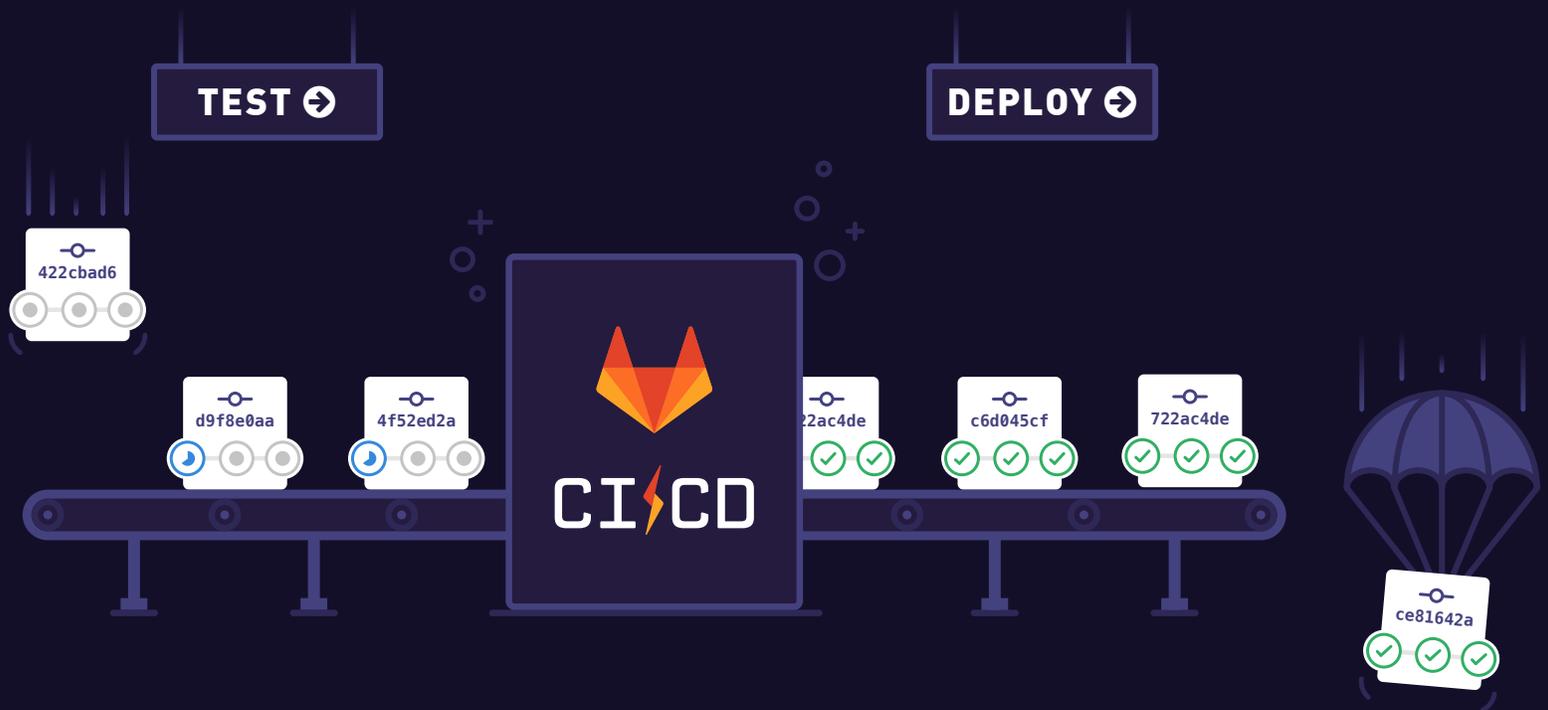
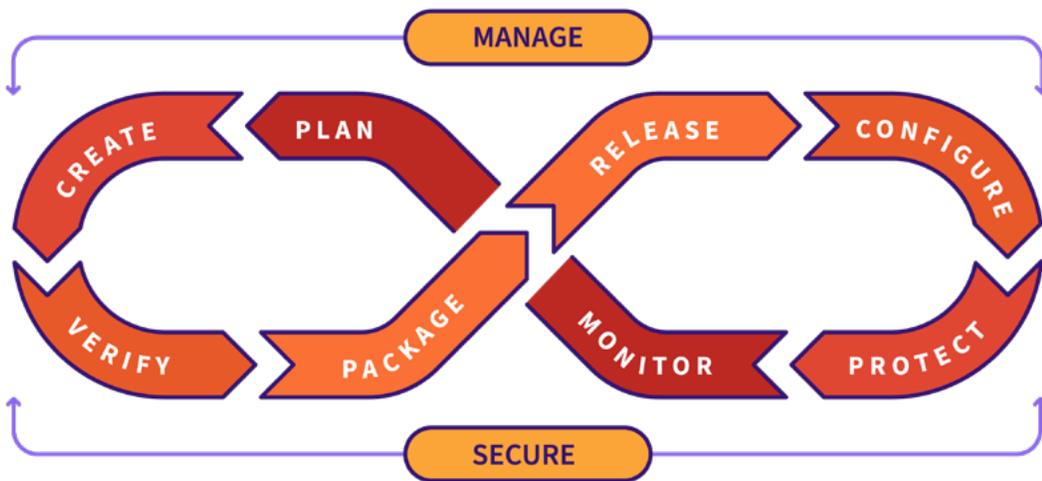


