



eBook

今こそはじめる

DevOps入門ガイド

属人化を脱却し、スピードと品質を可能にする仕組み

入門ガイドで得られるポイント

- ・ DevOpsの基本原則とメリット
- ・ 自社のSalesforceにおけるDevOpsの適正判断
- ・ DevOpsの成果測定方法と成功への道筋



はじめに

Salesforce開発や運用の現場では、属人化や非効率なプロセスにより、リリースの遅れや夜間・週末対応が常態化してしまうケースが少なくありません。こうした「現場の課題」は、結果的に経営スピードを鈍化させ、投資の成果を十分に引き出せないリスクにつながります。

その解決策のひとつが DevOps です。DevOpsは単なる新しい技術ではなく、開発と運用のプロセスを最適化し、自動化によって効率とスピードを高めることで、Salesforce投資を確実に事業成果へとつなげるための仕組みとなる基盤です。変化の激しい今、そしてAIが加速する時代に求められるのは、ツール導入そのものではなく、Time to Value(価値提供までの時間)を短縮し、持続的に成果を生み出す体制づくりです。

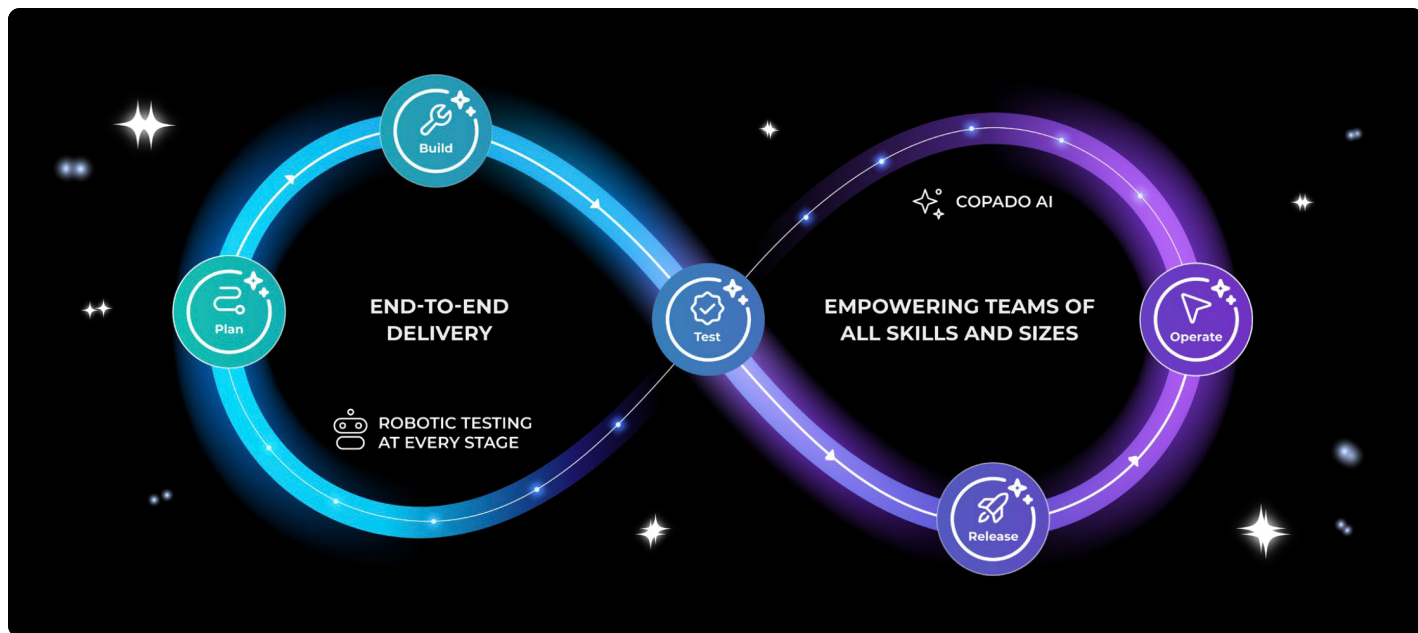
こんな方におすすめ

- ・ Salesforce開発・運用担当者: 属人化や手作業の多さに悩み、リリース対応が負担になっている方
- ・ DX推進・情報システム部門リーダー: スピードとガバナンスを両立した開発・運用体制を実現したい方
- ・ 経営企画・事業部門のマネジメント層: Salesforceへの投資を確実に事業成果につなげたい方

