



コンテナ、Kubernetes、 そしてクラウドネイティブ アプリケーション デリバリーへの道

ビジネスリーダーとITリーダーのための実践ガイド



このホワイトペーパーの目的

オリンピックは、ラテン語で「Citius, Altius, Fortius」、つまり「より速く、より高く、より強く」というスローガンを掲げています。これは現代のほぼあらゆる企業組織にも通用する考え方です。新たな顧客の需要と急速に変化する市況に押される形で、ビジネスは絶えず迅速な対応を実現し、高レベルな効率を追求し、より力強いパフォーマンスを目指しています。

2020年代を迎える私たちの生活の中心には、テクノロジーが推進するデジタルエコノミーがあります。この圧倒的なデジタルエコノミーは、ソフトウェアが実現するサービス、製品、ソリューションが支えています。さらに、第四次産業革命と呼ばれる大規模なカルチャーの変革も進行しています。人工知能 (AI)、機械学習 (ML)、ロボット、モバイルデバイス、モノのインターネット (IoT)、スマートシティまたはスマートスペース、仮想現実 (VR) などのテクノロジーが、ビジネスに変革をもたらしています。

テクノロジーは今やビジネスの価値創出に欠かせない要素です。先見の明に富んだ企業や IT リーダーは、以下のような目標を達成するために、よりスマートでアジャイルかつ新たなアプリケーションデリバリーの手法が必要だと考えています。

- 新しい、より優れた製品を市場に迅速に投入し、顧客の需要にすばやく対応する。
- イノベーションを加速し、コストを削減して、運用効率を改善する。
- 新たなビジネス機会をフル活用し、収益性と市場への浸透率を最大限まで高める。
- 競争力を向上させる。

業界のトレンドは、有益なテクノロジー改善をより迅速に導入するためのクラウドネイティブなアプリケーションデリバリーアプローチに移行しつつあります。このアプローチには、アジャイルな開発手法の導入、コンテナ化されたアプリケーション、マイクロサービスベースのアーキテクチャ、統合された DevOps チームとプロセス、エンドツーエンドのデリバリー自動化などが含まれます。こうした要素はすべてデジタル変革の主要な要素であり、現代的なビジネスの必須事項として広く認知されています。

このホワイトペーパーは、ビジネスリーダーや IT リーダーに次のような情報をお届けします。

1. アプリケーションデリバリーのクラウドネイティブアプローチの概要とともに、このアプローチを実現する実践的かつ現実的な方法
2. あらゆる要素を取り込むコンテナとモダンアプリケーションデリバリーソリューションを用いてクラウドネイティブアプローチを推進する論理的根拠
3. この新しいクラウドネイティブアプローチのメリットを活用する方法



クラウドネイティブなアプリケーションデリバリーアプローチへの移行

開発者や IT リーダーなど、テクノロジートレンドを追跡・分析する人はだれでも、コンテナやクラウドネイティブアプリケーションといったトピックに注目しています。こうしたトピックはニュースや業界カンファレンスで議題となり、多くの人々の関心を集めています。

「クラウドネイティブ」とは、正確にはどういう意味なのでしょう？パブリック、プライベート、ハイブリッドクラウドで実行することを意図したアプリケーションクラスというイメージですが、クラウドネイティブはそれだけに留まりません。クラウドネイティブアプリケーションは、効率的かつ自律的にスケールします。部分的な障害を想定して高い耐障害性を兼ね備え、コストやパフォーマンスの最適化に応じて速やかに柔軟な環境の変更を可能にするなど、多くのメリットをもたらします。つまり、クラウドネイティブは、今日のクラウドプラットフォームの非常にダイナミックなテクノロジー環境において効果的に機能します。

一方で、基本的な定義として、クラウドネイティブアプリケーションは、あらゆる規模で迅速かつ効率的にデリバリーできます。新機能をすばやく頻繁に、高度に自動化された方法で導入できるため、毎日数千ものアプリケーションを継続的にデリバリーできます。このクラウドネイティブアプローチをアプリケーションデリバリーに採用することで、イノベーションとビジネスを加速し、付加価値の高い新機能を迅速に導入することで、競争力を高めることができます。

簡単に言えば、クラウドネイティブアプローチは、大規模で迅速かつ効率的に新しいクラスの最新アプリケーションデリバリーを可能にします。一見単純なようですが、それにはパラダイムシフトが伴います。クラウドネイティブアプローチには、アプリケー

