

Accelerate

2021 年の State of DevOps



Sponsored by



1 章

エグゼクティブ サマリー

主な調査結果 06

2 章

比較方法

ソフトウェアデリバリーと運用
のパフォーマンス 09

3 章

改善のための提案

クラウド	15	セキュリティ	24
SRE と DevOps	19	技術面の DevOps 機能	26
ドキュメント	21	COVID-19	29
		文化	31

4 章

調査対象者

ユーザー属性と企業特性 34

5 章

最後に

6 章

謝辞

7 章

作成者

8 章

手法

9 章

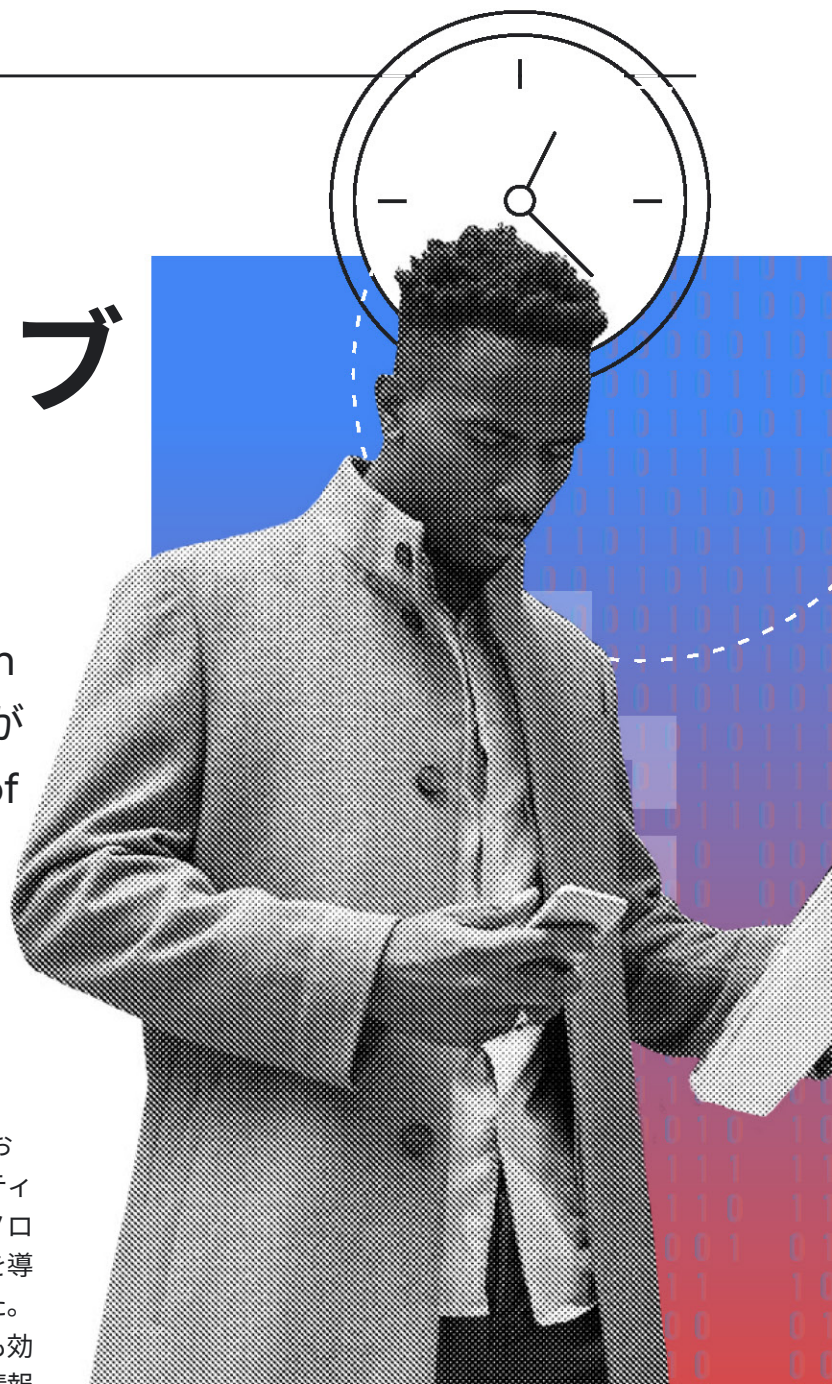
関連情報

1 章

エグゼクティブ サマリー

Google Cloud の DevOps Research and Assessment (DORA) チームが作成した今年の Accelerate State of DevOps レポートは、32,000 を超える全世界の専門家に対する 7 年間の調査と、それらから得られたデータを示すものです。

私たちの調査では、ソフトウェアデリバリー、運用、および組織的なパフォーマンスを推進する能力とプラクティスを調べました。厳密な統計的技法を活用して、テクノロジーの提供における優越性と、ビジネスの優れた結果を導くことのできる、プラクティスへの理解を目指しました。この目標に向けて、テクノロジーを開発、提供する最も効果的で効率的な方法について、データドリブンの分析情報を提示します。



調査により、優れたソフトウェアデリバリーと運用パフォーマンスが、技術変革における組織的なパフォーマンスを促進することがあらためて実証されました。チームが自分たちを業界と比較してベンチマークできるようにするため、私たちはクラスタ分析を使用して意味のあるパフォーマンスカテゴリを形成しました（低、中、高、エリートというパフォーマンス別のグループなど）。チームが、自分たちの現在のパフォーマンスが業界と比べてどの程度のものかを認識した後で、私たちの予測分析で発見された事項を使用して、主要な結果を改善し、最終的に自分たちの相対的な地位が向上するように、プラクティスと能力の目標を設定できます。今年は、信頼性のターゲットを満たし、ソフトウェアのサプライチェーン全体にわたってセキュリティを統合して、高品質の内部ドキュメントを作成し、クラウドのポテンシャルを最大限に活用することの重要性を強調しています。また、肯定的なチーム文化が、COVID-19（新型コロナウイルス感染症）パンデミックの結果としてリモートで勤務することの影響を軽減できるかどうかも探求しました。

意味のある改善を行うには、チームが継続的な改善という原理を採用する必要があります。ベンチマークを使用して現在の状況を測定し、調査された能力に基づいて制約を識別し、それらの制約を緩和するための改善を試みてください。探求には勝利と失敗の両方が伴いますが、どちらの場合もチームは得られた教訓の結果として有意義な行動をとることができます。

主な調査結果

01 最もパフォーマンスが高いグループは成長し続け、水準はさらに上がっています。

調査では、卓越したパフォーマンスのグループが全体の26%を占めるようになり、本番環境の変更のリードタイムも短縮しました。業界はさらに加速し続けており、チームはそれによって有意義な利益を受けています。

02 SRE と DevOps は相補的な理念です。

サイト信頼性エンジニアリング（SRE）で概説されている最新の運用プラクティスを活用しているチームは、より高い運用パフォーマンスを報告しています。デリバリーのパフォーマンスと運用のパフォーマンスの両方を重視するチームは、組織的なパフォーマンスが最も高くなることが報告されています。

03 より多くのチームがクラウドを活用し、それによって大きな恩恵を受けています。

各チームはワークロードをクラウドに移行し続けており、クラウドの5つの能力すべてを活用したチームにはソフトウェアデリバリーと運用（SDO）のパフォーマンスや、組織としてのパフォーマンスの向上が見られました。マルチクラウドの導入も増加傾向にあり、チームが各プロバイダ独自の機能を活用できるようになってきています。

04 安全なソフトウェア サプライ チェーンは 不可欠であると同時に、パフォーマンス を引き上げます。

近年における悪意のある攻撃の大幅な増加を踏まえて、組織は事後対応的なプラクティスから予防的および診断的な方式に転換していく必要があります。ソフトウェアのサプライチェーン全体にセキュリティ プラクティスを統合したチームは、ソフトウェアを迅速に、高い信頼性で、安全に提供しています。

05 良質なドキュメントの作成は、DevOps を 効果的に機能させるための基礎です。

今回の調査では初めて、内部ドキュメントの品質と、その品質を実現するために採用されたプラクティスを測定しました。高品質のドキュメントを保有するチームは、技術的なプラクティスをよりの確に実装でき、全体的なパフォーマンスも優れています。

06 チームに肯定的な文化があれば、困難な 状況での燃え尽き症候群を軽減できます。

チームの文化は、チームがソフトウェアを提供し、組織的な目標を達成する、またはそれを超えた成果を得るうえで大きな差異をもたらします。ジェネレーティブ^{1,2}な文化を持つ包括的なチームは、COVID-19 パンデミックの最中にも燃え尽き症候群はあまり発生しませんでした。

¹ Westrum の類型論の組織文化によると、ジェネレーティブなチーム文化とは、協調性が高く、分断化を排除して、失敗を調査につなげ、意思決定のリスクを共有するようなチームとされています。

² Westrum, R. (2004 年)。「A typology of organizational cultures」(英語)、BMJ Quality & Safety、

13 (補遺 2)、ii22~ii27。

2 章 比較方法

自分のチームが業界の他のチームと比較してどの程度のものかとお考えでしょうか？このセクションでは、DevOpsのパフォーマンスに関する最新のベンチマーク評価について説明します。

チームがソフトウェアシステムをどのように開発、デリバリー、運用するかを調べ、回答者をエリート、高、中、低パフォーマンスの4つに分類しました。チームのパフォーマンスを各クラスタのパフォーマンスと比較することで、このレポート全体で解説している調査結果に関して自分のチームがどのような位置にあるかを把握できます。



ソフトウェア デリバリーと運用のパフォーマンス

変化し続ける業界からの要求に対応するため、組織はソフトウェアを迅速に、高い信頼性で提供し、運用する必要があります。チームがソフトウェアへの変更を迅速に行えるほど、お客様に早く価値を提供し、実験を行い、貴重なフィードバックを得ることができます。Googleは7年間にわたってデータの収集と調査を行った結果、ソフトウェアデリバリーのパフォーマンスを測定するための4つの指標を作成し、検証しました。2018年から、組織の能力を捕捉するため5つの指標も追加しました。

5つの基準すべてに秀でているチームは、組織として非常に優れたパフォーマンスを示しています。Googleは、これら5つの基準をソフトウェアデリバリーと運用(SDO)のパフォーマンスと呼んでいます。これらの指標はシステムレベルの成果に重点を置いたものです。ソフトウェア指標の一般的な落とし穴、たとえば機能が互いに相反する、ローカルな最適化により総合的な性能が低下するという事態を避けるために役立ちます。

ソフトウェアデリバリーのパフォーマンス指標	エリート	高	中	低
🔄 デプロイの頻度 主なアプリケーションまたはサービスで、組織が本番環境にコードをデプロイする頻度、またはエンドユーザーにリリースする頻度はどのくらいか。	必要に応じて (1日に複数回のデプロイ)	週1回から 月1回の間	月1回から 6か月に 1回の間	6か月に 1回より 少ない
🕒 変更のリードタイム 主なアプリケーションまたはサービスで、変更のリードタイム (commitされたコードが本番環境で正常に実行されるまでの時間) はどのくらいか。	1時間未満	1日から 1週間の間	1か月から 6か月の間	6か月より 長い
🔄 サービス復旧時間 主なアプリケーションまたはサービスで、サービスのインシデントや、ユーザーに影響を与える障害 (計画外のサービス停止やサービス障害など) が発生した場合、サービスの復旧に通常どれくらいの時間がかかるか。	1時間未満	1日未満	1日から 1週間の間	6か月より 長い
⚠️ 変更時の障害率 主なアプリケーションまたはサービスで、本番環境に変更を加えた、またはユーザーに変更をリリースしたとき、サービスの低下 (サービス障害、サービスの停止など) が発生して、対策 (修正プログラム、ロールバック、フィックスフォワード、パッチなど) が必要になった割合はどのくらいか。	0%~15%	16%~30%	16%~30%	16%~30%

