにとって自然な到達点となり得ます。これらの技術やアプローチは、より高度な本番環境の構築を必要とすることなく、成熟したAIワークフロー（再現可能な実験、バージョン管理されたモデル、基本的なサービング）をサポートします。

この長い道のりの先には、サービスメッシュ、カオスエンジニアリング、マルチクラスター管理を中核に据えた、緊密に連携したエリートクラスターが待ち受けています。サービスメッシュはトラフィックの分割を実現し、カオスエンジニアリングはシステムの性能低下をシミュレートしてテストし、マルチクラスターは分散推論を可能にします。この高度な本番用クラスターを実現するには、前段階の機能フラグと不変インフラの両方が必要であり、そこから両者の間の相対的な難易度の差がうかがえます。

3. クラウドネイティブ成熟度向上への道

AI開発者のクラウドネイティブへの道のり

AI開発者間のバックエンド技術に関する連携を強化する (Q1 2026 n=1,270)



質問文: 過去12か月間に、バックエンドサービスにおいて以下のインフラストラクチャのアプローチや技術のいずれかを利用しましたか？もし利用した場合は、どれですか？

3. クラウドネイティブ成熟度向上への道

AI開発者間のバックエンド技術に関する連携を強化する

AI開発者間のバックエンド技術に関する連携を強化する (Q1 2026 n=1,270)

マイクロサービス	1.22											
サービスメッシュ	0.84	0.79										
APIゲートウェイ	1.09	1.11	0.92									
イベント駆動型アーキテクチャ	1.31	1.25	1.38	1.15								
可観測性ツール	1.25	1.06	1.05	0.99	1.25							
不変インフラストラクチャの実践手法	1.15	0.89	2.07	0.92	0.81	1.58						
カオス・エンジニアリング	0.78	0.81	2.36	0.50	0.84	0.50	1.28					
フィーチャーフラグ	1.26	0.99	1.35	1.02	1.45	1.41	1.10	1.10				
リモートプロシージャコール	1.11	1.13	0.69	1.17	1.01	0.94	1.19	0.59	2.07			
フロントエンドゲートウェイ	0.87	1.02	0.83	1.16	0.80	0.79	1.02	0.82	0.87	0.93		
マルチクラスタ管理	0.90	0.89	1.62	0.82	1.04	0.81	1.05	2.03	1.48	0.85	0.79	
ストリーミングおよびメッセージングサービス	1.21	1.23	1.18	0.97	1.50	1.34	1.13	0.97	1.53	1.19	1.07	0.99
	Kubernetes	マイクロサービス	サービスメッシュ	APIゲートウェイ	イベント駆動型アーキテクチャ	可観測性ツール	不変インフラストラクチャの実践手法	カオス・エンジニアリング	フィーチャーフラグ	リモートプロシージャコール	フロントエンドゲートウェイ	マルチクラスタ管理

質問文: 過去12か月間に、バックエンドサービスにおいて以下のインフラストラクチャのアプローチや技術のいずれかを利用しましたか？もし利用した場合は、どれですか？

AIチームの支援

この成熟度の段階モデルに基づき、クラウドネイティブ分野のAI開発者を支援する上で、AIのクラウドネイティブ導入を推進するための以下の戦略的優先事項が浮かび上がります：