シヨンの設計、開発、導入、運用方法について再考する必要があります。新たなテクノロジー、アプリケーション設計の見直し、ビジネスプロセスと組織の変更などを組み合わせて考える必要があります。こうしたハードルをすべて乗り越えなければなりません。それでも、大きな成果を上げることに焦点を合わせた組織、特にアプリケーションを高速かつ大規模に開発、展開、管理することを望む組織にとっては、労力と投資に見合うビジネス上の利益と価値があります。

多くの組織では、クラウドネイティブなアプリケーションデリバリーアプローチへの転換がすでにスタートしており、複数レベルで同時進行している状態です。

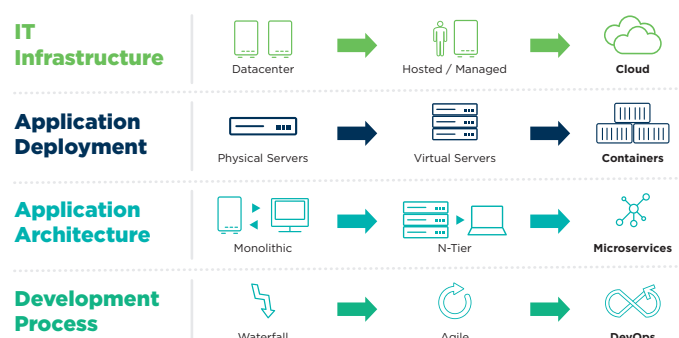


図1：アプリケーションデリバリーの変革は複数のレベルで進行

- インフラストラクチャ面では、ハードウェアで純粋に構成されたデータセンターから、柔軟なソフトウェアデファインドインフラストラクチャ (SDI) やクラウドベースアプローチへの移行が順調に進んでいます。
- アプリケーションの展開では、物理的なサーバー集約型環境の大部分が仮想環境や仮想マシン (VM) に移行しました。そして、現在進んでいるコンテナへのシフトは新たなパッケージメカニズムです。より少ないリソースで、ポータビリティとアジリティを高めることができます。
- 初期のモノリシックアプリケーションはマルチ階層アーキテクチャに置き換えられ、開発者はより柔軟に再利用可能なコードを構築できるようになっています。マイクロサービスアーキテクチャへの最新の移行により、小規模の疎結合コードからクラウドネイティブアプリケーションの作成が可能になっています。各マイクロサービスは個別の展開とスケールが可能です。
- ソフトウェア開発手法も同時に進化しています。従来のウォーターフォール方式は、高度に構造化された直線的で順次的なワークフローであり、概してソフトウェアリリースサイクルが長くなります。それに代わるアプローチとしてアジャイル開発方式の導入と成熟が進み、代替アプローチとして広く普及しています。反復的でコラボレーティブなチームベースのテクニックにより、コードを高い頻度で段階的に改善し、リリースサイクルを

短縮します。アプリケーション運用をループに組み込むことで、新たな DevOps 手法は、アジャイルな開発手法をクラウドネイティブ化した延長線上に位置付けることができます。自動化されたアプリケーションの展開とライフサイクル管理を追加することで、新しいアプリケーションを本番環境に迅速にリリースできます。開発チームと運用チーム間に存在した障壁も取り除かれ、ソフトウェアデリバリーを大幅に加速できます。

規模を問わず、ほぼあらゆる市場セグメントの組織は、新しいアプリケーションを本番環境や市場に短期間で投入するための取り組みをすでに開始しています。そうした組織は、アジリティを高めて、企業全体でアプリケーションの効率的なビルド、配信、管理方法を改善する必要性を把握しています。しかしながら、今のところ、多くの組織はクラウドネイティブなソリューションを熱望していますがゴールへの道のりはこれからです。

ほとんどのビジネスの意思決定者は、成功を収めているデジタルネイティブ、e コマース、インターネットベースのスマートビジネスがクラウドネイティブを武器としてうまく活用していることを知っています。それでも、クラウドネイティブがスタートアップの専売特許だと思い込んだり、先進的な取り組みを採用できる組織は一部だと考えたり、「クラウド時代生まれ」のメンタリティを持つ組織だけのものと決めつけたりする必要はありません。