このeBookで得られること

- ・ DevOpsの基本原則と世界中のチームが取り入れている理由
- ・ 自社のSalesforce体制にDevOpsが適しているかを判断する視点
- ・ 成功に向けた成果測定の方法
- ・ 実践を改善し、ビジネスと技術の両面で成果を高めるヒント

この入門ガイドを通じて、あなたのSalesforceチームが「属人化から脱却し、AI時代に成果を生み出す体制づくり」を始めるきっかけを見つけてください。



DevOpsの約束 — なぜ取り入れるべきなのか？

DevOps（デブオプス）は、「Development（開発）」と「Operations（運用）」を組み合わせた概念で、ソフトウェア開発とIT運用を統合・自動化するアプローチです。開発工程における重要な要素を包括し、開発期間の短縮とライフサイクル全体の改善を実現します。

DevOpsの6つの主要な約束

ソフトウェア提供の加速

DevOpsは、アプリケーションライフサイクル管理（ALM）を効率化し、新機能・アップデート・修正をより迅速に提供することを目指します。

テスト、統合、デプロイなどのプロセスを自動化することで、開発サイクルを短縮し、市場ニーズへの迅速な対応を可能にします。

この考え方はアジャイル開発の目的とも密接に関係しており、多くのDevOpsチームが何らかの形でアジャイルを採用しているのも不思議ではありません（両者の違いについては次の章で詳しく解説します）。

ソフトウェア品質の向上

DevOpsにおける継続的インテグレーション（CI）と継続的デリバリー（CD）の実践は、コードを高頻度でテストし統合することを可能にします。これによりバグの削減、コード品質

の向上、そして堅牢で信頼性の高いソフトウェアリリースが実現します。

コラボレーションとコミュニケーションの強化

DevOpsは、開発、運用、ビジネス各部門間の従来のサイロ（縦割り構造）を解消します。チーム間での責任共有とオーナーシップ文化を促進し、部門横断的なコラボレーションを向上させ、誤解や情報の行き違いを減らします。

リソース活用の最適化

自動化と効率的なプロセスにより、手作業を減らし、時間とリソースを節約します。これにより、チームは繰り返し発生しやすくミスの多い作業ではなく、イノベーションや価値創出につながる業務に集中できるようになります。

セキュリティとコンプライアンスの強化

DevOpsは、開発パイプラインにセキュリティ対策を組み込みます。このアプローチは「DevSecOps」と呼ばれます。自動化されたセキュリティチェックと継続的なコンプライアンス監視により、セキュリティを後付けではなくプロセスの中核に据え、コンプライアンスの確保と強化を実現します。



より優れた顧客体験の提供

DevOpsは、提供スピードの向上、信頼性の維持、ソフトウェア品質の改善を通じて、ユーザー満足度に直接的な影響を与えます。顧客からのフィードバックへの迅速な対応やバグ修正のスピード向上により、全体的な体験価値が高まります。

理想的に聞こえますが、これらのDevOps目標を達成するためには、成熟したDevOpsの実践と統合されたツール環境の導入が不可欠です。ここからは、Salesforce導入やチーム運営におけるDevOpsの価値を説明する前に、アジャイルとDevOpsの違いを簡単に振り返ります。この振り返りによって、DevOpsの実践方法、適用範囲、文化的側面についての理解をさらに深められるでしょう。

アジャイルとDevOps：主な違い

アジャイルとDevOpsは密接に関連した手法ですが、焦点には若干の違いがあります。

アジャイルは、短期間かつ反復的なソフトウェア開発ライフサイクル（SDLC）を構築し、ビジネス要件への迅速な対応や、小さな価値を段階的に提供することでリスクを低減することに重点を置きます。主にソフトウェア開発の計画面に関するアプローチです。

一方、DevOpsはSDLCを自動化し、人間が価値創出に集中できるようにし、ヒューマンエラーを減らすことを目的としています。

側面	アジャイル	DEVOPS
焦点と範囲	主にソフトウェア開発の計画と適応性に焦点を当て、作業を小さな単位やスプリントに分割して進める	開発チームと運用チームの双方にまたがるプロセス最適化を、自動化ツール活用と文化面の変革を通じて重視する
進化	長期的な計画に偏り、非効率となっていたソフトウェア開発管理への反省から生まれた手法	アジャイルから発展し、特定の技術的プラクティスを推進するとともに、開発チームと運用チームの間にあるサイロ（分断）を解消する
実践	スプリント計画、ユーザーストーリー、反復型開発を中心に展開する	CALMSフレームワーク（Culture：文化、Automation：自動化、Lean：リーン、Measurement：計測、Sharing：共有）に集約される
導入	スクラムなどのフレームワークを通じて実装され、計画面に重点を置く	継続的デリバリー、自動化テスト、デプロイ自動化、モニタリングなどを含む
文化	適応性と変化への迅速な対応を重視する	開発チームと運用チーム間で、責任共有、透明性、迅速なフィードバックを重視する文化を構築する