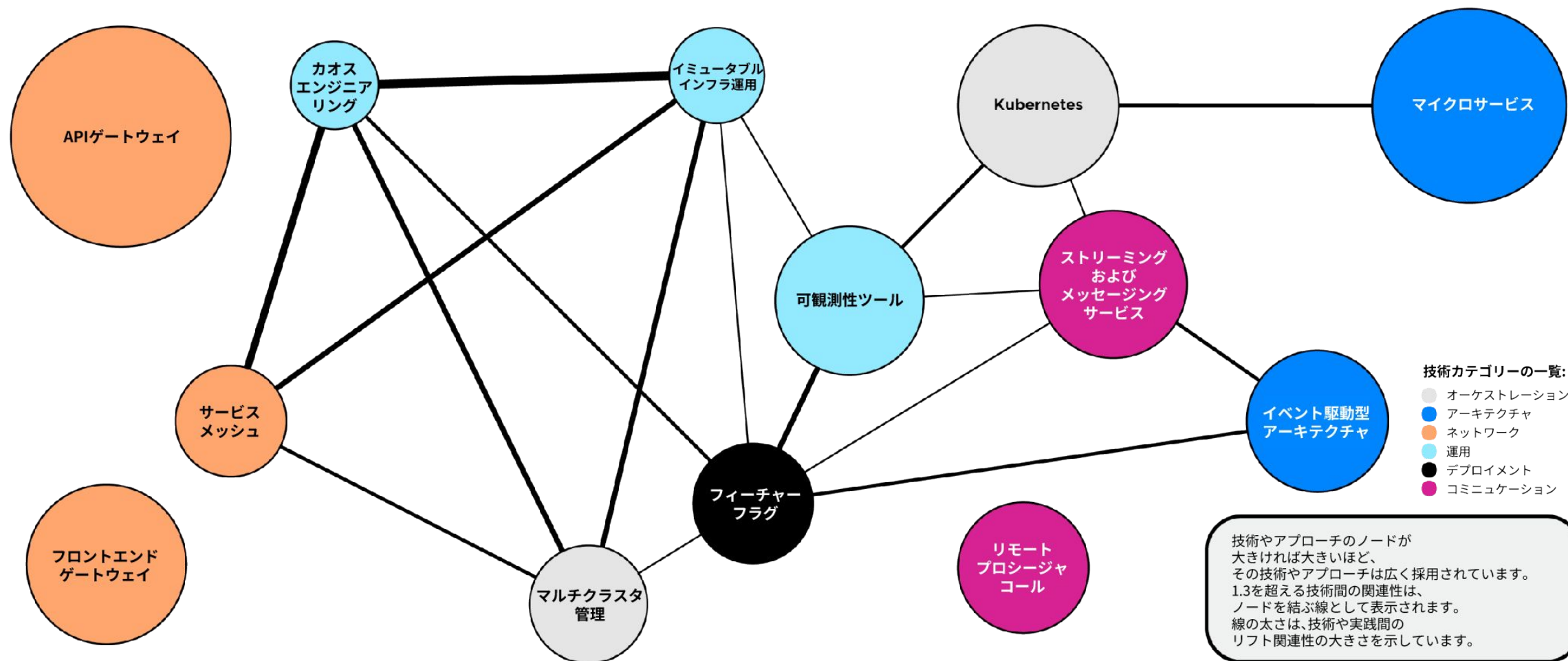
- すべてのAIチームが、本番環境レベルの高度な機能が必要というわけではありません。多くのチームは、実験やトレーニングの段階で十分に目的を達成しており、高度な配信インフラよりも、データパイプラインのツールや再現性の確保といった取り組みの改善から最大の恩恵を得ています。
- フィーチャーフラグと不変インフラは、AIの本番運用における2つの必須要件です。これらの中間段階の技術に関する教育を優先的に行うことで、AIチームが本番運用における成熟度に到達するのを妨げている主要な障壁を解消することができます。
- サービスメッシュ、カオスエンジニアリング、マルチクラスター管理には、AIに特化したアプローチが必要です。具体的には、モデルのA/Bテストのためのトラフィック分割、モデル性能低下のための耐障害性テスト、地理的に分散された推論などであり、一般的なインフラストラクチャのメリットとは異なります。
- AIクラウドネイティブ教育においては、RPCがAIサービス提供において中心的な役割を果たしている点を強調すべきです。なぜなら、モデル推論APIは本番環境での運用成熟度にとって基礎となるものだからです。