Application Delivery Transformation Journey

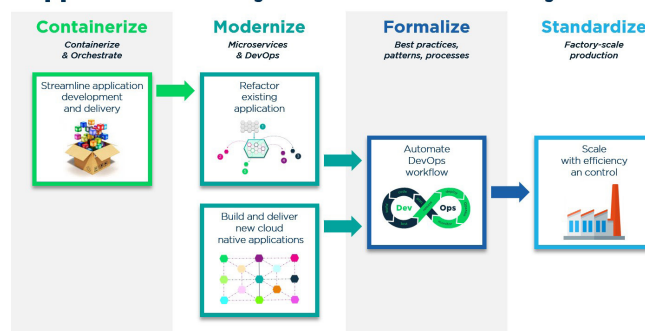


図2：クラウドネイティブへの道のり

先見の明のあるビジネスや IT リーダーは、競争力を獲得するにはアプリケーションデリバリーへの新たなアプローチが必要だと認識しています。

図2のように、クラウドネイティブへの道のりはいくつかの段階に分かれています。各段階は、この道のりにおける足掛かりとして独立した価値を持っています。つまり、クラウドネイティブは、実践的でバランスの取れた現実的かつ新たなアプリケーション構築アプローチを模索する成熟したビジネスにも適しています。既存のアプリケーションとシステムを並行して運用しながら、新しいコンテナ化アプリケーションでクラウドネイティブへの移行を開始することも可能です。さらに、適切あるいは最適と判断した時点で、古いアプリケーションを刷新することもできます。時間をかけて成熟させ、全体的なアプローチを拡張し、標準化して、しかるべきタイミングで規模をスケールできます。



ステップ1

クラウドネイティブへの道筋はコンテナから始まる

アプリケーションデリバリーへの変革への道筋では、どの段階においても、アプリケーションコンテナが不可欠なコンポーネントになります。これは紛れもない事実です。業界アナリストは、世界中の組織の75%がコンテナ化アプリケーションを本番環境で使用しており¹、コンテナ市場は2022年までに43億ドルに成長すると予想しています。² Linux Foundationは、コンテナの普及率と重要度が急速に上昇していることを受けて、採用担当者の57%は組織に役立つコンテナの専門知識を持った人材を積極的に採用していると報告しています。³

コンテナはすでに成熟した、メインストリームのテクノロジーです。現時点で、ほぼすべての業界や事業分野で使用されています。その分野は、金融、科学・研究、公共事業、小売、都市計画、スマートスペース、エンターテインメント、製造、ヘルスケア、農業など、多岐にわたります。

既存のアプリケーションのデリバリーを効率化

多くの組織は、コンテナとともに、自動コンテナ管理を活用しています。コンテナは、既存のアプリケーションの開発および展開作業を効率化する最も速く最も簡単な手法です。

コンテナは、開発者がコードをパッケージ化する便利な道具として、仮想マシンと比較しても多くの有益なメリットをもたらします。コンテナはVMより軽量なため、起動が速く、開発者自身で容易にビルドと展開が可能です。しかし、最も重要なのは、ポータビリティに優れているということです。コンテナは、コードの実行に必要なすべてのコンポーネントが含まれているため、あらゆる環境で一貫して動作することが保証できます。

その結果、開発チームと運用チームは、開発、テスト、本番投入まで、コンテナ化されたコードを複数の異なる環境で実行できます。それぞれの環境で同じように動作するようにわざわざ手作業で構成する必要はありません。さらに、異なる環境の所有者間で発生するストレスがなくなります。関係者全員が同じコードを実行する必要がありますが、担当者の優先度やスキルに依存することなくまったく同じ環境を構築可能です。

コンテナがないと、こうした差異を埋めるためにコストがかさみ、環境関連の問題のデバッグに要する時間が長引くこととなります。コンテナがあれば、異なるチームが異なる環境を維持したまま、全員で同じアプリケーションコードを容易に実行できます。

コンテナによる時間の節約効果は、アジャイルで反復的な開発モデルを採用している組織に最適です。アプリケーションを迅速に構築し、サイクル時間を大幅に短縮できます。運用の一貫性を確保できるため、コンテナ化されたコードを開発/テスト環境から本番環境に移行する際のリスクが減少し、アプリケーション開発と運用プロセスも容易に統合できます。

Kubernetesは運用チームのコンテナ体験を向上させます

コンテナとKubernetesを組み合わせることで、コンテナ化されたアプリケーションの導入と管理を自動化することで、大きな効果が得られます。これらには、コンテナのスケジューリングとサービスディスカバリ、コンテナの性能と可用性の監視、コンテナのスケール、ロードバランシングと自己修復機能、さらに更新の自動ロールアウトとロールバックを含むアプリケーションメンテナンスなども含まれています。

1 Gartner. 6 best practices for creating a container platform strategy. <https://www.gartner.com/smarterwithgartner/6-best-practices-for-creating-a-container-platform-strategy/>

2 451 Research. <https://451research.com/451-research-says-application-containers-market-will-grow-to-reach-4-3bn-by-2022>

