

運用体制やマインドセットの変革も必要 有識者が語る、クラウドネイティブ／ コンテナ技術を確実にビジネスに生かす勘所

企業は DX の推進、競争優位性獲得のためにクラウドネイティブやコンテナ技術の導入を進めている。だが、まだまだ十分に活用できていないのが現状だ。クラウドネイティブやコンテナ技術の恩恵を得るには、何がポイントになるのだろうか。

今、企業ではデジタル技術でビジネスを効果的に変容させ、競争優位性をいち早く獲得するための取り組みが急務となっている。ビジネスの中心となるアプリケーションを例に取れば、素早く形にしてユーザーに届け、より良いものへと改善を繰り返すことができる環境が必要になる。

企業ではクラウドネイティブの実現に向けた取り組みが進められつつあり、CI（継続的インテグレーション）／CD（継続的デリバリー）の実践をはじめとする開発運用の効率化に注目が集まっている。

では、運用の安定性や信頼性を高めつつ、アプリケーションのリリースサイクルを速めるためのポイントはどこにあるのか。クラウドネイティブという技術面の取り組みだけではなく、運用体制やマインドセットをどのように変革していけばよいのか。長年、先端技術に追随し、「Kubernetes」や CI/CD に詳しい 2 人の有識者に話を聞いた。

コンテナ、Kubernetes、 CI/CDを正しく活用するためのポイントとは

企業がデジタルトランスフォーメーション（DX）を推進していくには、IT 基盤のクラウドネイティブ化が欠かせない。レッドハットの北山晋吾氏（テクニカルセールス本部 クラウドソリューションアーキテクト）は、IT 基盤の現状を「クラウドネイティブの中心とされるコンテナの活用は進んではいるものの、まだ浸透が緩やか」と分析する。

北山氏によると、多くの企業の担当者がまず「これまでの仮想化環境とコンテナ仮想化は何が違うのか」という技術的な違いに関心を寄せ、理解が進むと次は「これでどうメリットを得るか」というビジネス的な価値の評価に移る。最近では「コンテナを学んだので、今後の導入の参考に事例を知りたい」という問い合わせがよく寄せられているという。

日立製作所（以下、日立）の金城教幸氏（プラットフォームサービス部）も「企業でのコンテナ技術の導入は進んでいます。ただ、開発環境で試してみたものの、そこで止まっているところが多く見受けられます」と現状を見ている。また、金城氏は「コンテナとセットで DevOps や CI/CD を検討されているお客さまも多いです」と話す。

コンテナは普及しつつあるものの、本格的な活用の手前にあるようだ。「あと一歩で抜け出そうとしている段階です。しかし、今後は確実に増えていく肌感覚はあります」（北山氏）と述べ、コンテナをはじめとしたクラウドネイティブ技術を本格活用する前にあらためて認識しておくべきこととして、重要な 3 つの要素を挙げた。

「『技術』は当然のこと、ルールで定める『プロセス』、人が培う『ノウハウ』も不可欠です。技術の理解は進んでいても、既存の運用プロセスの見直しにはまだ踏み込めていないと感じています」（北山氏）

技術は導入したものの、プロセスや人の意識も変えていかないと中途半端になりかねない。例えば、これまで手作業で環境構築やアプリケーションをデプロイしていた部分を、コンテナ化によって構築からデプロイまで幅広く自動化を進めることによって、労力やコストを大幅に減らすことができる。ところが、承認プロセスなどの社内ルールがそのままのためにビジネスのスピードを阻害してしまうと「せっかくコンテナ導入してもビジネスメリットを出しにくくなってしまいます」と北山氏は述べる。

金城氏は「コンテナ、特に Kubernetes では“宣言的な（コードを用いた）管理”ができるところが大きなメリットです。しかし、従来通りと変わらない方法でも管理できてしまいます。そのため、既存のプロセスを変えていくことが後回しになってしまい、開発環境と本番環境で不整合が起きたり、コンテナのメリットを生かし切れなかったりすることもあります」と指摘する。運用プロセスまで

改革のメスを入れなければ、せっかくコンテナや Kubernetes、クラウドネイティブ技術を導入しても十分に恩恵を享受できなくなるということだ。

コードを用いた運用管理が実現できれば、開発やテスト環境を用意するために必要なタイミングでリソースを払い出すことができるようになる。仮想化環境では運用管理者に依頼して環境を準備してもらっていたが、コンテナ環境であれば開発者が自ら（運用管理者に依頼することなく）セルフサービスで環境を構築できる。しかし、仮想化環境で運用していた社内の承認プロセスが残っているのは、スピード感を出せないという状況に陥るリスクがあるのだ。