デリバリー パフォーマンスの4つの指標

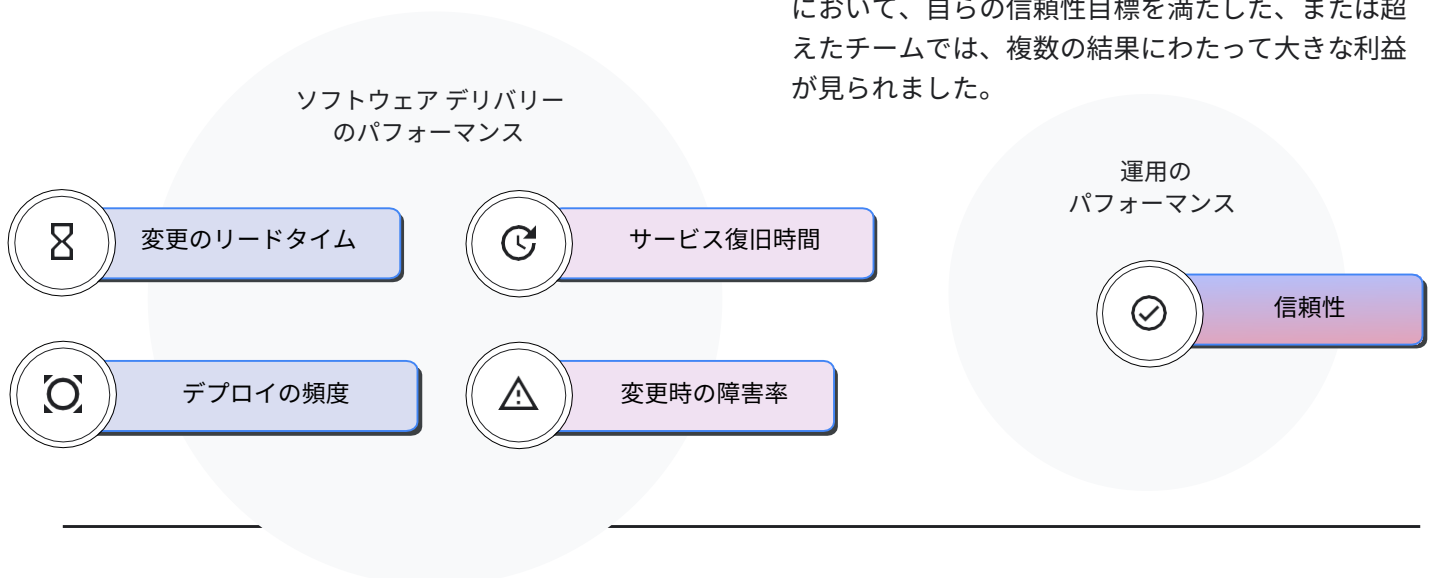
ソフトウェアデリバリーのパフォーマンスに関する4つの指標は、スループットと安定性に関するものと考えられます。調査では、**コード変更のリードタイム**（すなわち、コードのcommitから本番環境へのリリースまで）と**デプロイの頻度**を使用してスループットを測定しました。安定性は、**インシデント発生後のサービス復旧までの時間**と**変更時の障害発生率**を使用して測定しました。

ここでも、4つのソフトウェアデリバリー指標を分析することで、エリート、高、中、低という4つの異なるパフォーマンスプロファイルが明らかになりました。これらのグループ間にはスループットと安定性の測定値において統計的に有意な差が見られます。これまでの年と同様に、最もパフォーマンスの高いチームは4つの指標すべてにおいて明確に優れており、パフォーマンスの低いチームはすべての分野において明確に劣っています。

5番目の指標: 可用性から信頼性まで

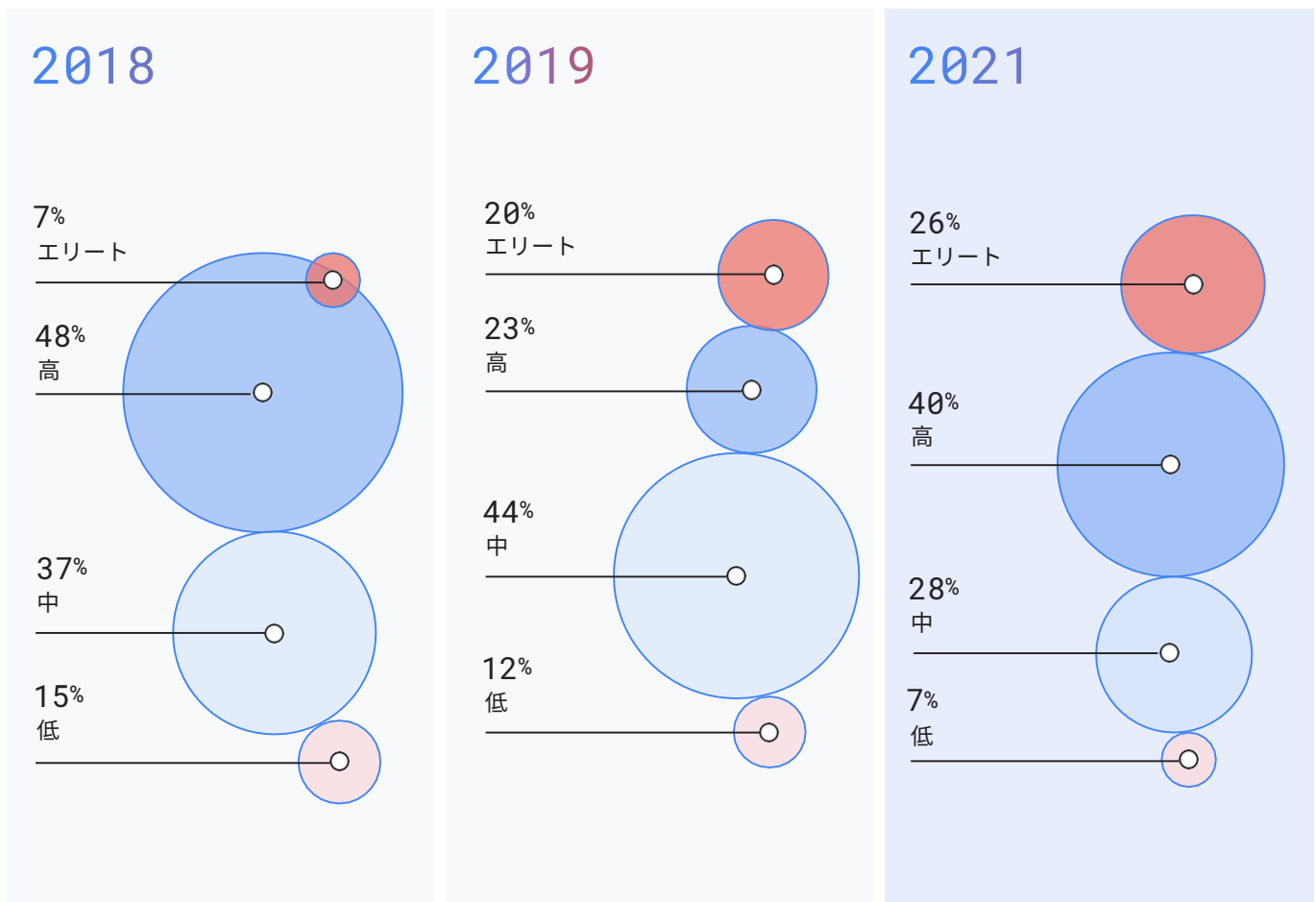
5番目の指標は**運用のパフォーマンス**を表すもので、最新の運用プラクティスを測定します。運用パフォーマンスの主な指標は**信頼性**で、チームが運用するソフトウェアについての期待と主張をどの程度満たすことができるかを示します。これまでの調査では信頼性ではなく可用性を測定してきましたが、可用性は信頼性エンジニアリングに特有の指標であるため、測定の幅を信頼性まで広げました。これにより、可用性、レイテンシ、パフォーマンス、スケーラビリティがより広く示されるようになりました。具体的には、回答者に、信頼性に関する目標を満たす、または上回る能力を評価するように依頼しました。デリバリーのパフォーマンスレベルの異なる複数のチームで、デリバリーだけでなく運用のパフォーマンスも重視する場合に、より優れた結果を得られることが確認されました。

以前のレポートと同様に、エリートの評価を得たチームと評価の低いチームとを比較し、特定の能力がどのような影響を及ぼすかを明らかにしました。しかし今年は、運用のパフォーマンスの影響も考慮に入れることを目指しました。デリバリーパフォーマンスのすべてのカテゴリ（低からエリートまで）において、自らの信頼性目標を満たした、または超えたチームでは、複数の結果にわたって大きな利益が見られました。



ますます加速していく業界

業界は年々進化し続け、ソフトウェアをより迅速に、より高い安定性で提供する能力が進歩していきます。調査開始以降初めて、回答者の2/3が高およびエリートのパフォーマンスを挙げているチームとなりました。さらに、今年エリートの成績を挙げたチームはさらに水準が上がり、以前の評価と比べても変更のリードタイムが短縮されました（たとえば、2019年には1日未満だったものが、2021年には1時間以内になりました）。また、これも初めてのこととして、エリートの成績を挙げたチームだけが変更の障害率を最小化しました。今年以前では中や高の成績を挙げたチームも変更の障害率を最小化していました。



スループット

デプロイの頻度

これまでの年と同様に、エリートのグループは常時オンデマンドでデプロイを行い、1日に何回もデプロイを行っていることを報告しました。これに対して、低のパフォーマンスだったグループはデプロイが6か月に1回未満（1年に2回未満）だと報告しており、同様に2019年と比較してパフォーマンスが低下しています。正規化された年間デプロイ回数は、最もパフォーマンスの高いグループで1,460回（1日4回×365日と計算）、低のパフォーマンスだったグループでは年間1.5回（年に2回と1回の平均）でした。この分析から、エリートのパフォーマンスを得たグループは、低のパフォーマンスのグループよりもコードを約973倍の頻度でデプロイしていることが示されました。

変更のリードタイム

2019年からの改善点として、エリートのパフォーマンスのグループはリードタイムが1時間未満になりました。このリードタイムは、コードがcommitされてから、本番環境に正しくデプロイされるまでの時間で計測したものです。2019年に最も高いパフォーマンスだったグループの報告したリードタイムは1日未満だったため、これと比べてもパフォーマンスが向上しています。エリートのパフォーマンスのグループと比べて、低のパフォーマンスのグループは6か月を超えるリードタイムを必要としていました。リードタイムが、エリートのパフォーマンスのグループでは1時間（「1時間未満」を多めにみた推定値）なのに対して、低のパフォーマンスのグループでは6,570時間（1年の8,760時間と6か月分の4,380時間の平均で計算）で、エリートのグループは低のグループよりも変更のリードタイムが6,570倍速くなっています。

安定性

サービス復旧時間

エリートのグループはサービス復旧時間が1時間未満だと報告したのに対して、低のグループは6か月を超えると報告しました。この計算では、控えめな推定時間として、高のパフォーマンスのグループでは1時間、低のパフォーマンスのグループでは1年（8,760時間）と6か月（4,380時間）の中間値としました。これらの数値に基づいて、エリートのグループは低のグループよりも6,570倍も速くサービスを復旧できます。2019年と比べて、サービス復旧時間のパフォーマンスはエリートのパフォーマンスのグループでは変化なしで、低のパフォーマンスのグループでは増加しました。

変更時の障害率

エリートのパフォーマンスのグループでは変更時の障害率が0%~15%なのに対して、低のパフォーマンスのグループでは変更時の障害率が16%~30%です。これら2つの範囲の値に対する中間値では、エリートのパフォーマンスのグループでは変更時の障害率が7.5%、低のパフォーマンスのグループでは23%になります。エリートのパフォーマンスのグループでは、低のパフォーマンスのグループと比べて、変更時の障害率が1/3です。今年、エリートのパフォーマンスのグループでは障害率が変化せず、低のパフォーマンスのグループでは2019年よりも改善されましたが、その中間のグループでは悪化しました。