3 Linux Foundation. 2018 open source jobs report. <https://www.linuxfoundation.org/press-release/2018/06/new-2018-open-source-technology-jobs-report-released-today-rapid-growth-in-demand-for-open-source-tech-talent/>

Kubernetes は、業界のデファクトスタンダードとして急速に普及しており、最新のアプリケーションが稼働するメインプラットフォームになると期待されています。調査によると、60% の企業がすでに Kubernetes を導入しており、さらに 27% が近い将来に導入を計画しています。⁴ すべての主要なクラウドサービスプロバイダーは Kubernetes プラットフォームサービスを提供しており、コンテナ化アプリケーションのオンプレミスでの導入を望んでいる企業は、自社データセンターへの Kubernetes プラットフォームの導入が必須となりつつあります。

コンテナのポータビリティに適したパブリッククラウドの Kubernetes サービスを組み合わせることで、マルチクラウドインフラストラクチャのメリットを容易に得ることができます。アプリケーションチームは、「リフト&シフト」技術を利用することで、既存のアプリケーションを必要に応じて新しいプラットフォームや異なるプラットフォームに移動できます。複数のクラウド環境にまたがるアプリケーション管理は長年望まれてきた機能ですが、コンテナと Kubernetes により、その機能が経済的に合理的な方法でついに実現されつつあります。

多くの独立系ソフトウェアベンダー (ISV) は、ソリューションのコンテナ化バージョンも提供しています。VM ではなくコンテナを利用することで、リソースを大幅に節約できます。Kubernetes を使用して ISV ソフトウェアのデプロイと継続的な運用管理を自動化することで、より大きなアドバンテージが得られます。特にアプリケーションをクラウドネイティブな手法で設計している場合は、メリットがさらに大きくなります。すでに稼働しているソリューションに頻繁なアプリケーション更新をスムーズに組み込むとともに、アプリケーションの可用性やスケールを自動的かつ効率的に管理できます。

60% の企業が Kubernetes をすでに導入しており、さらに 27% が近い将来に導入を計画しています。

⁴ RightScale, 2019.State of the Cloud Report.<https://info.flexerasoftware.com/SLO-WP-State-of-the-Cloud-2019>



ステップ 2

より大胆なモダナイゼーション戦略

クラウドネイティブへの道筋の次の段階では、よりクラウドネイティブな手法によるアプリケーションの構築、開発、展開、管理を開始する必要があります。多くの組織にとって、最も手軽なスタート地点は新しいアプリケーションプロジェクトを開始することです。ゼロから構築することで、既存システムの制約にとらわれることなく、確実な成果を得ることができます。まずはシンプルなアプリケーション配信を実践した後に、より複雑なアプリケーションに取り組む準備ができます。その過程での学習や教訓を活用し、既存のアプリケーションをリファクタリングすることで、クラウドネイティブなアプローチからより多くのメリットを享受できます。

新しいクラウドネイティブアプローチの構築

前述のとおり、クラウドネイティブは、アプリケーションをあらゆる規模で迅速かつ効率的にデプロイできるように、モダンアプリケーションの新たなクラスを構築するアプローチです。アプリケーションのリリースサイクルを数か月から数週間、数日から数時間、場合によっては数時間から数分まで短縮し、迅速なイノベーションと新たな価値創出を促進します。クラウドネイティブアプリケーション自体も拡張性と耐障害性に優れ、短時間で発展させることができます。さらに、独自の運用要件の多くを動的かつ自動的に管理できるため、まったく新しいレベルの運用拡張性を実現できます。

クラウドネイティブアプローチは、アプリケーションデリバリーライフサイクルのほぼあらゆる側面で効果をもたらします。たとえば、以下のような効果があります。

- マイクロサービスベースのアーキテクチャを導入し、アプリケーションを個別に展開、スケール、運用できるようにする。
- 継続的インテグレーション / 継続的デプロイメント (CI/CD) パイプラインの自動化がサポートする反復的でアジャイルな開発アプローチを適用する。

- コンテナを使用してコードの効率的なパッケージングとポータビリティを実現する。
- Kubernetesなどの強力なコンテナ管理プラットフォームを活用し、アプリケーションの展開とライフサイクル管理業務を自動化する。
- アプリケーション開発と運用プラットフォームおよびプロセス (DevOps) を統合し、サービスとアプリケーションをより頻繁かつ自動的に展開できるようにする。

既存のビジネスクリティカルなアプリケーションの多くは「壊れるまで修正しない」という姿勢で放置するというアプローチも有効です。現実には、再設計、再構成、再構築に適さないアプリケーションもあり、再構築してもビジネスや運用上のメリットが何も生まれないケースがあります。古くなったアプリケーションについては、モダナイゼーションを施しても、コンテナを使った前述の「リフト&シフト」のメリット以上の効果が得られない可能性があります。