ご想像のとおり、アジャイルとDevOpsは相互排他的ではありません。実際、多くのチームが両者を組み合わせて成功を収めています。

これでDevOpsの利点と特徴が明確になったところで、次はDevOpsが貴社のSalesforce導入やチームにとって適切なアプローチかどうかを見ていきましょう。

DevOpsはあなたのチームにとって最適な選択肢か？

すべての組織が、包括的なDevOps導入による恩恵を最大限に享受できるわけではありません。では、どのようにしてDevOpsが自社に適したアプローチかを判断すればよいのでしょうか。

この問いに答えるためには、以下の2つの要素を考慮する必要があります。

- ・ Salesforce導入の複雑さ
- ・ チームが正式なプロセスに従うことへの許容度

Salesforce導入の複雑さ

Salesforceの導入は複雑ですか？Salesforceの導入の複雑さを評価する際に、正しい方向性を示すいくつかの質問をご紹介します。

同じ建物で勤務する開発者が5名未満のチームであれば、正式なプロセスがなくても比較的容易に協力し合うことが可能でしょう。しかし、チーム規模が10名程度に増えると、たとえ同じ場所にいても変更内容の調整は難しくなります。さらに、チームが複数のタイムゾーンに分散している場合、リアルタイムでのコラボレーションは困難となり、正式なプロセスの導入がチームの効率に大きく寄与します。



チームには何人の開発者が在籍していますか？

すべての開発者は同じ建物または同じタイムゾーンに所在していますか？

チームメンバーが退職し、新メンバーが加入する頻度はどのくらいですか？

導入内容の詳細

- ・ 本番組織は幾つありますか？
- ・ Salesforceのクラウドは幾つ使用していますか？
- ・ ISVアプリケーションは幾つ使用していますか？
- ・ 組織には複数の事業部門や事業地域がありますか？
- ・ Salesforceは他に幾つのシステムと統合されていますか？
- ・ 貴社は医療や金融サービスなどの規制のある業界に属していますか？

組織が複数のSalesforce本番環境を管理している場合、またはSales CloudやService Cloudだけでなく、複数のクラウドやISVアプリを利用している場合、複雑性は急速に高まります。

本番環境が2つのシンプルな構成（いわゆるバニラ環境）であれば大きな問題にはなりませんが、これにMarketing Cloud、Revenue Lifecycle Management、Industry Cloud、さらにAppExchangeからの複数アプリが加われば、小規模チームでは管理しきれないほど多くの要素が動くことになります。

たとえ本番環境が1つでも、複数の事業部門（BU）や複数の地域をサポートしていれば、複雑性は必然的に増します。BUが異なれば、プロセスや取り扱う製品も微妙に異なります。地域が異なれば、多言語対応や多通貨対応、さらには異なる規制やルールが必要となり、それぞれに異なるプロセスが生じます。

かつてSalesforceは独立したシステムでしたが、現在ではビジネスの中核となる傾向が強まっています。Customer 360の概念が示すように、最低でも会計システムとの連携が必要になるでしょう。加えて、地図サービスなど、APIを介したその他の外部連携も存在するかもしれません。

いずれの場合も、Salesforce内での変更がこれらの連携を破壊する可能性があり、そのためにはより信頼性の高いプロセスが求められます。

チームが正式なプロセスに従う能力

スタートアップ企業は、多くの場合、正式なプロセスよりもスピードを優先します。「プロセスに時間を割く余裕はない、迅速に行動しなければならない!」という考え方が、こうした環境では一般的です。しかし、以下の点からわかるように、長期的に持続可能なスピードを実現するには、時にはスピードを落とすことも不可欠です。成熟した DevOps の実践を採用することで、企業は最終的に、適応力を維持しながら迅速な行動が可能になります。

対照的に、確立された企業は、より構造化されたプロセスを採用する傾向があります。組織内の個々のチームは小規模であっても、より大きな企業フレームワークで要求される正式な手続きに従わなければならない場合があります。