エリート パフォーマンスのグループ

パフォーマンスがエリートのグループと低のグループを比べると、エリートのグループは次の点で優れています。

973倍

コードデプロイの頻度

6,570倍

Commit からデプロイまでの
リードタイムの高速化

誤植ではありません。
これが正しい
数値です。

3倍

変更時の障害率の低さ
(変更が失敗する可能性が 1/3)

6,570倍

インシデントからの
回復時間の短縮

3 章

改善のための 提案

SDO と組織的なパフォーマンスを改善するにはどうしたらいいでしょうか？調査結果から、パフォーマンスを推進する能力に集中するために役立つ、証拠に基づいたガイドを提供します。

今年のレポートでは、クラウド、SRE プラクティス、セキュリティ、技術的な実践方式、文化の影響について調べました。このセクションを通して、これらの能力のそれぞれについて紹介し、それらが各種の結果にどのような影響を与えるかについて言及します。

DORA の State of DevOps 調査モデルについてご存じの方のために、今年のモデルと、過去のモデルすべてをホストしたオンラインリソースを作成しました³。



³ <https://devops-research.com/models.html>

クラウド

Accelerate State of DevOps 2019 のときと同様に、マルチクラウドやハイブリッドクラウドソリューションを選択する組織の数は増え続けています。Google のアンケートで、メインのサービスまたはアプリケーションがホストされる場所を回答者に質問したところ、パブリッククラウドの利用が増加していました。回答者の56%はパブリッククラウド（複数のパブリッククラウドを使用している回答者も含む）を使用していると回答し、これは2019年よりも5%増えています。今年はマルチクラウドの使用についても別の質問を行い、回答者の21%は複数のパブリッククラウドにデプロイしていると回答しました。回答者の21%はクラウドを使用しておらず、代わりにデータセンターやオンプレミスのソリューションを使用していると回答しました。最後に、回答者の34%はハイブリッドクラウドを、29%はプライベートクラウドを使用していると回答しました。

メインのサービスまたはアプリケーションがホストされる場所



ハイブリッドやマルチクラウドによってビジネスの成果を促進

今年見られたのは、ハイブリッドやマルチクラウドの増加で、ビジネスにとって重要な成果にも明確な影響を及ぼしています。ハイブリッドやマルチクラウドを使用している回答者は、使用していない回答者よりも、組織のパフォーマンス目標を超える割合が1.6倍も高くなっています。また、SDOへの強い影響も見られました。ハイブリッドやマルチクラウドのユーザーは、デプロイの頻度、変更のリードタイム、復旧時間、変更時の障害率、信頼性について優れている可能性が1.4倍高くなっています。

マルチクラウドを使用する理由

2018年の評価と同様に、複数のパブリッククラウドプロバイダを使用する理論的根拠を回答者にたずねました。今年は、該当する選択肢をすべて選んでもらうのではなく、複数のプロバイダを使用する一番の理由について回答してもらいました。回答者の1/4以上（26%）は、各クラウドプロバイダに固有の利点を活用するために、複数のプロバイダを使用していました。すなわち、回答者が追加のプロバイダを選択するときは、現在のプロバイダと他の選択肢との差別化要因を探していると推定されます。マルチクラウドに移行する理由として2番目に多かったのは可用性です（22%）。複数のクラウドプロバイダを採用した回答者において、自分たちの信頼性目標を満たす、または超えている割合が1.5倍も高かったことは驚くにあたりません。

複数のプロバイダを利用する主な理由

各プロバイダ固有の利点の活用	26%
可用性	22%
障害復旧	17%
法令遵守	13%
その他	08%
交渉戦術または調達要件	08%
プロバイダ1社では信頼できない	06%

ベンチマークの変更

クラウド インフラストラクチャの実装方法が重要

これまでの調査で、すべての回答者がクラウドを同じ方法で採用しているわけではないことが判明しています。このため、クラウドの採用がどれだけビジネス成果に有効となるかという点について差異が発生します。この調査では、米国国立標準技術研究所（NIST）で定義されたクラウドコンピューティングの本質的な特性に的を絞り、これを指針とすることで、この制限に対処します。NISTによるクラウドコンピューティングの定義を使用して、クラウド採用がSDOに及ぼす影響を調査するだけでなく、重要なプラクティスがSDOのパフォーマンスに及ぼす影響を調査しました。

本当に重要なのは、チームがクラウドテクノロジーを使用するというだけでなく、チームがどのようにクラウドサービスを実装するかだということです。ここでも明らかになっています。エリートのパフォーマンスのグループは、すべての本質的なNISTクラウド特性を満たしている割合が3.5倍も高くなっています。クラウドインフラストラクチャを使用していると回答した回答者のうち、NISTにより定義されたクラウドコンピューティングの本質的な特性5つのすべてを満たしているかという点について、そう思う、または非常にそう思うと答えたのはわずか32%で、2019年よりも3%増加しています。全体として、NISTが定義するクラウドコンピューティングの特性を使用している割合は14~19%増加しており、そのうち「迅速な弾力性」が最も増加しています。

オンデマンド、セルフサービス

ユーザーは、必要に応じてコンピューティングリソースを自動的にプロビジョニングでき、プロバイダ側で人手による介入は必要ありません。

広範なネットワークアクセス

機能が広範に利用でき、スマートフォン、タブレット、ノートパソコン、ワークステーションなど複数のクライアントからアクセス可能です。

73%

オンデマンドのセルフサービスを使用している回答者の割合、2019年よりも16%増加

74%

広範なネットワークアクセスを使用している回答者の割合、2019年から14%増加

リソースのプーリング

プロバイダのリソースはマルチテナントモデルでプールされ、物理および仮想リソースがオンデマンドで動的に割り当ておよび再割り当てされます。顧客は一般に、提供されるリソースの正確な場所を直接コントロールできませんが、より高レベルの抽象化で、たとえば国、州、データセンターなどの場所を指定できます。

迅速な弾力性

能力を弾力的にプロビジョニングしてリリースし、必要に応じて迅速にスケールアウトまたはスケールインできます。ユーザーからは、無制限の能力がプロビジョニング可能で、いつでもどれだけでも充当できるように見えます。

メジャード サービス

クラウドシステムは、サービスのタイプに適切なレベルの抽象化、たとえばストレージ、処理、帯域幅、アクティブなユーザー アカウントなどで測定機能を利用し、リソースの使用を自動的にコントロールして最適化できます。リソースの使用をモニタリング、コントロール、報告することで透明性を実現できます。

73%

リソースのプーリングを使用している回答者の割合、
2019年から15%増加

77%

迅速な弾力性を使用している回答者の割合、
2019年から18%増加

78%

メジャード サービスを使用している回答者の割合、
2019年から16%増加

SRE と DevOps

DevOps コミュニティが公共の会議や話題に登場し始めたころ、Google の中でも同じ志の運動であるサイト信頼性エンジニアリング (SRE) が形成されていました。SRE や、Facebook のプロダクションエンジニアリング分野など同様な手法は、DevOps を動機付けたのと同じ目標と技法の多くを取り入れています。SRE は 2016 年に、サイト信頼性エンジニアリングに関する最初の本⁴が刊行されたとき、一般的に知られるようになりました。それ以後にこの運動は成長し、今日では SRE 実務者が技術的な運用のプラクティスに関して協力し合う国際的なコミュニティとなりました。

おそらく必然でしょうが、混乱も生まれました。SRE と DevOps との相違は何なのか？どちらかを選ぶ必要があるのか？どちらの方が良いものなのか？実際のところ、この 2 つは競合するものではありません。SRE と DevOps は補完性が高いもので、調査から双方の連携が示されています。SRE は学習の規律で、職能間にわたる連絡と心理的な安全性を優先しています。これらはエリート DevOps チームに典型的な、パフォーマンス指向のジェネレーティブな文化の中核をなすものと同じ価値です。SRE はその中核的な規律を拡大し、サービスレベル指標 / サービスレベル目標 (SLI / SLO) 指標フレームワークを含む実践的な技法を提供しています。リーンプロダクト フレームワークで、Google の調査で明らかになった迅速な顧客フィードバックサイクルを達成する方法が指定されているように、SRE フレームワークでは、チームがユーザーに対する約束を常に果たし続ける能力を向上させるための実践とツールが定義されています。

Google は 2021 年に、運用への質問の幅を、サービスの可用性の分析から、信頼性のより一般的なカテゴリへと拡大しました。今年の調査では、SRE の実践からヒントを得て、チームが次のような行動をどの程度まで行っているかを評価するため、いくつかの項目を付け加えました。

- ユーザーへの対応に関して信頼性を定義する
- SLI / SLO 指標フレームワークを採用して、エラー バジレットにより作業に優先順位を付ける
- 自動化を利用して、手作業と、仕事の妨げになるアラートを削減する
- インシデント対応の Protokol と、対応準備をテストするための演習を定義する
- ソフトウェア デリバリーのライフサイクル全体にわたり、信頼性に関する原則を取り入れる (「信頼性のシフトレフト」)

結果を分析すると、このような最新のオペレーションのプラクティスに優れているチームは、優れた SDO パフォーマンスを報告する可能性が 1.4 倍も高く、ビジネス上の良好な成果を報告する可能性も 1.8 倍高いことが明らかになりました。

SRE のプラクティスは、調査対象となったチームの大部分に取り入れられていました。回答者の 52% は、これらのプラクティスのある程度の範囲で使用していると報告しましたが、どの程度採用しているかはチーム間で大きく異なっています。データから、これらの手法を使用することで、より高い信頼性と、より高い総合的な SDO パフォーマンスを期待できます。SRE により DevOps の成功を促進できます。

⁴ Betsy Beyer その他による共同編集、「Site Reliability Engineering」(英語) (O'Reilly Media, 2016 年)。

さらに、開発者とオペレータがどの程度共同で信頼性に取り組むかに反映される運用責任の共有モデルは、信頼性に関しても良い結果を期待できることが明らかになりました。

パフォーマンスの目標となる測定値の改善だけでなく、SREによって作業の実務担当者向けの環境も改善されます。一般に、運用タスクの過剰な負荷を負わされた人物は燃え尽き症候群になりやすいものですが、SREはプラス効果があります。チームがSREプラクティスを採用するほど、メンバーが燃え尽き症候群を体験する可能性が低くなることを発見しました。SREはリソースの最適化にも役立ちます。SREプラクティスの適用によって信頼性目標を満たしたチームは、SREプラクティスを実践していないチームよりも多くの時間をコードの作成に費やしていると報告しています。

調査により、低からエリートまで、どのSDOパフォーマンスレベルのチームも、SREプラクティスの使用を増やすことで恩恵を受けられる可能性が高いことが明らかになりました。チームのパフォーマンスが高いほど、現代的な運用モードを採用する可能性も高くなります。エリートのパフォーマンスのグループは、低のパフォーマンスのグループよりも、SREプラクティスを使用していると回答する割合が2.1倍も高くなっています。しかし、最高レベルで運用を行っているチームでも成長の余地はあります。エリート回答者のうち、Googleが調査したすべてのSREプラクティスを完全に実装していたことが示されたのはわずか10%です。各産業におけるSDOパフォーマンスが進歩し続けるにつれ、各チームの運用の手法は、継続的なDevOps改善の重要な推進力となります。

52%

SREプラクティスを使用していると報告した回答者の割合

エリートのパフォーマンスのグループは、低のパフォーマンスのグループよりも、SREプラクティスを使用している割合が2.1倍高くなっています。しかし、最高レベルで運用を行っているチームでも成長の余地はあります。エリート回答者のうち、Googleが調査したすべてのSREプラクティスを完全に実装していたことが示されたのはわずか10%です。