# シングルアプリケーション CI/CD の価値



継続的インテグレーションとデリバリー (CI/CD) により、ソフトウェアを構築し、テストし、デプロイする方法が変わりました。CI/CD ツールはこれらのプロセスを自動化することで、エラーが起こる確率を減らし、ワークフローを最適化します。各開発段階を行き来するコードとプロセス全体を通して自動化されたテストにより、エラーを確実に検出でき、本番環境に移行する前にロールバックできます。

CI/CD ツールを使用する企業は増え続けており、ソフトウェア開発の方法も改善されています。CI/CD により、DevOps チームによるデプロイは年、四半期、月単位のイベントから1日に数回行われるイベントに変わりました。



# CI/CD によりもたらされる10の価値

CI/CD の概要を理解したところで、その価値をまとめてみましょう。

## 1. 受け渡しが減る

開発パイプラインでの受け渡しが多ければ多いほど、障害が起こる機会が増え、プロセスの複雑性が増します。

## 2. 開発スピードが上がる

CI/CD により、すべての開発ステージがスピードアップされます。プロセス全体にわたって反復が高速化されるため、すべてのチームの効率が上がり、開発者は自信を持って他のプロジェクトに取り掛かれます。

### 3. デプロイの頻度が上がる

多くとも数週間に1回だったリリースを1日に6回以上行えます。

### 4. テストスピードが上がる

開発ワークフローで最も時間のかかる部分が取り除かれ、開発者は価値の高いプロジェクトに取り掛かれます。自動テストにより、チームは素早くフィードバックを得られ、本番環境（最悪の場合には最終リリース）でバグを検出するよりも早く障害に気づけます。

### 5. バグが減る

開発プロセス全体でテストが自動化されることにより、発生した時点でバグを検出でき、マスターブランチに移行する前にロールバックできます。これにより、全体のコードの質が上がり、各リリースを期待通りに動作させることができます。

### 6. コンプライアンスを改善する

コンプライアンスタスクを開発ライフサイクルに組み込むことで、コンプライアンス違反のアプリケーションをリリースするリスクを減らせます。コンプライアンスの自動化により、監査を簡単に完了できるようになり、（特に規制が厳しい業界で）コストのかかるミスを防ぐことができます。

### 7. イノベーションに取り組む時間が増える

インテグレーションのメンテナンスや差別化につながらない IT 投資に費やす時間を短縮すれば、他の分野にリソースを使うことができるようになります。

### 8. 開発者の満足度が上がる

開発者が自信を持って作業を行い、バグの発見に数週間を費やすことなく、素早く修正できるようになります。

### 9. 間接費が減る

組織が抱える開発者の数には限りがあります。開発時間は有償の業務時間であることが多く、手動でのデプロイやテストを行うと IT 予算を逼迫する場合があります。ワークフローの自動化により、手動タスクが減り、予算効率を向上させることができます。