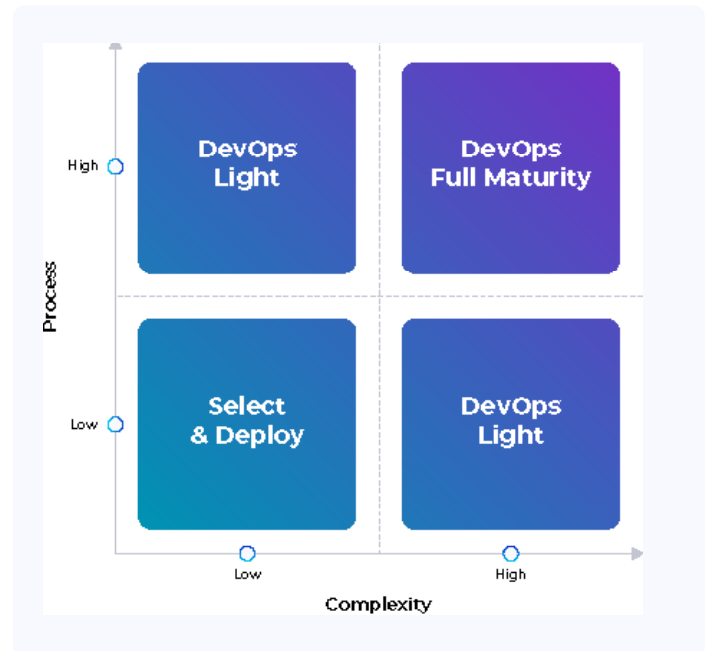
小規模チームには DevOps が必要か？

1 つの本番環境、1 つの地域にある 1 つの BU、そして厳格なプロセスを嫌う小さなチームがあるとします。正式な DevOps プロセスに代わる選択肢は何でしょうか？

このような状況にあるほとんどの企業は、本番環境で直接変更を行うか、変更セットを使用してサンドボックスから変更を移行しています。本番環境での開発は危険であり、ほとんどのチームは一度その教訓を学べば、二度と過ちを犯すことはありません。

しかし、変更セットに代わる「選択してデプロイ」という方法があります。このソリューションにより、小規模なチームでも、サンドボックスで行った変更をより確実に選択して本番環境にデプロイすることができます。

このようなソリューションの例としては、チームのデプロイの迅速化、リリースの簡素化、コラボレーションの促進に役立つ Copado Essentials があります。また、ユーザーはテストの自動化機能や、変更をユーザーストーリーにリンクして、同じ環境で複数の開発者が行った作業を追跡する機能も利用できます。



チームの未来について考えてみましょう。

多くの小規模企業は、現在の課題に対処するために Salesforce を小規模に運用しており、その成果に満足しています。大企業も、カスタマイズや定期的な変更を行わず、ほぼそのままの状態で Salesforce を使用しています。このような場合、Select and Deploy ソリューションは、今後数年間、お客様のビジネスに最適な選択肢となるでしょう。

今後数年間で現在の状況に変化が見込まれる場合、会社の成長に合わせて Salesforce の導入をシームレスに拡張できる強固な基盤を構築することが理にかなっています。

ロードマップを理解することで、ビジネスニーズをサポートする次のプロセスを計画しやすくなります。上の図では、現在の複雑さのレベルやプロセスに対する許容度が低い場合は、今すぐ DevOps に投資し、ニーズに合わせてプロセスの規模を適正化すべきです。

DevOps指標 — 成果の見える化と評価方法

チームは、DevOps Research and Assessment (DORA) チームによって開発された DORA メトリクスを使用して、DevOps におけるプロセスを測定するための標準的なメトリクスを使用しています。これらのメトリクスは、ソフトウェアのデリバリー効率と運用信頼性を測定する一連のパフォーマンス指標です。これらは、DevOps の実践とプロセスの有効性を評価するための業界標準となっています。

DevOpsの進捗を測定するための4つのDORA指標

デプロイ頻度 (Deployment Frequency, DF)

1

組織が新しいソフトウェアをリリースする頻度を示す指標です。高パフォーマンスのチームは、小さな変更を頻繁にデプロイし、1日に複数回から週1回程度のペースで実施します。これにより、迅速な提供サイクルを実現していることを示します。

変更のリードタイム (Lead Time for Changes, LT)

2

コミットされたコード変更が本番環境に到達するまでにかかる時間を示します。短いリードタイムは、ワークフローの効率性が高く、ボトルネックが最小限であることを意味します。

平均復旧時間 (Mean Time to Restore, MTTR)

3

本番環境で障害が発生してから復旧するまでの平均時間を示します。MTTRが低いほど、インシデント対応力と運用の回復力が強いことを示します。

変更失敗率 (Change Failure Rate, CFR)