ドキュメント

今年の調査では、内部ドキュメントの品質を調査しました。これは、チームが扱っているサービスやアプリケーションのマニュアル、README、コードのコメントなどのドキュメントを意味します。ドキュメントの品質は、そのドキュメントが次の条件をどの程度満たしているかで測定されました。

- 読者が目的を達成するため役立つ
- 正確かつ最新の内容で、理解しやすい
- 情報が見つけやすく、よく整理されており、明瞭である⁵

内部システムについての情報の記録とアクセスは、チームの技術的な作業の重要な部分です。調査から、回答者の約25%が良い品質のドキュメントを保有していることがわかっており、そのドキュメント作業の影響は明白です。高品質のドキュメントを保有しているチームは、ソフトウェアデリバリーと運用（SDO）パフォーマンスも優れている割合が2.4倍です。良いドキュメントを保有しているチームは、そうでないチームと比べて、ソフトウェアをより迅速に、高い信頼性で提供できます。ドキュメントが完璧である必要はありませんが、調査では、ドキュメント品質のどのような改善も、パフォーマンスに肯定的かつ直接的な影響があることが示されています。

今日のテクノロジー環境は、システムとして日々複雑化しつつあり、それらのシステムの異なる側面について専門家や、特化した役割が存在します。セキュリティからテストまで、ドキュメントは専門のサブチームと、より広範なチームの両方にとって、専門的な知識とガイダンスを共有する主要な方法となります。

ドキュメントの品質から、そのチームが技術的なプラクティスの実装に成功するかどうかを予測できることが明らかになっています。さらにこれらのプラクティスは、システムの技術的な能力、たとえばオペラビリティ、継続的なテスト、デプロイの自動化などの改善も予測できます。高品質のドキュメントを保有するチームは、次のような優位点があることが判明しました。

- セキュリティ プラクティスを実装する割合が3.8倍
- 自らの信頼性目標を満たす、または超える割合が2.4倍
- サイト信頼性エンジニアリング（SRE）プラクティスを実装する割合が3.5倍
- クラウドを完全に活用する割合が2.5倍

⁵ 技術ドキュメントについての次のような既存の調査から得られた品質指標

— Aghajani, E 他、(2019年)。「Software Documentation Issues Unveiled」(英語)、2019年のIEEE/ACM第41回ソフトウェアエンジニアリング国際会議の議事録、1199~1210。
<https://doi.org/10.1109/ICSE.2019.00122>

— Plösch, R., Dautovic, A., および Saft, M., (2014年)。「The Value of Software Documentation Quality」(英語)、高品質のソフトウェアについての国際会議の議事録、333~342。
<https://doi.org/10.1109/QSIC.2014.22>

— Zhi, J. 他、(2015年)。「Cost benefits and quality of software development documentation: A systematic mapping」(英語)、Journal of Systems and Software, 99(C), 175~198。
<https://doi.org/10.1016/j.jss.2014.09.042>

ドキュメントの品質改善の方法

技術的な作業には、情報の発見と使用が含まれますが、高品質のドキュメントを作成するにはコンテンツの作成と保守を行う人員が必要です。2019年のGoogleの調査では、内部および外部の情報ソースへのアクセスにより生産性を保つことができることが明らかになりました。今年の調査ではさらに一歩進めて、アクセスされたドキュメントの品質や、そのドキュメントの品質に影響するプラクティスについて調べました。

調査から、以下のようなプラクティスがドキュメントの品質に大きな好影響をもたらすことが示されました。

プロダクトやサービスの重要なユースケースを文書化する。 システムについて何を文書化するかが重要で、読者はユースケースによって情報とシステムを有効に活用することができます。

既存のドキュメントの更新と編集について明確なガイドラインを作成する。 ドキュメントに関する作業の多くは、既存のコンテンツの保守です。チームのメンバーが、更新の方法や、不正確または古くなった情報を削除する方法を理解していれば、長期的にシステムが変更されても、チームはドキュメントの品質を維持できます。

オーナーを定義する。 高品質のドキュメントを持つチームは、ドキュメントの所有権も明確に定義している割合が高くなっています。所有権により、新しいコンテンツの作成や、既存のコンテンツの更新と変更内容の検証について、責任の所在を明らかにできます。高品質のドキュメントを持つチームは、取り扱っているアプリケーションのすべての主要な機能についてドキュメントが作成されており、その広範な機能を文書化するために明確な所有権が役立っていると回答する割合が高くなっています。

高品質のドキュメントを保有するチームは

3.8 倍

セキュリティプラクティスを実装する割合

2.4 倍

信頼性目標を満たす、または超える割合

3.5 倍

サイト信頼性エンジニアリング（SRE）プラクティスを実装している割合

2.5 倍

クラウドを完全に活用している割合

ソフトウェア開発プロセスの一部としてドキュメントを含める。

ドキュメントを作成した後で、システム変更と同時にドキュメントの更新を行うチームは、より高品質のドキュメントを持っていると言えます。ドキュメントの作成と保守は、テストと同様に、パフォーマンスの高いソフトウェア開発プロセスに欠かせない部分です。

パフォーマンスレビューとプロモーションのときにドキュメント作業を評価する。 評価は、ドキュメントの総合的な品質と関連します。ドキュメントの作成と保守はソフトウェアエンジニアリング作業の中核部分で、正しく扱うことで品質向上に結び付けられます。

高品質のドキュメントを支える他のリソースとして、次のようなものが判明しました。

- ドキュメントの作成と保守の方法についてのトレーニング
- コードサンプルや不完全なドキュメントの自動テスト
- ドキュメントのスタイルガイドや、グローバルな対象層に向けた作成の手引きなどのガイドライン

ドキュメントは、DevOpsを効果的に機能させるための基礎です。高品質のドキュメントにより、DevOpsのセキュリティ、信頼性など個別の能力への投資からより大きな結果を引き出し、クラウドを十分に活用できるようになります。高品質のドキュメントを支えるプラクティスを実行すれば、技術的な能力の強化と、SDOパフォーマンスの向上として見返りが期待できます。

セキュリティ

[シフトレフト] と全体の統合

技術チームが加速し進化し続ける一方で、セキュリティへの脅威も、数が増加しているとともに複雑性が増しています。2020年には、「Tenable's 2020 Threat Landscape Retrospective Report」（英語）によれば、220億レコードを超える機密個人情報やビジネスデータが漏えいしました⁶。セキュリティは後付けや、提供前の最後のステップではいけません。ソフトウェア開発プロセスの全体を通じて組み込まれている必要があります。

ソフトウェアを安全に提供するには、悪意のある人物によって使用される技法より速く、セキュリティプラクティスが進化する必要があります。2020年のSolarWindsおよびCodecovソフトウェアのサプライチェーン攻撃において、ハッカーはSolarWindsのビルドシステムと、Codecovのbashアップローダースクリプトを侵害し⁷、それらの企業が抱えている数千もの顧客のインフラストラクチャに自分たちのコードを密かに埋め込みました。これらの攻撃の広範な影響を考えれば、業界は予防的な手法から診断的な手法に移行する必要があります。すなわち、ソフトウェアチームは自分たちのシステムがすでに侵害されていると想定し、自分たちのサプライチェーンにセキュリティを構築すべきです。

以前のレポートと同様に、エリートのパフォーマンスのグループはセキュリティプラクティスの使用でも優れていることが判明しました。今年の調査において、エリートのパフォーマンスで信頼性の目標を満たした、または超えたグループは、ソフトウェアの開発プロセスにセキュリティが統合されている割合が2倍でした。ここから、信頼性の標準を維持しながら提供を迅速化したチームは、ソフトウェアを迅速に、高い信頼性で提供する能力を犠牲にすることなく、セキュリティチェックとプラクティスを統合する方法を見つけたことが推測されます。

開発プロセス全体にセキュリティプラクティスを統合したチームは、提供と運用の高いパフォーマンスを示しただけでなく、組織としての目標を満たす、または超える割合も1.6倍高くなっています。セキュリティを採用した開発チームは、ビジネスに多くの価値を生み出しています。

エリートのパフォーマンスで、信頼性の目標を満たした、または超えたグループは、ソフトウェア開発プロセスにセキュリティを統合している割合が2倍になっています。

⁶ <https://www.tenable.com/cyber-exposure/2020-threat-landscape-retrospective>

⁷ <https://www.cybersecuritydive.com/news/codecov-breach-solarwinds-software-supply-chain/598950/>

正しく行う方法

セキュリティの重要性を強調し、チームでセキュリティを優先する必要があると勧告するのは簡単ですが、それを行うには従来型の情報セキュリティ手法にいくつかの変更を加える必要があります。以下のようなプラクティスを活用することで、セキュリティを統合し、ソフトウェアのデリバリーと運用のパフォーマンスを改善して、組織としてのパフォーマンスも改善できます。

テストにセキュリティを含める。自動化されたテストプロセスの一部としてセキュリティ要件をテストします。事前に承認されたコードを使用する必要がある領域もテストの対象になります。

すべてのフェーズにセキュリティレビューを統合する。情報セキュリティ (InfoSec) を、ソフトウェアデリバリーのライフサイクル全体で日常業務に統合します。これには、アプリケーションの設計とアーキテクチャのフェーズでInfoSecチームがアドバイスを提供する、ソフトウェアのデモに参加する、デモ中にフィードバックを行うことが含まれます。

セキュリティのレビュー。主要な機能のすべてについて、セキュリティのレビューを行います。

事前に承認されたコードをビルドする。InfoSecチームが事前承認済みの使いやすいライブラリ、パッケージ、ツールチェーン、プロセスを構築し、開発者やITオペレータが作業で使えるようにします。

InfoSecを早期から頻繁に関与させる。アプリケーション開発の計画の段階、および以後のすべてのフェーズにInfoSecを含め、セキュリティ関連の弱点を早期に見つけられるようにします。これによってチームは、その弱点を修正するため十分な時間を費やせるようになります。

以前に言及したように、高品質のドキュメントによってさまざまな側面において好影響が生まれます。セキュリティも例外ではありません。高品質のドキュメントを持つチームは、開発プロセス全体にセキュリティを統合している割合が3.8倍でした。組織のすべての人々が暗号の専門技術を持っているわけではありません。そうした専門技術は、文書化されたセキュリティプラクティスという形により、組織内で最も効果的に共有されます。

セキュリティプラクティス

セキュリティのテスト	58%
すべてのフェーズにセキュリティレビューを統合	54%
セキュリティのレビュー	60%
事前に承認されたコードのビルド	49%
早期からInfoSecを頻繁に関与	63%

技術面の DevOps 機能

調査から、継続的デリバリーを採用して DevOps 変換を遂行した組織は、高品質、低リスクで、費用対効果の大きいプロセスを保有している割合が高いことが示されました。

具体的には、次の技術的なプラクティスを測定しました。

- 疎結合アーキテクチャ
- トランクベース開発
- 継続的なテスト
- 継続的インテグレーション
- オープンソーステクノロジーの使用
- モニタリングとオペラビリティのプラクティス
- データベースの変更管理
- デプロイの自動化

これらのプラクティスはすべて継続的デリバリーの改善に効果がありますが、疎結合アーキテクチャと継続的なテストが最も大きな影響を持つことが判明しました。たとえば、エリートのパフォーマンスで、自らの信頼性の目標を満了したグループは、低パフォーマンスのグループよりも、疎結合アーキテクチャを採用している割合が3倍も高くなっています。