### 10. 一貫したプロセスを実現する

開発ワークフローでの自動プロセスを増やすことで、誰かがプロセスのステップを忘れるということがなくなります。一貫したビルドを実現できる上に新人開発者のトレーニングを簡略化でき、組織はビルド方法やリリースのタイミングに対するコントロールを強化できます。

---

# お使いの CI/CD は最適化されていますか？

生産スピードを正確に測定するには、部分ごとではなく、SDLC 全体を評価する必要があります。テック企業の役員である Gary Gruver 氏は著書『Starting and Scaling DevOps in the Enterprise (企業で DevOps を開始し、スケーリングする方法)』で DevOps を開始する際に、開発者だけでなく、最初に新しい機能に影響を与えたビジネスアイデアまで視点を広げることを勧めています。アイデア、開発、テスト環境、本番環境に至るまで、SDLC を開発者目線で分析しましょう。アイデアを実践するために個人が行うべきステップは何でしょうか。このプロセスにより、組織での価値の流れが決まり、システムのボトルネックを特定できます。

シームレスな継続的インテグレーションおよびデリバリーは、アプリをデプロイする上で必要なすべてを備えたものであるべきです。残念ながら、CI/CD を利用する多くの組織が、現在使用しているツールにより、ワークフローの最適化を阻まれています。この傾向は、何らかの問題が発生した場合に特に顕著になります。

パイプラインに障害があると、開発者は特定のツールへのアクセス権がないために、問題を修正するために必要な情報を得られない場合があります。この場合、問題を修正する権限を持つチームと問題を確認する権限を持つチームを作ってお互いに調整しなければならないこともあります。両方のチームが権限を持っている場合でも、異なるツールを使用して共同作業をすることになります。例えば、Jenkins CI はコードレビュー機能を備えていないため、チームが正しい記録を行うには、SCM と CI ツールを行き来しなければならなくなります。

複雑なツールチェーンはインテグレーションや定期メンテナンスがなければ機能しません。不安定なサービスと脆弱な構成はライフサイクル全体に影響を及ぼす中断につながり、ビルドを開始して完了するまで、長時間待つことになります。善意を持って行動した管理者が「簡単な」プラグインのアップデートまたはインストールでシステムを停止させてしまう話もよくあります。Jenkins のクリエイターである川口耕介氏は [ブログ記事](#) で信頼性の欠如や使用するプラグインの多さからフランケンシュタインプラットフォームと評価されている点などのプラットフォームの欠点を認めています。清々しくも正直な見解であり、Jenkins が積極的に問題に取り組んでいることを示していますが、多くの DevOps チームがすでによく知っている問題を強調する内容でもあります。

それは、インテグレーションはすべて完璧に動作するけれども、いつ停止するか分からないということです。

---

# シングルアプリケーション CI/CD

当社では、SLDC 全体に可視性をもたらすシングルアプリケーションが、すべての開発ステージを含めた最適化の点でベストであると考えています。すべてを一元的に管理できれば、ワークフローのボトルネックを特定し、各要素がデプロイスピードに与える影響を評価するのも簡単です。

シングル DevOps プラットフォームのステップはそれぞれ、コードをマージできるかどうかという意思決定を下すための情報を提供します。オールインワンの CI/CD ソリューションがなくとも全ステージを実行することはできますが、総合的なツールを持つことで、信頼できる唯一の情報源 (Single Source of Truth) を作成でき、各ステップを素早く監視し、エラーを見逃す可能性を削減することができます。Forrester 社のリサーチでも、追加設定のいらぬツールチェーンソリューションにより、全体的なセキュリティが改善したと IT 担当者が認めていることを示しています。