Increased Agility with Cloud Native Applications

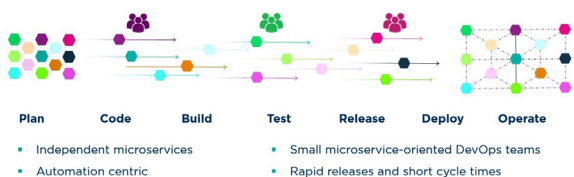


図 3：クラウドネイティブなアプリケーションデリバリーのワークフロー

現状で、ほとんどのアプリケーションはクラウドネイティブではありませんが、多くの企業は多少のクラウドネイティブ手法をすでに導入しています。たとえば、アジャイル開発プロセスをすでに使用していたり、CI/CDパイプラインの自動化を導入していたり、コンテナを一定の範囲で使用している場合があります。こうした要素は個別に効果を発揮するため、単独でも実装する価値があります。しかし、より多くのクラウドネイティブ手法を併用することで、さらに大きな成果が得られます。このように新しい要素を既存の環境に追加することが、次のステップになります。たとえば、アジャイルな開発/テストチームを拡張して運用スタッフを加える、自動化された Kubernetes 開発を既存の CI/CD パイプラインに統合するなどです。

コンテナは重要な要素ですが、ここで説明しているクラウドネイティブへの移行は、単純にコンテナを使用するケースより複雑になります。しかし、さまざまなメリットが得られるクラウドネイティブアプローチを、そのアプローチに適したエンタープライズアプリケーションに対して、さらにその組織に適したタイミングで導入することができれば、魅力的な成果を獲得できます。

既存のアプリケーションのモダナイゼーション

コンテナを使用したアプリケーション開発と展開の合理化が勢いを増しているのは疑いの余地がありません。一方で、新たなアプリケーションに対するクラウドネイティブアプローチについても、多くの組織が将来的な導入を検討しています。既存のアプリケーションのモダナイゼーションは第三のユースケースであり、間違いなく最も複雑で最も実装が困難です。基本的な外的挙動を変えずにアプリケーションを再構築しなければならず、拡張性、安定性、メンテナンス性といった機能以外の特性も改善しなければなりません。

Modernizing Application Architectures

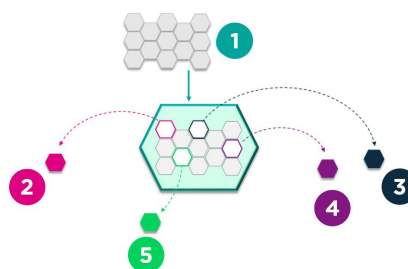


図 4：既存のアプリケーションのモダナイズ

しかし、既存のアプリケーションをモダナイズした方が明らかにメリットがあるケースもあります。通常、このプロセスでは、アプリケーションの一部を独立したマイクロサービスにリファクタリングします。アプリケーション全体をリファクタリングしても効果が得られない場合もありますが、候補となる領域を特定することはできます。あるアプリケーションで他のアプリケーションも使用できる機能を導入する場合は、マイクロサービスを個別に作成してその機能を提供した方がよいでしょう。たとえば、シングルログイン用のマイクロサービスを作成すると、複数のアプリケーションでログインを一貫して処理することができ、開発時間も短縮できます。モダナイゼーションにより、既存のアプリケーション全体ではなく、アプリケーションの一部を独立したマイクロサービスに置き換えることもできます。一部の頻繁に更新が必要とされるアプリケーションコンポーネントは独立した効率の高いマイクロサービスとして導入した方がよいでしょう。もし他への影響度がなく、入念に設計され、メンテナンスが行き届き、きれいな形で機能的なロジックとコードがカプセル化されているアプリケーションであれば、一定レベル以上のリファクタリングによりメリットを享受できます。

クラウドネイティブアプローチの目的は、新たなクラスのモダンなアプリケーションをあらゆる規模で迅速かつ効率的にデリバリーすることです。



ステップ 3

クラウドネイティブなアプリケーションデリバリーアプローチの成熟

クラウドネイティブなアプリケーションデリバリーの初期体験は、学習機会として大いに有用です。たとえば、コンテナと Kubernetes の使用、マイクロサービスベストプラクティスのアプリケーション設計および開発の実践、分散したシステムの管理、人員とプロセスを結びつける新たなワークフローの導入などの初期体験があります。失敗しても、インサイト（知見）を得て改善を進めることができます。時間を重ねて得られた教訓が、ベストプラクティスとして蓄積されます。これらは、ミスなく、繰り返し円滑な導入を進めるためのガイドラインになります。しかし、このプロセスに終わりはありません。新たに獲得した知識が絶えず統合され、システム全体が成熟し進化していくからです。