疎結合アーキテクチャ

この調査ではさらに、サービスとチームとの間で粒度の細かい依存関係を減らすよう作業することで、IT パフォーマンスを改善できることを示します。実際のところ、これは継続的デリバリーの成功を予測するうえで最も強い要素の1つです。疎結合アーキテクチャを使用することで、チームはスケーリング、失敗、テスト、デプロイを互いに独立して行うことができます。チームはそれぞれ独自のペースで手順を進められ、より小さなバッチで作業を行え、技術的な負債が少なくなり、障害から迅速に回復できるようになります。

継続的テストと継続的インテグレーション

以前の年の調査で判明したことと同様に、継続的テストが継続的デリバリーの成功を予測する強い要因であることが示されました。エリートのパフォーマンスで、自らの信頼性の目標を満了したグループは、継続的テストを活用している割合が3.7倍です。デリバリープロセス全体を通して早期かつ頻繁なテストを取り入れ、常にテスト担当者が開発者がともに作業を行うようにすることで、チームはプロダクト、サービス、アプリケーションへの反復処理や変更を迅速に行えるようになります。このフィードバックループを使用し、顧客に価値を提供するとともに、自動テストや継続的インテグレーションなどのプラクティスを簡単に組み入れられるようになります。

継続的インテグレーションにより、継続的デリバリーも改善されます。エリートのパフォーマンスで、自らの信頼性の目標を満了したグループは、継続的インテグレーションを活用している割合が5.8倍です。継続的インテグレーションでは、commit の

たびにソフトウェアのビルドがトリガーされ、一連の自動テストが行われて、数分以内にフィードバックが得られます。継続的インテグレーションによって、正しく統合を行うために必要な手動の、そして複雑になることが多い調整を減らすことができます。

継続的インテグレーションはKent Beckとエクストリームプログラミングコミュニティで生まれたもので、その定義によれば、次に説明するトランクベースの開発のプラクティスが含まれます⁷。

トランクベース開発

調査では、パフォーマンスの高い組織は、トランクベース開発、すなわち開発者が小さなバッチで作業を行い、各自の作業を定期的に共有トランクに結合する方式が使用されている割合が高いことが、一貫して示されています。実際に、エリートのパフォーマンスで、自らの信頼性の目標を満たしたグループは、トランクベース開発を使用している割合が2.3倍です。パフォーマンスの低いグループは、ブランチを長く使用し、結合が遅れる傾向があります。

チームは、作業を最低でも1日に1回、できれば1日に何回も結合するべきです。トランクベース開発は継続的インテグレーションと密接に関連しているため、これら2つの技術的プラクティスを同時に取り入れることが望まれます。一緒に使用されることで、最大の影響を生み出すことができます。

デプロイの自動化

理想的な作業環境では、パソコンが反復作業を行い、人間は問題の解決に集中します。デプロイ自動化の導入により、チームはその目標に近づくことができます。

ソフトウェアをテストから本番環境へ自動的に移行できれば、より高速で効率的なデプロイが可能となり、リードタイムを短縮できます。また、デプロイでエラーが発生する可能性も、手動でのデプロイに比べて減らすことができます。チームでデプロイの自動化を使用すると、フィードバックが直ちに得られ、サービスやプロダクトをはるかに迅速に改善できるようになります。継続的なテスト、継続的インテグレーション、自動デプロイを同時に導入する必要はありませんが、これら3つのプラクティスを一緒に使用すれば、より大きな改善を期待できます。

データベースの チェンジ マネジメント

バージョン管理による変更のトラッキングは、コードの作成と保守、データベース管理の重要な部分です。調査では、エリートのパフォーマンスで、自らの信頼性の目標を満たしたグループは、パフォーマンスの低いグループと比べて、データベースのチェンジ マネジメントを実施している割合が3.4倍です。さらに、データベースのチェンジ マネジメントを正しく行うには、関係するすべてのチーム間での共同作業、連絡、透明性が鍵となります。導入する実際の手法はいくつか選択できますが、データベースに変更を加える必要があるときは必ず、データベースを更新する前にチームが集合して変更内容をレビューすることをおすすめします。

⁷ Beck, K. (2000年)。「Extreme programming explained: Embrace change」(英語)、Addison-Wesley Professional

モニタリングと オブザーバビリティ

これまでの年と同様に、モニタリングとオブザーバビリティのプラクティスが継続的デリバリーの維持に役立っていることが判明しました。エリートのパフォーマンスで、自らの信頼性の目標を満了したグループは、システム全体の健全性にオブザーバビリティを組み入れたソリューションを保有している割合が4.1倍です。オブザーバビリティのプラクティスにより、チームはシステムをよりの確に理解でき、問題点の識別とトラブルシューティングに必要な時間が短縮されます。また調査では、優れたオブザーバビリティのプラクティスを持つチームは、コーディングにより多くの時間を費やしていることも示されています。この発見について可能な説明の1つは、オブザーバビリティのプラクティスを導入することで、開発者が問題の原因を調べるのに必要な時間が減少し、その時間をトラブルシューティングに、最終的にはコーディングに振り分けられるようになったということです。

オープンソース テクノロジー

多くの開発者はすでにオープンソーステクノロジーを活用しており、それらのツールを使い慣れていることが組織にとって長所となります。クローズドソーステクノロジーの主な欠点は、組織と外部との間で知識の移転が制限されることです。たとえば、自組織のツールをすでに使い慣れている人材を雇用することはできず、開発者は獲得した知識を他の組織に移転できません。

これに対して、ほとんどのオープンソーステクノロジーは、それを取り巻くコミュニティが存在し、開発者はそれらをサポートとして利用できます。オープンソーステクノロジーはより広範にアクセス可能で、比較的成本が低く、カスタマイズ可能です。エリートのパフォーマンスで、自らの信頼性の目標を満了したグループは、オープンソーステクノロジーを利用している割合が2.4倍です。DevOps変革を導入するときは、オープンソースソフトウェアの使用を増やすことをおすすめします。

技術的な DevOps 能力の詳細については、
<https://cloud.google.com/devops/capabilities>
で DORA 能力についてご覧ください

COVID-19

今年は、COVID-19（新型コロナウイルス感染症）パンデミックの渦中におけるチームのパフォーマンスに影響した要因を調査しました。具体的には、COVID-19のパンデミックはソフトウェアデリバリーと運用（SDO）のパフォーマンスに悪影響を及ぼしたのか？その結果、チームで燃え尽き症候群が増えたのか？最後に、燃え尽き症候群を減らすにはどのような要素が有望か？という内容です。

まず、パンデミックがデリバリーと運用のパフォーマンスに与えた影響を把握するための調査を行いました。多くの組織では、市場の劇的な変化（たとえば、対面での購入からオンラインへの移行）に対応するためのモダナイゼーションを優先させました。「比較方法」の章では、ソフトウェア業界でのパフォーマンスが大幅に加速し、さらに加速し続けていることについて解説しています。現在では、調査対象のサンプルの大多数がパフォーマンスの高いチームで、エリートのパフォーマンスのグループは引き続き水準を押し上げ、より短いリードタイムで頻繁にデプロイを行い、復旧時間が短く、変更時の障害率は低下しています。同様に、GitHubの調査員による調査でも、2020年を通して開発者のアクティビティ（すなわち、push、pullリクエスト、レビューされたpullリクエスト、およびユーザーごとのコメントされた問題数⁹）の増加が示されています。おそらく業界はパンデミックの結果というよりも、パンデミックにもかかわらず加速し続けていると考えられます。この未曾有な期間においてSDOパフォーマンスの低下傾向がみられなかったのは特筆すべきでしょう。

パンデミックにより人々の勤務形態が変化し、多くの人々にとっては勤務場所も変化しました。このため、パンデミックの結果であるリモート勤務の影響を調べました。調査の結果、回答者の89%はパンデミックのため在宅勤務していることが判明しました。パンデミックの前から在宅勤務の経験があると回答したのは、20%にすぎませんでした。リモート作業環境への移行は、ソフトウェアの開発、ビジネスの運用、共同作業の方法に明確な影響を及ぼしました。多くの人にとって、在宅勤務により廊下で偶然会って会話したり、対面で共同作業したりといったつながりを持つことができなくなりました。

⁹ <https://octoverse.github.com/>

89%

パンデミックのために在宅勤務している回答者の割合

燃え尽き症候群を減らした要因

それにもかかわらず、リモート作業の結果としてチームに燃え尽き症候群が生じたかどうかを大きく左右する要因が見つかりました。それは文化です。ジェネレーティブなチームの文化を持ち、自分がチームに受け入れられており、チームに属していると感じる人々で構成されたチームは、パンデミックの最中に燃え尽き症候群を経験する割合が半分でした。この発見は、チームと文化の優先順位付けの重要性の裏付けとなりました。良い業績を挙げているチームは、チーム全体と個人の両方にプレッシャーをもたらす困難な期間を乗り切る手段を持ち合わせています。

ジェネレーティブなチームの文化を持ち、自分がチームに受け入れられており、チームに属していると感じる人々で構成されたチームは、パンデミックの最中に燃え尽き症候群を経験する割合が半分でした。

文化

一般的な話として、文化はあらゆる組織において避けられない、対人関係の底にある流れです。従業員が組織に対して、および互いについてどのように考え、感じ、行動するかに影響を及ぼす何かです。どの組織にも独自の文化があり、調査の発見事項から、組織とITのパフォーマンスを推進する最も重要な要素の1つが文化であることが一貫して示されています。具体的には、分析から**ジェネレーティブな文化**は、これはWestrumの組織文化の類型論と、人々がその組織に属し、含まれているという感覚によって測定されるものですが、より高いソフトウェアデリバリーと運用（SDO）パフォーマンスを期待できることが示されています。たとえば、エリートのパフォーマンスで、自らの信頼性の目標を満たしたグループは、パフォーマンスの低いグループと比べて、ジェネレーティブなチーム文化を持っている割合が2.9倍であることが判明しました。同様に、ジェネレーティブな文化では組織的なパフォーマンスが高く、従業員の燃え尽き症候群の割合は低いことが予測されます。つまり、文化は非常に重要です。さいわい文化は流動的で、多くの面を持ち、常に変化し続けているため、変えていくことが可能です。

DevOpsを正しく運用するには、組織のチームが共同で、他部門と連携して作業を行う必要があります。Googleは2018年に、パフォーマンスの高いチームは単一の、他部門と連携するチームでソフトウェアを開発してデリバリーする割合が2倍だということを見出しました。これは、あらゆる組織の成功には共同作業と協力が最も重要だということを確認するものです。重要な質問があります。他部門と連携する共同作業を奨励し、褒賞する環境を作り上げるには、どのような要素が関係するのでしょうか？

Googleは何年にもわたり、文化のなりたちを明白にし、文化が組織とITのパフォーマンスに及ぼす影響をDevOpsコミュニティが的確に理解できるようにするべく務めてきました。最初の段階として、Westrumの組織文化の類型論を使用して、文化を運用的に定義しました。同氏は組織について、パワー指向、ルール指向、パフォーマンス指向の3つの類型を識別しました。Googleはこのフレームワークを調査に使用し、パフォーマンス指向で、情報の流れ、信頼、イノベーション、リスク共有を最適化する組織文化が、高いSDOパフォーマンスを期待できることを発見しました。

文化とDevOpsに関する理解が進化するにつれ、Googleは文化の最初の定義を拡大し、**心理的な安全性**など他の心理社会的な要素も含める作業を行いました。パフォーマンスの高い組織は、従業員が悪い結果を恐れることなく、計算された妥当なリスクを冒すことを奨励する文化を持っている可能性が高いものです。

文化は流動的で、多くの面を持ち、常に変化し続けているため、変えていくことが可能です

帰属と受け入れ

文化が常に大きな影響をパフォーマンスに及ぼすことを前提として、今年モデルを拡大し、従業員が組織に属しており、受け入れられているという認識が、文化からパフォーマンスへの有益な影響を生み出すかどうかを調査しました。