北山氏は「こうしたルールは廃止するか、チケット管理など違うものに置き換えてもいいと思います。そうでないと、承認プロセスが運用の足かせになってしまいます」と指摘する。また、現場側では問題意識やもどかしさを抱えてはいるものの、標準化やワークフローの変更という大きな話になるため、どう変えればいいのかアイデアが浮かばず、模索しているところもある。北山氏もその空気は察しており「プロセスの問題は分かっているけど、解決できずにいるお客さまがいる印象です」と話す。

CI/CD の観点からも、リリースまでのフローに無駄や停滞するところがないかどうか、全体のプロセスを再度棚卸しする必要がある。作業の属人化や、特定ツールに CI や CD が依存していないかどうかなどだ。作業のボトルネックや負荷が偏しているかどうかやうまく可視化されていないと、課題として認識できない。

よくある状況としては、開発面では Git リポジトリにアプリケーションコードをプッシュすることでテストもビルドも自動で終わる環境が整っているが、それを継続的に開発と運用チーム間で実装する運用フェーズではまだ手作業が残っていることもある。金城氏は「開発の中で、運用も含めツールで全て問題ないことを確認したら、後はクリックすれば運用できるというように、Dev と Ops の双方が高速化すれば相乗効果が生まれます」と述べる。

権限管理も重要な課題だ。金城氏は「CI ツールが CD を兼業するために、CI ツールに認証情報を渡した結果、権限が集中してしまうリスクもあります」と話す。北山氏は「テスターやレビュアーなど、デリバリー全体で権限を整理する必要があります」と指摘する。

コンテナ環境における課題解決に特化した GitOps/Pipelines

「コンテナを導入したものの、ビジネスに価値を生み出せるほど活用できていない」という状況を前進させ、プロセスや運用も含めてスピードを高めるための全体最適を図ること、そしてセキュリティも万全にしていく必要がある。

こうした状況で頼もしい存在となるのが、日立のテクニカルサービスだ。日立は Red Hat 製品の提供とサポートで長い間、経験とノウハウを積み重ね、大手企業や金融機関への導入実績を通じた



日立製作所
プラットフォームサービス部
金城教幸氏

知見を持つ。どのような課題があるかをヒアリングし、最適な構築や運用を実現するためのツールの選定やテクニカルな組み合わせを提案できる体制が整っている。

「コンテナ環境において効率化や自動化、プロセス面に課題を抱えているお客さまには、まず『OpenShift』を提案しています。そこに CI/CD、セキュリティ、マニフェスト管理など、さまざまな課題に応じて、『OpenShift GitOps』と『OpenShift Pipelines』を組み合わせ提供しています」（金城氏）

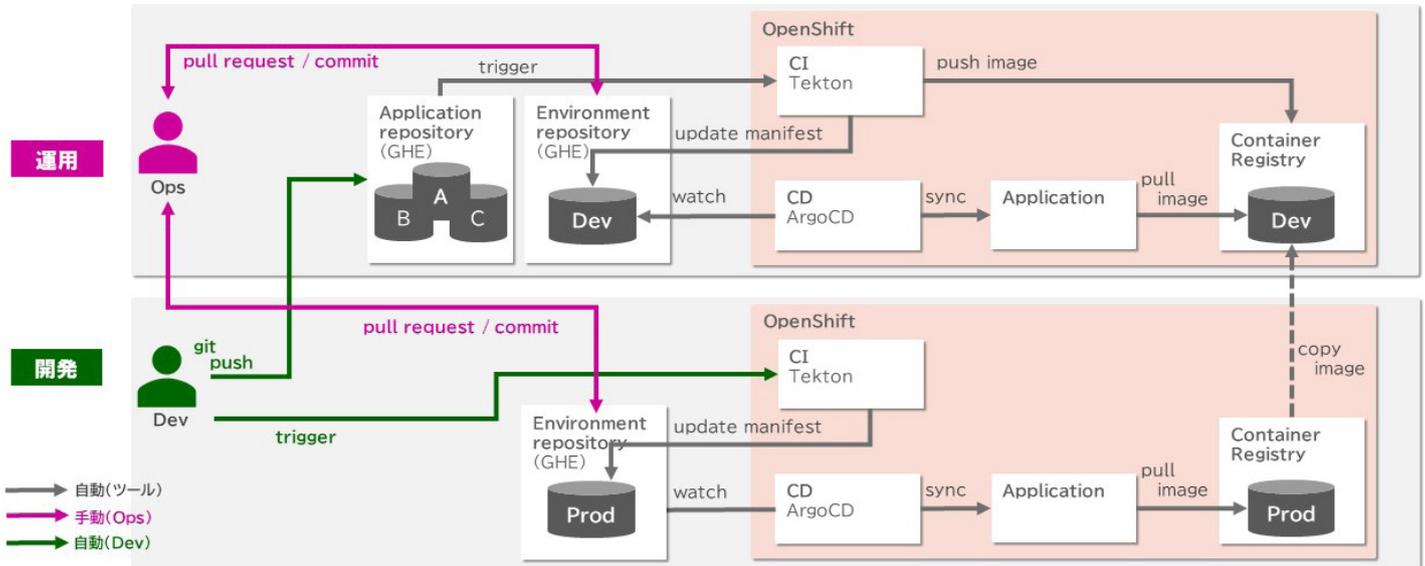
GitOps では、コード化されたクラスタの状態（マニフェスト）を Git リポジトリで管理し、GitHub の機能（プルリクエスト）を通じてアプリケーションをデプロイできる。また、CI/CD の仕組みを利用して、コンテナや Kubernetes の管理もできるようになる。いわば IaC（Infrastructure as Code）の進化形といえる。