## シングルアプリケーション CI/CD の価値

- » 開発ライフサイクル全体を可視化
- » 全開発ステージで唯一の情報源 (Single Source of Truth) を提供
- » 複数のアプリケーションへのログインが不要
- » 単一のインターフェイスで操作が簡単に
- » 単一のアプリケーションでインストール、メンテナンス、スケーリング、バックアップ、ネットワーク、セキュリティを実現
- » 認証管理が簡単に
- » 運用コストを削減

継続的デリバリーは CI と緊密に連携しており、すべてのコードのテストが完了し、本番環境に移行する準備ができた際の移行プロセスもシームレスになります。効率的な CI/CD 戦略では、部分的ではなく、最初から最後まで自動化を採用しています。

---

## CI/CD ツールの差異の比較

チームは豊富な CI/CD ツールから選択できます。ソリューションの価値を差別化する大きな要素はコストと機能になるでしょう。また、現在および今後のニーズに CI/CD ツールが適切かどうかといった要素も関係します。特定のプラットフォームの機能を並べて比較することも組織にとって役立つでしょう。

## GitLab CI/CD と Jenkins CI の比較

Jenkins はビルド自動化ツール兼 CI/CD ツールとして最も人気が高いものの1つです。利用可能なプラグインが数百もあり、その機能を組み込める柔軟性で、ビルドやデプロイ、自動化プロジェクトをサポートします。



### ビルトイン CI/CD

CORE	STARTER	PREMIUM	ULTIMATE
FREE	BRONZE	SILVER	GOLD



GitLab には CI/CD が組み込まれており、別途インストールする必要はなく無料で利用できます。これを使って、ウェブサイト (GitLab Pages) や Web アプリのビルド、テスト、デプロイを行います。ジョブの結果はマージリクエストに表示されるので、簡単にアクセスできます。

Jenkins も GitLab も個別インストールが不要の CI/CD ツールを提供しています。

### CI/CD 水平オートスケーリング

CORE	STARTER	PREMIUM	ULTIMATE
FREE	BRONZE	SILVER	GOLD



GitLab CI/CD のクラウドネイティブアーキテクチャでは、ワークロードが増えても、新しいノードを追加することで簡単に水平オートスケーリングを実行できます。GitLab Runners は新しいコンテナを自動でスピンアップまたはスピンダウンすることで、パイプラインを直ちに処理してコストを最小限に抑えます。

クラウドコストを最小限に抑え、パイプラインを効率よく稼働させ続ける自動スケーリングランナーを提供するのは GitLab のみです。

### CI/CD ログのステップフォールディング

CORE	STARTER	PREMIUM	ULTIMATE
FREE	BRONZE	SILVER	GOLD



各コマンドのジョブログ出力を折りたたみます。

ジョブログを折りたためるのは現在 Jenkins のみです。GitLab は今後のリリースでこの機能を追加する予定です。

透明性は当社のコアバリューの1つであり、当社が GitLab とその他の DevOps ツールとの比較を当社 Web サイトで表示しているのもそのためです。Jenkins も CI/CD を提供していますが、他のビルトイン機能で劣っており、プラグインへの依存性を見ても Jenkins の所有コストは徐々に高くなっていく可能性があります。 [詳細な比較を見る](#)



## 煩雑な設定なしに CI/CD でプロジェクトを開始

GitLab CI/CD はソースコードの管理や計画、監視などを含むアプリケーションにすでに組み込まれています。DevOps ライフサイクル全体を対象としたシングルアプリケーションですので、すべての操作を1つの会話でやり取りでき、チーム全体で表示できます。GitLab の CI/CD は追加設定が必要ありません。当社が CI/CD を提供する目的は DevOps チームが管理にかかる時間を短縮し、クリエイティブな作業にかかる時間を増やすことです。