ベストプラクティス、プロセス、手法の形式化

この形式化プロセスは、アプリケーションチーム内だけでなく、事業部単位、コミュニティ、業界全体でも実施されます。この 10 年、クラウドネイティブコンセプトは、テクノロジーイノベーターやこのコンセプトをいち早く採用した組織によって導入され、洗練されてきました。現在、こうした組織はクラウドネイティブの主流となり、これまでの知識を有効に活用してクラウドネイティブの習得と導入に要する時間を短縮しています。主流となった組織は、自由に利用できるリソースの大規模なエコシステムを活用し、Twelve-Factor App⁵ 手法などの実証済みのモデルやクラウドネイティブアプリケーションの導入ライフサイクル全体を自動化できてすぐに利用可能な Cloud Foundry⁶ のメリットを享受しています。

企業全体でのスケーリング

前述のように、クラウドネイティブアプローチの目的は、モダンアプリケーションの新たなクラスをあらゆる規模で迅速かつ効率的にデリバリーすることです。クラウドネイティブのアーティファクト、設計原則、プロセスワークフローは、明確に定義され、高度に自動化されたアプリケーションデリバリーフレームワークに

組み込むことができます。こうした技術を組み合わせ、あらゆる規模で高効率、低コストでアプリケーションを提供した際に、高い生産効率を提供する工場のような大規模生産に対応可能な完成されたシステムとなり、クラウドネイティブアプリケーションの究極のメリットを実現します。

ヘンリー・フォードは、工場生産ラインと組み立て手法を導入して有名な T 型フォードを製造し、自動車業界に革命をもたらしました。自動化、合理化されたこの連続フロー製造プロセスは、カスタムビルドアプローチに取って代わるパラダイムシフトでした。迅速で効率的な大規模生産を可能にすることで、自動車の価格が大幅に下がり、身近なものになりました。

同様に、クラウドネイティブの原則に従うことで、個々のアプリケーションのためにカスタムメイドされた数多くの導入・運用のプロセスを使う必要がなくなります。そして、最新のアプリケーションをデリバリーするための標準化されたプロセスを導入できます。この自動化、合理化されたプロセスにより、工場のような生産量と速度が得られます。

ただし、完全にはクラウドネイティブではないアプリケーションを管理するためのカスタムメイドのワークフローとプロセスがまったく不要になるわけではありません。カスタムメイドが必要なアプリケーションは常に存在します。しかし、今後はクラウドネイティブアプリケーションがますます増えることになります。標準の自動化プロセスで効率的に開発、展開、管理することができ、T 型フォードと同様に、効率、コスト、競争力の点で圧倒的なメリットをもたらします。

その場合、当然ながら組織全体でアプリケーション生産のスケールリングにまつわる実務的な作業が発生しますが、通常その作業は IT 運用チームが担当します。一貫性、ガバナンス、管理のための新たな要件が必要になります。特に、規模の調整において、異なる手法やプロセスを使用する多種多様なチーム間の統合と集約が求められる場合はなおさらです。標準化されたグローバルな手法を確立することは、全体的な運用効率を確保するために不可欠です。リソースへのアクセスを含めた複数のチームと組織の管理は、アプリケーションデリバリーシステムに関わる人員が増えるにつれてますます重要になります。さらに、リソース管理自体も複雑になり、膨大な規模になることが予想されるマルチクラウドインフラストラクチャと多数の Kubernetes クラスタを巧みに管理する必要があります。

⁵ 12factor.net: <https://12factor.net/>

⁶ Cloudfoundry.org

この工場モデルをサポートするには、独自の標準ワークフローとリソース管理フレームワークを作成して運用します。あるいは、既存のフレームワークを利用して速やかに生産量を増やし、徐々に最適な生産性を維持することもできます。私たちのお勧めは、成熟度の高い認定済みオープンソースのオプションである Cloud Foundry モデルです。



アプリケーションデリバリープラットフォームの考慮事項

適切なテクノロジー、プラットフォーム、ベンダーを選択することは、あらゆるアプリケーションデリバリー移行戦略を成功に導くために不可欠です。ここでは、考慮すべき要素について解説しましょう。