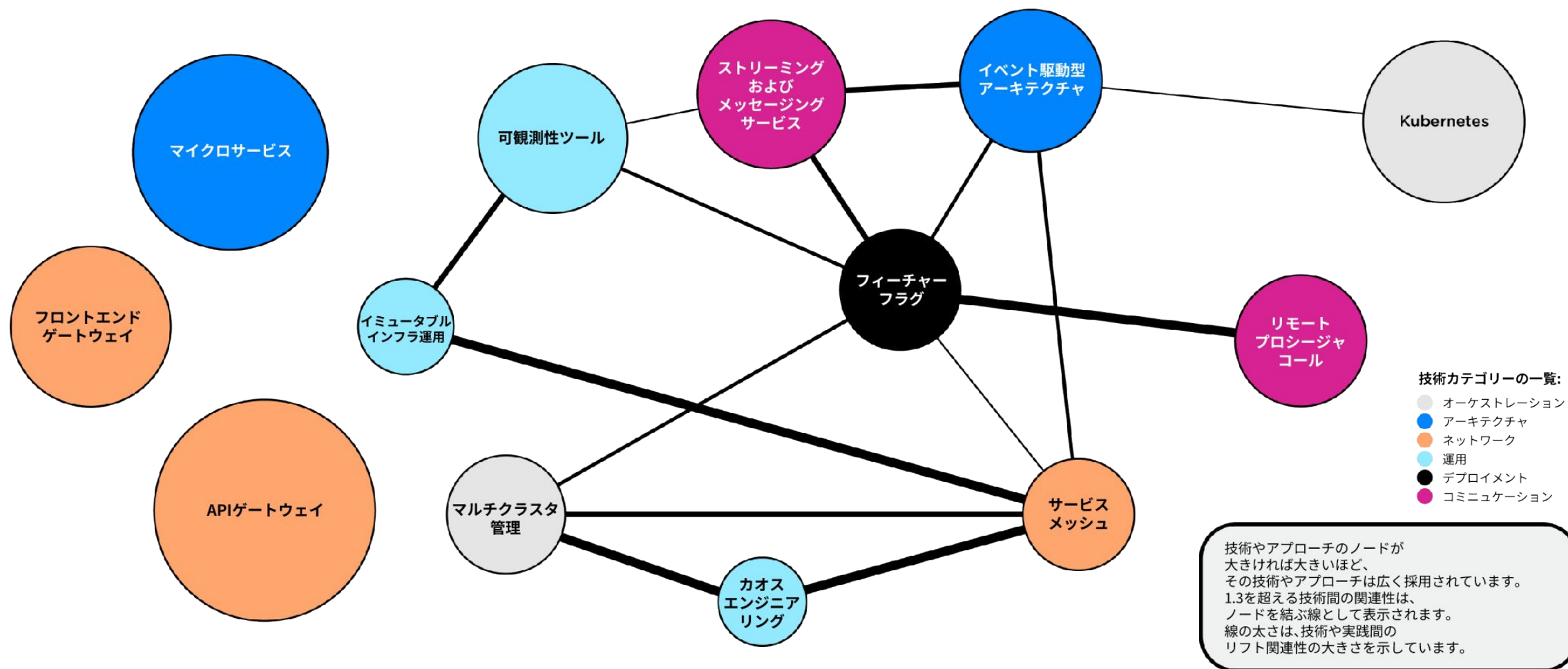
付録



バックエンド技術を利用する開発者間における技術リフトのネットワークグラフ



バックエンド技術を利用する AI開発者間の技術リフトを示すネットワークグラフ





調査方法



調査方法

リフトの計算

概要

リフト(リフト比や関心係数とも呼ばれる)は、2つの変数の個々の出現頻度を考慮しつつ、それらの間の関係の強さを測定するアソシエーションルールの指標です。技術の採用という文脈において、リフトは、単に両方が人気があるという理由だけで一緒に現れているのではなく、一貫したアーキテクチャパターンの一環として真に組み合わせて採用されている技術がどれかを明らかにします。

数学的定義

技術Aと技術B間のリフトは、次のように計算されます。

$$Lift(A,B) = P(A \cap B) / [P(A) \times P(B)]$$

ここで、

$P(A \cap B)$ は結合確率、すなわち技術Aと技術Bの両方を使用している開発者の割合です。

$P(A)$ は辺縁確率であり、技術Aを使用している開発者の割合です。

$P(B)$ は辺縁確率であり、技術Bを使用している開発者の割合です。

調査のサンプリング設計を考慮し、結果が開発者全体を代表するものとなるよう、すべての確率計算に調査ウェイトを適用します:

解釈

リフト指標は次のように解釈されます。リフト = 1.0: 独立性。技術Aと技術Bは、採用の決定が互いに独立している場合に予想される割合とまったく同じ割合で併用されています。これは、両技術の間に相関関係がないことを示しています。

Lift > 1.0: 正の相関。技術Aと技術Bは、予想される頻度よりも頻繁に併用されている。その大きさは相関の強さを示す:

1.0 < Lift < 1.3: 弱い正の相関

1.3 ≤ Lift < 1.7: 中程度の正の相関

Lift ≥ 1.7: 強い正の相関

Lift < 1.0: 負の相関。技術Aと技術Bは、偶然に期待される頻度よりも低い頻度で併用されている。



調査方法

「Developer Nation」調査

グローバルな展開

第31回「Developer Nation」グローバル調査の回答者の地域別分布
(2025年12月～2026年1月)