心理的な調査から、人々は他者との強く安定した関係を形成し、維持するよう生得的に動機付けされていることが示されました¹⁰。人々は、他者とつながっており、自分が所属しているさまざまなグループに受け入れられていると感じたいと動機付けられています。帰属の感覚は、広範な好ましい身体的および心理的な結果をもたらします。たとえば、帰属の感覚は動機付けに良い影響を与え、学業成績の向上に結び付くことが研究で示されています¹¹。

このつながっているという感覚の根底には、人々が無理なく自分らしさをもって仕事するべきであり、各人の経験や背景が評価され、尊重されると感じるべきだという考えがあります¹²。組織内にインクルーシブで帰属意識がある文化を作り上げることに集中すると、繁栄し、多様性があり、動機付けされた労働力を生み出すために役立ちます。

調査の結果では、パフォーマンス指向で、帰属とインクルーシブネスを重視する組織は、肯定的でない組織文化を持つ組織と比べて、従業員の燃え尽き症候群のレベルが低い可能性が高いことが示されています。

心理社会的な要素がSDOパフォーマンスと、従業員の間での燃え尽き症候群のレベルに影響を及ぼすという証拠から、DevOps改革を成功させるには、改革作業の一部として、文化に関連する問題点の対処に力を注ぐことをおすすめします。

10 Baumeister & Leary, 1995 年。「The need to belong: Desire for interpersonal attachments as a fundamental human motivation」(英語)、Psychological Bulletin, 117 (3)、497~529
<https://doi.org/10.1037/0033-2909.117.3.497>

11 Walton 他、2012 年。「Mere belonging: the power of social connections」(英語)、Journal of Personality and Social Psychology, 102(3):513-32
<https://doi.org/10.1037/a0025731>

12 Mor Barak & Daya, 2014 年。「Managing diversity: Toward a globally inclusive workplace」(英語)、Sage.
Shore, Cleveland, & Sanchez, 2018 年。「Inclusive workplaces: A review and model」(英語)、Human Resources Review <https://doi.org/10.1016/j.hrmr.2017.07.003>

4 章

調査対象者

Accelerate State of DevOps

2021 では、7 年間の調査と、業界専門家からの 32,000 を超える調査回答者を基に、チームと組織が最も成功するようなソフトウェア開発と DevOps のプラクティスを示します。

今年は、全世界から各種の業界に属する1,200 人の現役の専門家が、より高いパフォーマンスを促進する要素についての Google の理解を助けるため、自らの経験を共有してくれました。全体として、さまざまなユーザー属性と企業特性にわたる回答者層は引き続き一貫性の高いものです。



ユーザー属性と企業特性

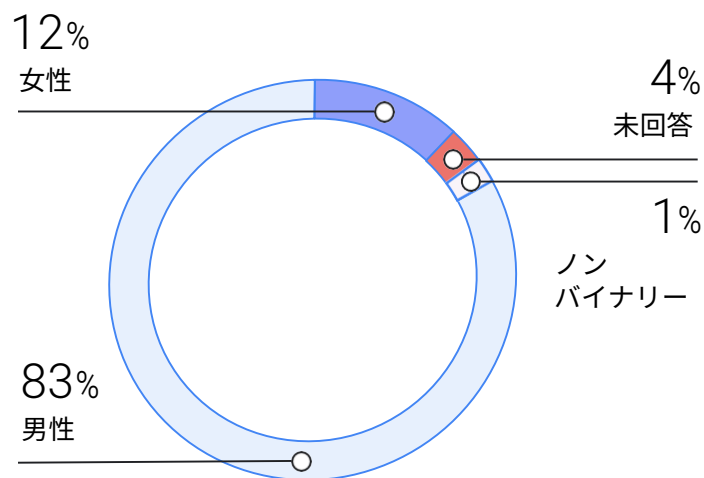
これまでの年と同様に、それぞれの調査回答者について、ユーザー属性の情報を収集しました。カテゴリには、性別、障がい、社会的地位の低いグループがあります。

今年は、会社のサイズ、産業、地域など、企業特性間にわたって、これまでのレポートと一貫した構成比率が見られました。ここでも、回答者の60%以上はエンジニアまたはマネージャーとして勤務しており、1/3はテクノロジー業界に勤務しています。金融サービス、小売、工業/製造業の会社からも回答を頂いています。

ユーザー属性

性別

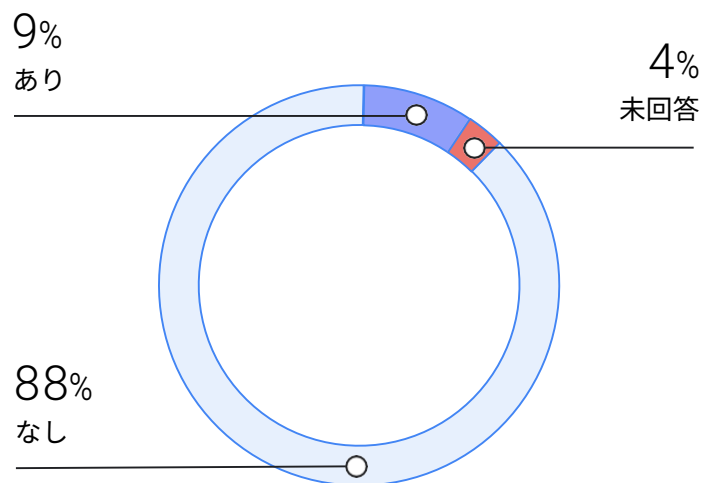
今年のサンプルは以前の調査と同様に、83%が男性、12%が女性、1%がノンバイナリーです。回答者は、チームの約25%が女性だと回答しました。これは、2019年（16%）よりも大きく増えており、2018年の数値（25%）と並ぶものです。



今年の回答者は、チームの25%に女性が含まれていると答えました（中央値）。これは2019年の減少からの回復を示すものです。

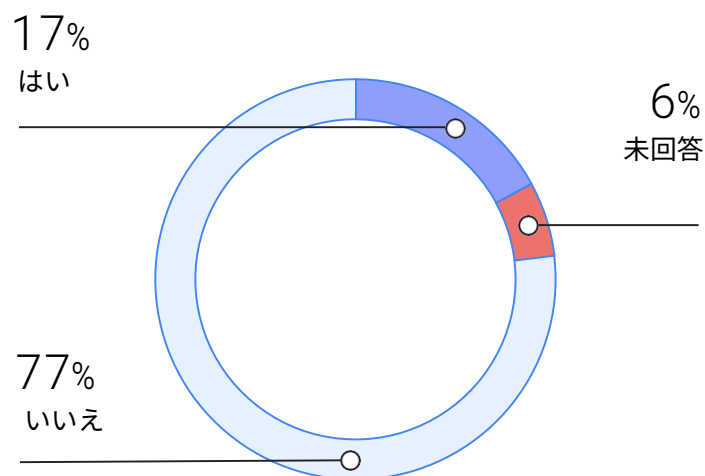
障がい

障がいについては、Washington Group Short Set¹³の基準に従い、6つの観点で識別しました。障がいについての質問を行うのは今年で3年目です。障がいを持つ人員の割合は、2019年のレポートと変わらず9%です。



社会的地位の低いグループ

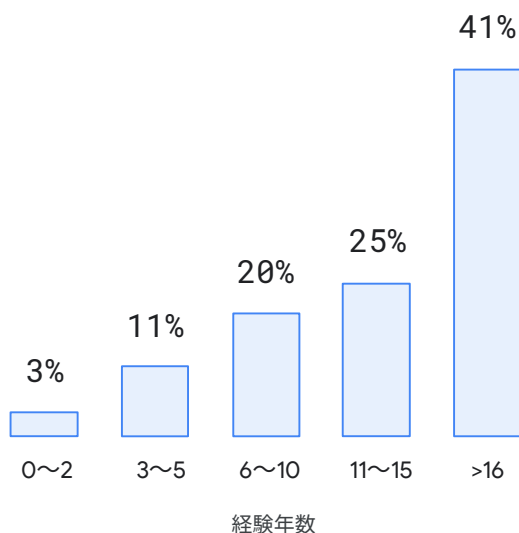
社会的地位の低いグループのメンバーは、人種、性別、その他に関する特性を有しています。社会的地位の低さについての質問を行うのは、今年で4年目です。「社会的地位が低い」として識別される人員の割合は2019年の13.7%から多少増えて、2021年には17%でした。



¹³ <https://www.washingtongroup-disability.com/question-sets/wg-short-set-on-functioning-wg-ss/>

経験年数

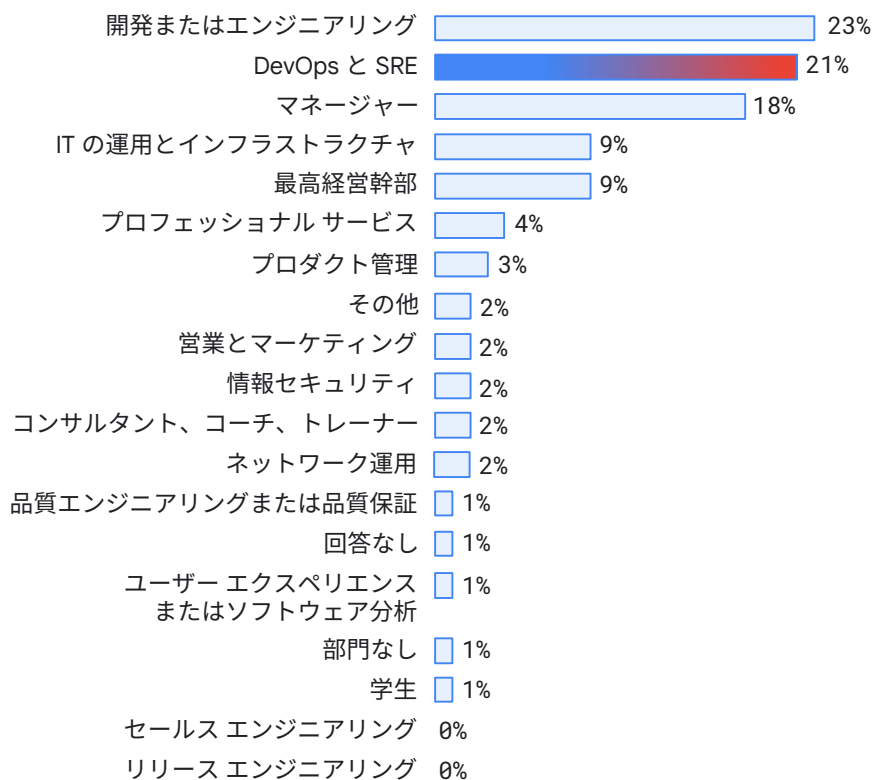
今年の調査の回答者は高度な経験を持ち、41%は16年間以上の経験がありました。回答者の85%以上は少なくとも6年間の経験がありました。



企業特性

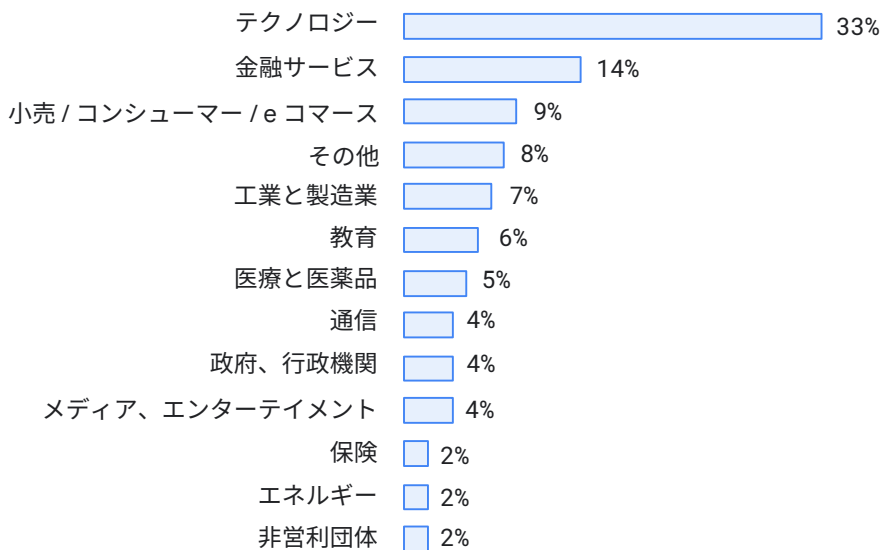
部門

回答者は主に、エンジニアリングチーム（23%）、DevOpsとSREチーム（21%）、マネージャー（18%）、およびITの運用とインフラストラクチャチーム（9%）の業務に携わる個人です。コンサルタントの割合が減少し（2019年の4%から2%に）、最高経営幹部の割合が増えています（2019年の4%から9%に）。