オープンソース：クラウドネイティブへの原動力

クラウドネイティブでは、ほぼすべてのサポートテクノロジーが根本的にオープンソースです。この点は注目に値します。Kubernetes はその典型例であり、Cloud Native Computing Foundation (CNCF) と Linux Foundation の管理下でホストされています。他にも、Cri-o (コンテナランタイム)、Helm (パッケージマネージャー)、Prometheus (モニタリング)、Grafana (視覚化)、Cilium (ネットワークセキュリティ)、Cloud Foundry Application Runtime (クラウドネイティブアプリケーションプラットフォーム)、Stratos (マルチクラスター管理) などがあります。オープンソースソフトウェアの長所と使用するメリットは、オープンソースソフトウェアソリューションへの準備を整えてワールドクラスのサポートを提供できるベンダーと連携する価値と重要性とともに、十分に理解されています。

クラウドネイティブテクノロジーとプラットフォームの選択肢を比較評価する際の最終候補として、世界最大の純粋なオープンソースソフトウェアベンダーである SUSE® をご検討ください。

SUSE CaaS Platform：アジャイルな企業のための Kubernetes

Kubernetes は世界中で急速に普及し、受け入れられています、すべてが順調なわけではありません。アップストリームコードを

使用して Kubernetes 環境を構築しようとする、そのプロセスは複雑で困難になります。少なくとも有用な Kubernetes 環境を構築するには、Kubernetes だけでなく、多くの補完的なコンポーネントやオペレーティングシステムも必要です。こうしたすべてのコンポーネントは個別に進化するため、相互運用性に関する継続的なテストを実施して企業の対応準備を進める必要もあります。

Kubernetes では頻りにコードがリリースされるため、機能をはるかに短時間で発展させることができます。しかし、プラットフォームを維持管理しながら、このリリーススケジュールに常に対応するのは容易ではありません。その難しさは多くの企業にとって現実的ではありません。特に、自社のコア開発の強みを集約し、運用上の複雑さを回避したいと考えている企業にとっては困難です。

SUSE CaaS Platform など、パッケージ化され、サポートが充実した CNCF 認定の Kubernetes ディストリビューションは、こうした多くの複雑さを排除し、問題を解決するのに役立ちます。SUSE CaaS Platform は、エンタープライズ対応の Kubernetes ディストリビューションです。Kubernetes を容易に使用して管理するために必要なすべてのコンポーネントを提供し、IT 運用チームに優れた体験をもたらします。プラットフォーム提供者は、本番環境に対応する Kubernetes クラスターを利用することで、オンプレミスやパブリッククラウドインフラストラクチャにかかわらず、典型的なアップストリーム Kubernetes コードのデプロイメントから 75% の高速化を達成できます。⁷ クラスターのスケールやソフトウェアメンテナンスなど、導入以降の運用も大幅に簡素化されます。

SUSE Cloud Application Platform：スケール対応のクラウドネイティブによる生産性

クラウドネイティブでは、膨大な数のアプリケーションのビルドおよびデプロイの速度と頻度をシフトアップする必要があります。「大規模生産」の技術で、新たなレベルの自動化、スケール、高速化を達成しなければなりません。そのためには、標準化されたアプローチとプロセスを適用してアプリケーションの大部分を処理し、生産性を高める必要があります。

⁷ Tymlez success story. <https://www.suse.com/c/success/tymlez/>

標準化されたプロセスにより、モダンアプリケーションを商用レベルの規模とスピードで提供。

現在幅広く浸透している Kubernetes は、クラウドネイティブに適した基盤となります。しかし、Kubernetes だけでは、あらゆるクラウドネイティブの課題を完全には解決できません。たとえば、Kubernetes は運用チームにとって柔軟でパワフルなプラットフォームとなりますが、本質的に直感的な開発者体験は提供できません。生産性を大幅に高めるには、アプリケーションデリバリーライフサイクルに携わるすべての人員が同じように使用できる、拡張性に優れ、完全に自動化された統合プラットフォームが不可欠です。

Kubernetes は、他のプラットフォームのプラットフォームとして機能するように設計されています。つまり、基盤テクノロジーレイヤーとして使用するために設計されています。他の抽象レイヤーや機能を追加することで、異なるユーザーグループに対してより優れた体験を提供できます。つまり、Kubernetes を基盤とすることで、より豊富な機能を備えたプラットフォームを構築し、クラウドネイティブアプリケーション開発者の具体的なニーズに対してより包括的に対処できる可能性があります。前述のとおり、こうしたプラットフォームをお客様ご自身で組み立てることは可能です。しかし、その必要はありません。SUSE は、[SUSE Cloud Application Platform](#) で、こうしたプラットフォームをすでにご用意しています。