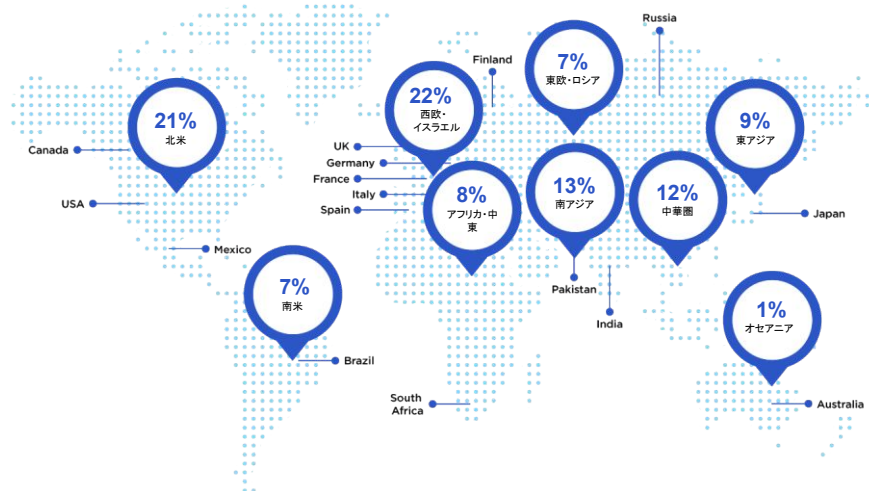
第31回
GLOBAL WAVE

12,500+

開発者数

95

カ国に展開



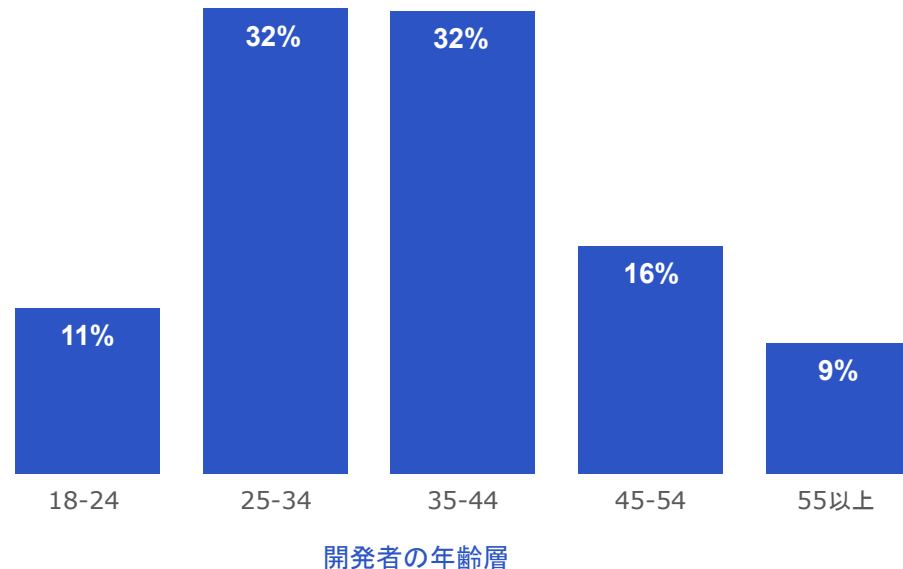
「Developer Nation」の第31回調査では、世界中から12,500人以上の回答者を集めました。これにより、「Developer Nation」シリーズは、モバイル、デスクトップ、産業用IoT、民生用電子機器、サードパーティ製アプリのエコシステム、バックエンド、Web、ゲーム、拡張現実(AR)／仮想現実(VR)、人工知能/機械学習の各分野の開発者およびデータサイエンティストを対象とした、これまでに実施された中で最も包括的な独立調査であり続けています。本レポートは、2025年12月から2026年1月にかけての7週間にわたり、SlashDataが企画・制作・実施した大規模なオンライン開発者調査に基づいています。

このオンライン調査の回答者は、米国、中国、インド、日本、ブラジル、英国を含む95カ国から集まりました。この調査の地理的範囲は、デベロッパー経済の世界的な規模を反映しています。このオンライン調査は、英語に加え9つの言語(簡体字中国語、繁体字中国語、フランス語、スペイン語、ポルトガル語、ベトナム語、ロシア語、日本語、韓国語)に翻訳され、ソフトウェア開発業界の主要パートナー14社および「Developer Nation」パネルを通じて周知されました。

回答者の年齢層は幅広く、18歳未満の若いプログラマーやクリエイターから、55歳以上のベテランまで多岐にわたりました。年齢の回答を希望しない方を除いた回答者の年齢分布は、以下の通りです。