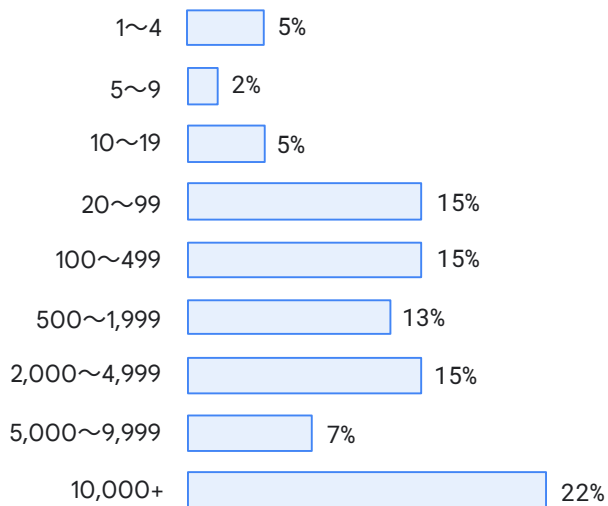
業種

以前の Accelerate State of DevOps レポートと同様に、テクノロジー業界に従事している回答者が最も多く、その後に金融サービス、小売などが続いています。



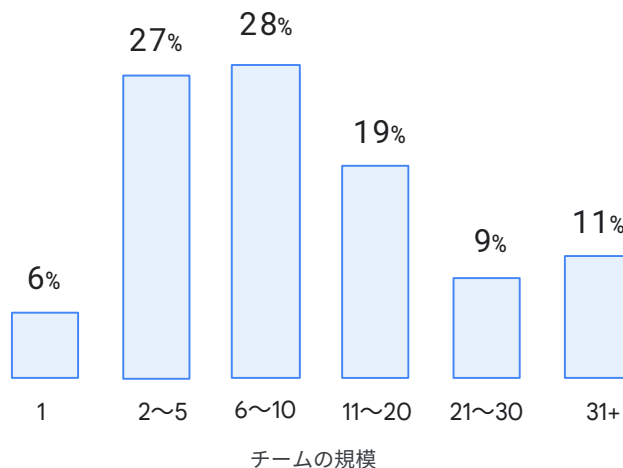
従業員数

以前の Accelerate State of DevOps レポートと同様に、回答者の所属する組織の規模はさまざまです。回答者の22%は従業員数が10,000人を超える企業に勤務しており、7%は従業員数が5,000～9,999人の企業に勤務しています。さらに15%の回答者は、従業員数が2,000～4,999人の組織に属しています。従業員数が500～1,999人の組織が13%、100～499人が15%、最後に20～99人が15%と、さまざまな規模の組織から幅広い回答を得ました。



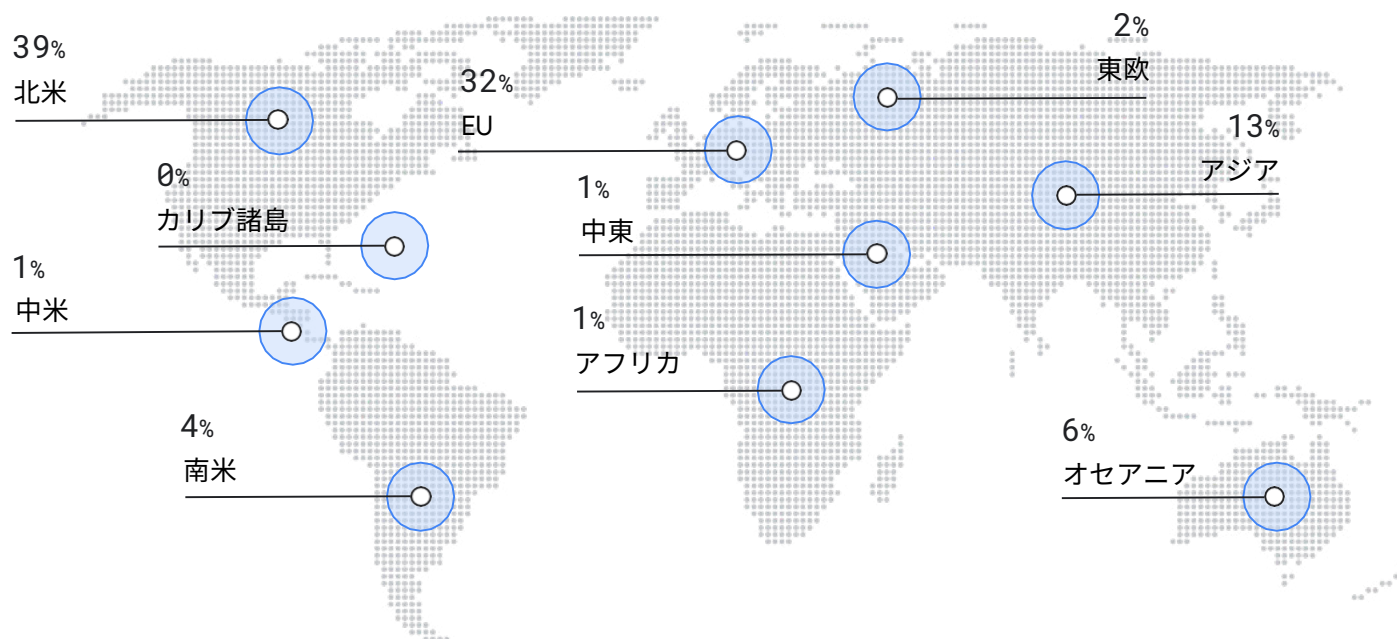
チームの規模

回答者の半数以上（62%）はメンバーが10人以下のチームに属しています（6～10人が28%、2～5人が27%、1人だけのチームが6%）。他の19%は、メンバーが11～20人のチームに属しています。



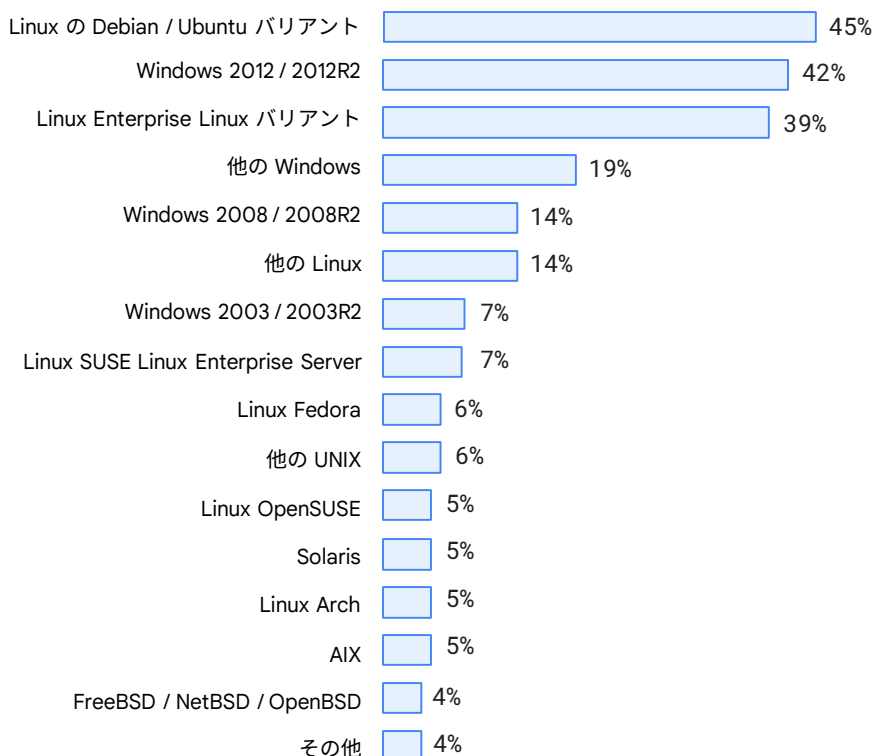
地域

今年の調査では、北米の回答者の割合が減少しました（2019年の50%から2021年には39%に）。その代わりに、欧州（2019年の29%から2021年には32%に）、アジア（2019年の9%から2021年には13%に）、オセアニア（2019年の4%から2021年には6%に）、南米（2019年の2%から2021年には4%に）の回答者の割合が増加しました。



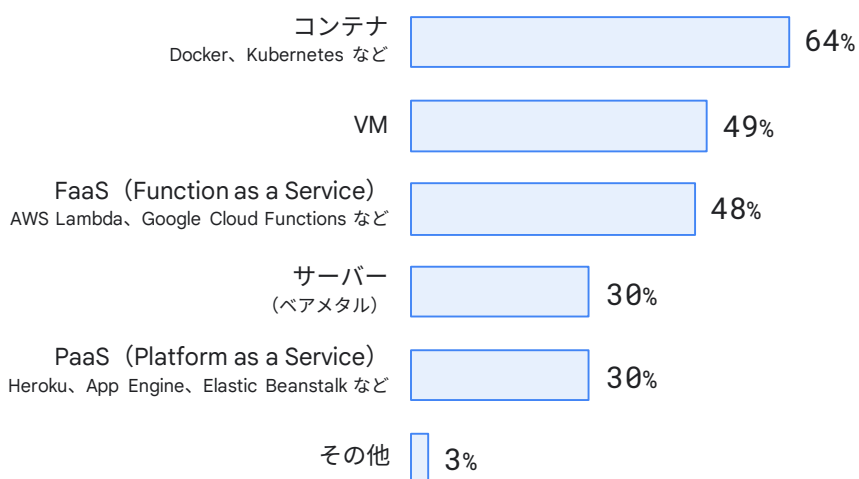
オペレーティング システム

オペレーティングシステムの構成比も、以前のState of DevOpsレポートと一貫しています。Googleのオペレーティングシステムのリストを更新すべきであることを指摘して下さった回答者の皆様に感謝申し上げます。



デプロイ ターゲット

今年は、回答者が取り扱う主なサービスやアプリケーションを、どこにデプロイしているかを調べました。驚くべきことに、回答者の大部分はコンテナを使用しており（64%）、48%は仮想マシン（VM）を使用しています。これは、業界でのより新しいデプロイターゲット技術への移行を反映している可能性があります。企業の規模による相違を調べましたが、デプロイターゲット間で有意な差異は見られませんでした。