4

本番環境で障害を引き起こす変更の割合を示します。失敗率が低いほど、開発・テストプロセスの品質が高いことを意味します。

DORA メトリクスのメリット



継続的改善の推進

DORA メトリクスは、DevOps パイプラインのボトルネック、非効率性、弱点を特定するためのデータ駆動型のアプローチを組織に提供します。これらのメトリクスを分析することで、チームは情報に基づいた意思決定を行い、プロセスの改善、より優れたツールの採用、開発チームと運用チーム間のコラボレーションの強化を図ることができます。

たとえば、デプロイメント頻度が低い場合は、手作業によるプロセスが多すぎる、または優先順位が適切でないことを示している可能性があります。同様に、変更失敗率が高い場合は、テスト戦略やデプロイメント手順の見直しが必要かもしれません。

オープンで責任を分かち合う組織づくり

標準化されたベンチマークに対してパフォーマンスを測定することで、組織は問題が発生した際に責任のなすり合いではなく協力を促進できます。MTTR（平均復旧時間）のようなメトリクスは、開発チームと運用チームの両方を巻き込むため、インシデントとその解決に対する共同責任を促進します。

チームパフォーマンスを評価する共通フレームワークの提供

チームは、公開されている業界ベンチマークに基づき、自らを低・中・高・エリートいずれかのパフォーマンス層に分類できます。これにより、改善へのモチベーションが高まるだけでなく、時間の経過とともに進捗を追跡し、現実的かつ段階的な目標を設定することが可能になります。

これらの指標を継続的にモニタリングし改善することで、チームは受動的な対応型の作業から、プロアクティブ（先回り型）かつ自動化され、スケーラブルなワークフローへと進化できます。

DORAメトリクスは、DevOpsで優れた成果を目指す組織にとって非常に強力なツールです。これらはDevOpsパフォーマンスを客観的に測定する基準であり、継続的改善を可能にし、技術的目標とビジネス目標の整合性を高め、パフォーマンス評価と最適化のための標準化されたフレームワークを提供します。

これらの指標にフォーカスすることで、チームはより迅速な提供、高い信頼性、そして急速に変化するデジタル市場での競争優位性を獲得できます。



補足: DORAメトリクスはDevOpsに特化した指標であり、デリバリー頻度や、コードのコミットからリリースまでのリードタイムといったTime to Valueを測定します。

一方で、コードを書くのに要した時間や、ビジネス要件が作成されてからその要件を実装するコードをコミットするまでの時間は測定しません。

これらの指標は、より包括的なバリューストリームマッピング（Value Stream Mapping, VSM）と呼ばれる手法の一部として組み込むことができますが、本書では範囲外としています。

結論

DevOps は、ソフトウェアのデリバリーと運用効率を向上させるための堅牢なフレームワークを提供します。DevOps は、コラボレーション、自動化、継続的な改善をプロモートすることで、チームのデプロイメントの迅速化、ソフトウェアの品質向上、リソースの利用最適化を実現します。

シンプルなニーズを持つ小規模チームでも、複雑なSalesforceの実装を行う成長中の組織でも、DevOps は、効果的に規模を拡大し、競争力を維持するための構造とツールを提供します。その原則を理解し、DORA などの主要な指標で進捗状況を測定し、プロセスをビジネス目標に整合させることで、DevOps は持続的な成功の推進に貢献します。

DevOpsプロセスを今すぐ自動化および拡張しましょう。

ミーティングで相談(無料)

Copado(コパード)について

コパードは、企業がSalesforceでの構築とデプロイメントの方法を変革する、エンタープライズグレードの DevOps ソリューションを提供しています。開発ライフサイクル全体に AI による自動化を統合することで、コパードは、計画から自動テスト、本番デプロイメントに至るまでのプロセス全体を効率化します。5,000 万件以上のデプロイメントを処理し、1,400 以上のグローバル企業をサポートしてきた実績を持つコパードは、組織がSalesforceへの投資を最大限に活用する方法を変革しています。

著者について



David Brooks

エバンジェリズム担当
上級副社長 コパード

シリコンバレーで35年の経験を持つ David は、2004 年にサンドボックスの最初のバージョンをベータテストしました。

コパード入社以前は、Salesforceの製品管理担当副社長として、AppExchange の立ち上げと Force.com プラットフォームの構築を担当しました。

2018 年にコパードに入社し、同社の製品管理業務を構築し、Salesforceの DevOps に革命をもたらしました。