## GitLab CI/CD へ移行したチームの例

### Ticketmaster

GitLab CI/CD のサポートにより週1回のモバイルリリースが可能に

「当社では2月以降、モバイルアプリを1週間に1回リリースできています。過去数回のリリースが成功につながったのは GitLab CI の貢献も大きいでしょう。サイクルタイムやリリースを高速化できたことで、他の価値も享受できました。各リリースでの変更が少なくなったため、クラッシュ回避率やストアでの評価も上がりました。また、ビルドを待つ時間が短縮され、製品の質の向上に使う時間を増やすことができました。当社のファンはより早く機能を利用できるようになり、質が高く、定期的に改善される製品をお楽しみいただけます。また、GitLab で利用できる CI 分析を追加のスコアボードとして利用することで、当社チームは将来に向けた最適化と改善を実現しています。」

— ANDROID 開発チーム リードエンジニア、JEFF KELSEY 氏

[同社の事例を読む](#)

### Cloud Native Computing Foundation

一元化された CI/CD システムにより多数のクラウドプロバイダーにわたって複数のプロジェクトを管理する上での複雑性を排除

「所属するコミュニティがプロジェクトに集中しながら、そのプロジェクトと他のプロジェクトを組み合わせられるようにすることは至難の業です。CNCF CI ワーキンググループのようなグループを持つことでガイドを提供し、対話の機会を設けることで、各チームがより大きなグループにおける役割と背景を認識できるようにすることは、今でも当社の課題であり、CNCF のさまざまなグループと話し合

っています」と McClimans 氏は語ります。「話し合いの後に素早く GitLab を利用できるようになり、意義のあることを共に成し遂げ、成果を出すことができました。DevOps および CI 従事者として、これほどまでに素早く成果が出たのは初めてのことです。」

— CROSS-CLOUD CI PROJECT 共同創設者、CHRIS MCCLIMANS 氏

## 同社の事例を読む

プロセスを自動化し、コードの質を高めることで、CI/CD ツールは組織の時間と費用を節約します。コードの自己テストやビルドの自動化、バグの早期検出、進捗の監視といった機能により、時間効率を最適化してリリースサイクルを短縮できます。また、CI/CD の競争優位性は、アプリケーションにとどまらず、チーム間のハンドオフを促進し、一体感のある DevOps 文化を創生することにあります。プロセス効率上がり、手動タスクが減り、可視性が上がることで、組織の人材保持率も上がり、より優秀な人材を惹きつけることができます。シングルアプリケーション CI/CD により、チームはこういった利点をさらに拡大し、より優れたシームレスな体験を提供します。

[GitLab 無料トライアルを開始](#)

## GitLab の概要

GitLab はシングルアプリケーションとして提供される完全な DevOps プラットフォームです。Concurrent DevOps を利用できる唯一のプラットフォームである GitLab は、今日のツールチェーンから組織を解放します。GitLab は比類なき可視性、ワンランク上の高い効率性、総合的なガバナンスを提供し、変更の計画からその効果のモニタリングまでにかかる時間を大幅に短縮します。これにより、ソフトウェアのライフサイクルが 200% 速くなり、事業スピードを抜本的に改善します。

GitLab と Concurrent DevOps はソフトウェア開発ライフサイクルの全段階を通して効率性を高め、サイクルタイムを短縮します。製品、開発、品質管理、セキュリティ、運用チームがシングルアプリケーションで同時に作業を進められる初の機会です。ツールを統合して同期したり、ハンドオフを待って時間を無駄にする必要はありません。1つの会話に全員が参加でき、さまざまなツールにまたがる複数のスレッドを管理する手間がなくなります。また、単一の信頼できるデータソースを使用してライフサイクルを完全に可視化し、トラブルシューティングを簡略化して管理記録を強化できるのは GitLab だけです。すべてのアクティビティを一貫したコントロールの下に管理し、セキュリティとコンプライアンスを後付けではなく、最優先事項として扱います。

オープンソースで構築された GitLab は数千人の開発者と数百万人のユーザーから成るコミュニティの貢献により、新しい DevOps イノベーションを継続的に提供しています。Ticketmaster、ING、NASDAQ、Alibaba、Sony、Intel など、10 万社を超える企業が GitLab の優れたソフトウェアを素早く提供する能力を信頼しています。