[SUSE Cloud Application Platform](#) は、開発者と IT 運用担当者のニーズを完全に満たす、優れたモダンアプリケーションデリバリープラットフォームです。クラウドネイティブアプリケーションデリバリーにおいて実績のある Cloud Foundry ワークフローを導入することで、開発者の作業をシンプルにし、クラウドネイティブアプリケーションデリバリーを合理化し、生産性、効率、機敏性を高めます。SUSE CaaS Platform はもちろん、主なパブリッククラウドサービスプロバイダーが提供する Kubernetes サービス上でも稼働するため、マルチクラウド環境で使用できます。マルチテナント、マルチクラスター、マルチクラウド管理の追加サポートを備えた SUSE Cloud Application Platform は、スタート地点として最適であり、クラウドネイティブアプリケーションデリバリーの規模がどれほど拡大しても、そのまま使用し続けることができます。



結論と提案

圧倒的な規模で進んでいるデジタル時代では、最新のクラウドネイティブアプリケーションが成功の鍵になります。企業や組織は、新たな顧客の需要と急速に変化する市況に対応するため、より迅速な対応、より高いレベルの効率化、より優れたパフォーマンスを目指して、よりスマートでアジャイルな新しいソリューションを模索しています。

この状況を背景に、新たなアプリケーションデリバリー手法を獲得し、最終的に完全なクラウドネイティブソリューションへと移行するための準備が始まっています。その過程には、アジャイルな開発手法の導入、コンテナ化されたアプリケーション、マイクロサービスベースのアーキテクチャ、統合された DevOps チームとプロセス、エンドツーエンドのアプリケーションデリバリーパイプラインの自動化など、複数のステップがあります。その究極の目標は、新たなクラスの最新のアプリケーションを大規模に迅速かつ効率的にデリバリーする能力を高めることです。

どうすればこの過程を確実に進められるのでしょうか？ビジネスおよび IT 部門の意思決定者への提案は次のとおりです。

1. アプリケーション、製品、サービスが現在どのように構築、デリバリーされているか明らかにし、IT リソースと日常業務を管理するためのベースラインを確立します。
2. アプリケーションデリバリー手法とプロセスを転換するタイミングと方法について現実的な計画を立てます。シンプルで小規模なプロジェクトからスタートし、短期的な成果を上げて、そのメリットを目に見える形で提示します。その後、各段階を明確に定義した上で、プロジェクトの範囲を拡大して、より複雑な作業に取り組みます。こうした作業を通じて、組織的な体験を蓄積し、自信を高めていきます。まずはコンテナを使って既存のアプリケーションのデリバリーを合理化し、新たなクラウドネイティブアプリケーションの構築に移行して、最終的に既存のアプリケーションのモダナイズを達成します。
3. これが変革の過程であることを認識します。従来のアプリケーションとクラウドネイティブアプリケーションは長期にわたって共存します。ただし、徐々にクラウドネイティブに重心を移していきます。最終的な成熟プランでは、組織全体でベストプラクティス、プロセス、手法、プラットフォームを標準化して拡張する方法も検討する必要があります。
4. アプリケーションのコンテナ化、ワークロードのモダナイズ、新たなクラウドネイティブイニシアチブを含むエンドツーエンドのアプリケーションデリバリー変革戦略をサポートする基盤プラットフォームとして、Kubernetes をフル活用します。
5. Kubernetes 環境を強化できる既成のアプリケーションデリバリープラットフォームの調査と評価を進めます。大規模なクラウドネイティブアプリケーションデリバリーを加速するために、ビジネスニーズに最適なオプションを選択します。
6. クラウドネイティブへの道筋を計画し、先導して、ガイド役を果たすことができる SUSE などのエキスパートやプロフェッショナルなテクノロジーパートナーを選択します。純粋なオープンソースを開拓してきた SUSE は、アプリケーションデリバリーソリューション、ツール、サービスなどを網羅した包括的なポートフォリオを提供するとともに、豊富な経験をお客様と共有できます。

クラウドネイティブテクノロジーとプラットフォームの選択肢を比較評価する際の最終候補として、世界最大の純粋なオープンソースソフトウェアベンダーである SUSE® をご検討ください。



詳細については、
SUSE の Web サイトをご覧くださいか、
お近くの **SUSE ソリューションプロバイダー**
または **SUSE** までお問い合わせください。

1-800-796-3700 (米国およびカナダ)
1-801-861-4500 (その他の国)

SUSE
Maxfeldstrasse 5
90409 Nuremberg
Germany

[suse.com](https://www.suse.com)