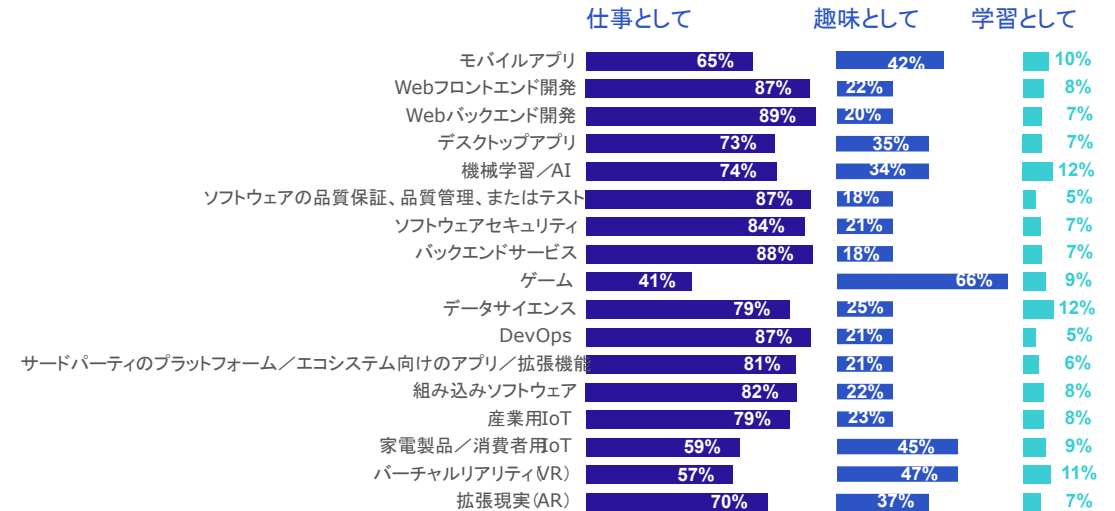
調査回答者の年齢分布

全体との比率(%) (Q1 2026 n=12,818)



各ソフトウェア分野において、回答者がどのように関与しているか

各地域における回答者の割合(%) (Q1 2026 n=12,021)



回答者には、調査対象の13種類のプロジェクト(ウェブアプリ/SaaS、モバイルアプリ、デスクトップアプリ、バックエンドサービス、拡張現実、仮想現実、ゲーム、データサイエンス、機械学習/人工知能、産業用IoT、民生用電子機器、組み込みソフトウェア、およびサードパーティ製アプリエコシステム向けのアプリ/拡張機能)のうち、どの種類のプロジェクトに関与しているかを尋ねました。また、回答者には、自身が関与している分野を「プロフェッショナル」「愛好家」「学生」、あるいはこれらの組み合わせのいずれとして取り組んでいるか、および各分野での経験年数についても回答していただきました。

地域ごとのサンプリングバイアスの影響を排除するため、9つの地域にわたる地域別分布に対し、「Developer Nation」調査で特定された地域別分布および成長傾向に基づいて決定された係数で重み付けを行いました。また、各アウトリーチチャンネルにおけるその他の重要なサンプリングバイアスを最小限に抑えるため、回答に重み付けを行い、使用されている技術および開発者セグメントについて代表的な分布を導き出しました。アンサンブルモデリング手法を用い、独立した代表的なチャンネルからのデータに基づいて加重分布を導出しました。この際、調査パートナーのチャンネルは除外し、これらのチャンネルを通じて募集された回答者に起因するサンプリングバイアスを排除しました。産業用IoT、民生用電子機器、サードパーティ製アプリエコシステム、バックエンド、組み込み、拡張現実(AR)、仮想現実(VR)の各分野について、個別に重み付けを行った後、それらを統合しました。

当社の調査方法に関する詳細については、以下のサイトをご覧ください

: <https://www.slashdata.co/methodology>.



調査方法



Liam Bollmann - Dodd

*Principal Market Research
Consultant*

Liamは元実験反物質物理学者であり、CERNに勤務中に物理学の博士号を取得しました。彼は、クラウド技術の進化、サイバーセキュリティ、そして技術の発展とそれが社会に与える影響との関係について関心を持っています。

✉ liam.dodd@slashdata.co



Álvaro Ruiz Cubero

Research Manager

Álvaroは、戦略およびオペレーションコンサルティングの経験を持つ市場調査アナリストです。経営管理学の修士号を取得しており、データに基づく意思決定の力を強く信じています。アルバロは、企業が複雑な戦略的課題に取り組み、綿密な調査と分析に裏打ちされた戦略的決定を下せるよう支援することに情熱を注いでいます。

✉ alvaro.ruiz@slashdata.co

著者について

Navigate **AI** **technology decisions** **with confidence & clarity**

SlashData is an AI analyst firm which has been working with the top Tech brands to provide clarity and confidence in their decision-making.

For 20 years, we have been tracking software technology trends and helping technology brands make product and marketing investment decisions, challenging assumptions and reframing market trends to empower industry leaders to drive the world towards the future.

Find us at slashdata.co





/DATA



CLOUD NATIVE
COMPUTING FOUNDATION