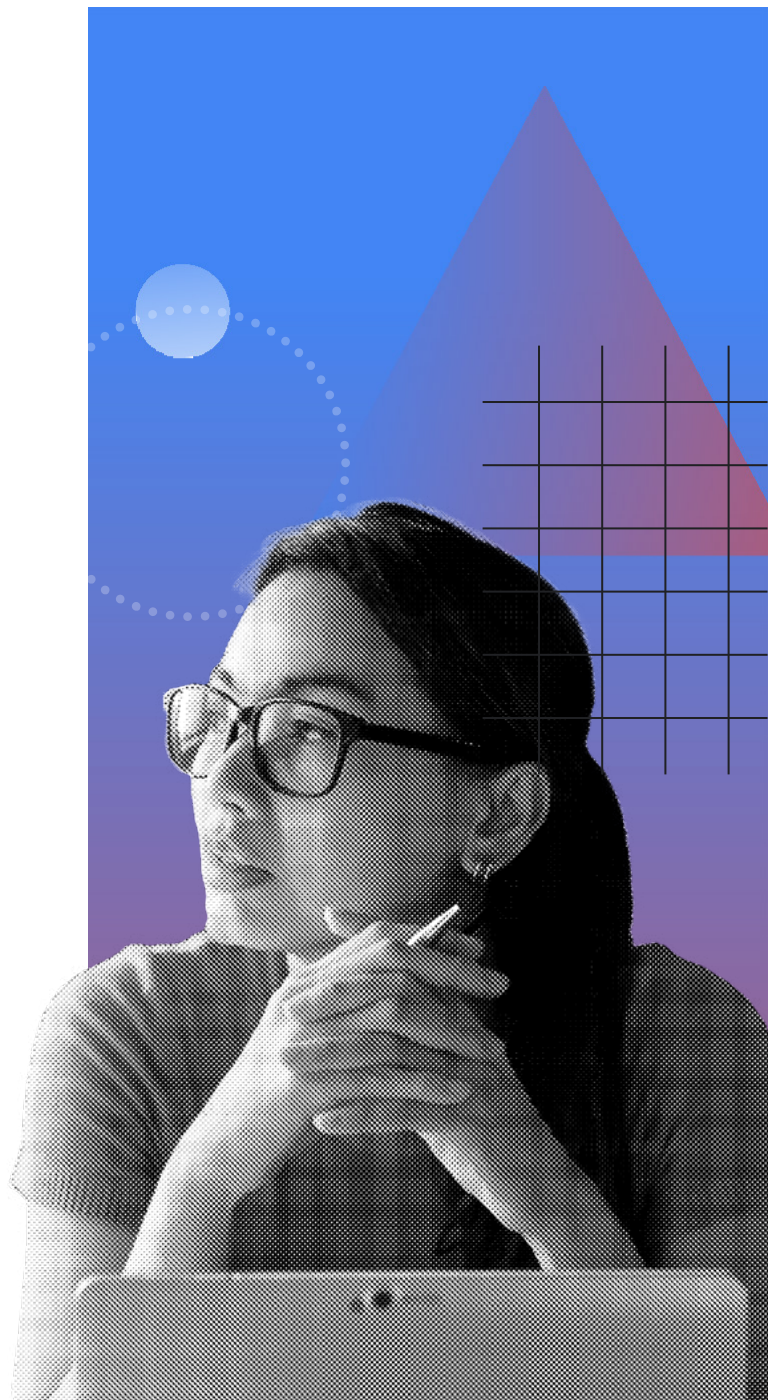


5 章 最後に

7年間にわたる調査から、DevOpsが組織にもたらす利益は依然として明白です。組織は年々加速し、改善され続けています。

DevOpsの原則と能力を取り入れたチームは、ソフトウェアを迅速に、高い信頼性で提供でき、同時にビジネスへと価値を直接推進できます。今年も、SREプラクティス、安全なソフトウェアのサプライチェーン、高品質のドキュメントの影響を調査し、クラウドの活用の探求も再び調査しました。それぞれの分野で、人員とチームはより効果を上げることが可能です。ソリューションの構造に人員を適応させるのではなく、ソリューションの能力を利用する人員に適した構造を作り上げることの重要性に重点を置きました。

今年の調査に参加してくださったすべての方々に感謝申し上げますとともに、この調査が皆様とその組織において、より良いチームとより良いソフトウェアを構築し、同時にワークライフバランスを維持するため役立つことを望んでおります。



6 章

謝辞

今年のレポートは、多数の貢献者の熱意により実現しました。このレポートは、調査の質問設定、分析、著述、編集、レポートのデザインなどの作業の担当者なしには完成できませんでした。著者一同は、こうした作業協力者から今年のレポートのために頂いた協力と教示に感謝します。以下の名前はアルファベット順です。

Pali Bhat

Maria Bledsoe

James Brookbank

Jan Bultmann

Lolly Chessie

John Day

Rakesh Dhoopar

Siobhán Doyle

Alex Eldemir

Nicole Forsgren

Aaron Gillies

Kelsey Hightower

Jez Humble

David Huh

Vic Iglesias

Harish Jayakumar

Nikhil Kaul

Lital Levy

Amanda Lewis

Ríona MacNamara

Andrew Macvean

Steve McGhee

Erin McKean

Jacinda Mein

Eric Maxwell

Raghu Nandan

Claire Peters

Garrett Plasky

John Ryan

Vinay Srinivasan

Christina Storm

Oren Teich

Finn Toner

Marcin Treder

Seth Vargo

Salim Virji

Brenna Washington

Michael Winser

Julia Yager-Reisen

7 章

作成者



Dustin Smith

Dustin Smith はヒューマンファクターズ心理学者で、Google のユーザー エクスペリエンス調査マネージャーを務めており、3年間にわたって DORA プロジェクトに従事してきました。過去7年間にわたって、人々が周囲のシステムや環境によってどのような影響を受けるかを、ソフトウェアエンジニアリング、無料プレイのゲーム、医療、軍事といったさまざまな分野で調査してきました。Google における彼の調査から、ソフトウェア開発者が開発をより快適に行い、生産性を高められる分野が識別されました。彼は2年間にわたって DORA プロジェクトに従事しました。Dustin はウィチタ州立大学から、ヒューマンファクターズ心理学で博士号を授与されています。



Daniella Villalba

Daniella Villalba は、DORA プロジェクト専任のユーザーエクスペリエンス調査員です。開発者が快適に作業を行い、生産性を高めるための要素を理解するため注力しています。彼女は Google に入社する前、瞑想トレーニングの利点、大学生の体験に影響を及ぼす心理社会的な要素、目撃記憶、虚偽の自白について研究してきました。Daniella はフロリダ国際大学から、実験心理学で博士号を授与されています。



Michelle Irvine

Michelle Irvine は Google のテクニカルライターで、デベロッパー ツールと、それを使用する人々との橋渡しをする作業を行っています。Google に入社する前は教育系出版社で勤務経験があり、また、物理シミュレーションソフトウェアのテクニカルライター職に従事していました。Michelle はウォータールー大学から、物理学の学士号と、レトリックおよびコミュニケーション設計の修士号を授与されています。



Dave Stanke

Dave Stanke は Google のデベロッパー リレーション エンジニアで、DevOps と SRE を採用するためのプラクティスについてお客様にアドバイスしています。職歴を通して、スタートアップの CTO、プロダクト マネージャー、カスタマー サポート、ソフトウェア開発者、システム管理者、グラフィック デザイナーなど、あらゆる職務を経験してきました。Dave はコロンビア大学からテクノロジー管理の修士号を授与されています。



Nathen Harvey

Nathen Harvey は Google のデベロッパー リレーション エンジニアで、チームがその可能性を実現し、同時にテクノロジーとビジネスの成果を同期できるように支援する作業を行ってきました。最良のチームやオープンソースコミュニティのいくつかと一緒に作業する機会を得て、それらが DevOps と SRE の原則とプラクティスを適用するための支援を行いました。Nathen は O'Reilly 2020 の「97 Things Every Cloud Engineer Should Know」(英語)を共同で編集し、寄稿しました。

8 章 手法

調査方法の設計

この調査では、セクション間にまたがる、理論に基づいた設計を採用しました。理論に基づいた設計は推論予測と呼ばれ、今日行われるビジネスやテクノロジーの調査のうち、最も一般的なタイプの1つです。純粋に実験的な設計が不可能で、現場での試験が好まれるときは、推論設計が使用されます。

対象層とサンプリング

この調査における対象層は、テクノロジーと変換の内部、またはその近くで作業を行っている実務担当者とリーダー、特にDevOpsに馴染んでいる人々です。調査はメーリングリスト、オンラインプロモーション、オンラインパネルでプロモートされ、対象の人々に対して、各自のネットワークでアンケートを共有するように求めました（雪だるま式サンプリングです）。

潜在的な構成の作成

可能な場合は、以前に検証済みの構成を使用して、仮説と構成を定式化しました。新しい構成は理論、定義、専門家の意見に基づいて作成しました。その後で、意図を明確にし、アンケートによって収集されるデータの妥当性と有効性が高くなるよう、追加手順を行いました¹⁴。

統計分析の方法

クラスタ分析。 ソフトウェアデリバリーパフォーマンスのプロファイルを、デプロイの頻度、リードタイム、サービス復旧時間、変更時の障害率に基づいて識別するため、クラスタ分析を使用しました。潜在クラス分析¹⁵を使用しました。これは、あらかじめクラスタの数を決定する業界的または理論的な理由がなく、クラスタの最適な数を決定するためにベイズ情報量基準¹⁶を使用したためです。

測定モデル。 分析を行う前に、探索的因子分析と、バリマックス回転を使用する主成分分析を使用して構成を識別しました¹⁷。抽出された平均分散（AVE）、相関、クロンバックのアルファ¹⁸、複合信頼性を使用して、収束の妥当性および発散的妥当性と信頼性の統計テストを確認しました。

構造方程式のモデリング。 構造方程式のモデリング（SEM）は部分的最小二乗（PLS）分析を使用してテストしました。これは、相関に基づくSEMです¹⁹。

14 Churchill Jr, G. A. 「A paradigm for developing better measures of marketing constructs (英語)」 Journal of Marketing Research 16:1、(1979年)、64~73。

15 Hagenaars, J. A. および McCutcheon, A. L. (編集) (2002年)。「Applied latent class analysis (英語)」、Cambridge University Press

16 Vrieze, S. I. (2012年)。「Model selection and psychological theory: a discussion of the differences between the Akaike information criterion (AIC) and the Bayesian information criterion (BIC)」 (英語)、Psychological methods、17 (2)、228。

17 Straub, D., Boudreau, M. C., および Gefen, D. (2004年)。「Validation guidelines for IS positivist research」 (英語)、Communications of the Association for Information systems、13 (1)、24。

18 Nunnally, J. C., 「Psychometric Theory」 (英語)、New York: McGraw-Hill、1978年。

19 Hair Jr, J. F., Hult, G. T. M., Ringle, C. M., および Sarstedt, M. (2021年)。

「A primer on partial least squares structural equation modeling (PLS-SEM)」 (英語)、Sage publications。

9 章

関連情報

DevOps の能力の詳細について:

<https://cloud.google.com/devops/capabilities>

サイト信頼性エンジニアリング (SRE) のリソース:

<https://sre.google>

DevOps のクイック チェック:

<https://www.devops-research.com/quickcheck.html>

DevOps 調査プログラムについて調べる:

<https://www.devops-research.com/research.html>

Google Cloud Application Modernization Program について調べる:

<https://cloud.google.com/camp>

「DevOps 変革の ROI: モダナイゼーション イニシアチブの効果を
数値化する方法」 ホワイトペーパー:

<https://cloud.google.com/resources/roi-of-devops-transformation-whitepaper>

過去の State of DevOps レポート:

State of DevOps 2014 (英語) : <https://services.google.com/fh/files/misc/state-of-devops-2014.pdf>

State of DevOps 2015 (英語) : <https://services.google.com/fh/files/misc/state-of-devops-2015.pdf>

State of DevOps 2016 (英語) : <https://services.google.com/fh/files/misc/state-of-devops-2016.pdf>

State of DevOps 2017 (英語) : <https://services.google.com/fh/files/misc/state-of-devops-2017.pdf>

Accelerate State of DevOps 2018 (英語) : <https://services.google.com/fh/files/misc/state-of-devops-2018.pdf>

Accelerate State of DevOps 2019 (英語) : <https://services.google.com/fh/files/misc/state-of-devops-2019.pdf>