大きな特徴は、GitOps を活用することでマニフェストの状態と、デプロイメント環境を一致させる「SSOT（Single Source of Truth：信頼できる唯一の情報源）」にできることだ。運用の実態でありがちなのが、あらゆる設定やマニフェストがその場その場で書き換えられていることで、コマンドラインなどで設定を直接変更すると履歴が残らない。そのため、後から問題が見つかった原因を探すのが難しくなってしまう。

そこで、マニフェストをリポジトリで管理する仕組みにするというわけだ。金城氏は「リポジトリで管理するメリットは、何か起きたときにでもサービスへの影響を最小限に抑えて、ロールアウトですぐに元の環境に戻せることです」と話す。言い換えれば「コマンドラインによる手作業をやめる」ということ。つまり、属人的な作業や作業履歴が残らないタスクを撤廃することで、IT 基盤全体の安全性を高めていくことにつながる。

提案： OpenShift GitOps/Pipeline

- コード化されたクラスタの状態(Manifest)をgitリポジトリで管理することで、変更時のレビューや承認・履歴の管理をgitに集約
- コマンドラインによる運用を撤廃し、属人的な作業・履歴の残らない作業を撤廃
- 継続的インテグレーション(CI)と継続的デリバリー(CD)を分離することで、担当者や責任範囲の明確化と適切な権限付与が可能



OpenShift GitOps / Pipelines による開発の流れ (提供：日立製作所)

また、GitOps では CI/CD を分離することで、担当者や責任範囲の明確化と適切な権限を付与できる。これは、GitOps を構成するコンポーネントの一つ「ArgoCD」が実現する。

ArgoCD は OpenShift GitOps で提供される CD ツールで、Git リポジトリで管理されたマニフェストを継続的に監視し、変更を検知すると自動で差分をクラスタに同期する。さらに「Kustomize」や「Helm」などのテンプレートも利用可能で、GUI や CLI でマニフェストの適用状態を確認できる。金城氏は「権限が外に出ないため、セキュリティリスクを軽減できます」と説明する。

OpenShift の利用は、特定のクラウドベンダーにロックインされないこともメリットの一つとなる。どの大手クラウドベンダーでもマネージド Kubernetes サービスと連携する CI/CD ツールを提供しているが、どこかのタイミングで Kubernetes 環境をオンプレミスや別のクラウドベンダーに移行しようとする、連携していた CI/CD ツールなどを再構築する必要があるため、身動きが取りにくくなってしまふ。その点、OpenShift はオンプレミスに限らず、大手

クラウドベンダー各社でも利用可能なため、運用そのもののポータビリティに優れるというわけだ。

北山氏は最新情報として「最近、GitOps を構成する CI ツールの『Tekton』、『ArgoCD』が GA (正式版) となり、より安全かつ安定した状態で提供できる状態になりました」と話す。

オープンソース版をそのまま利用するとサポートが受けられないということもあるが、日立のようにテクニカルサービスを提供しているベンダーのサポートを受ければ、そうした問題を気にせずにオープンソースのメリットを享受できる。北山氏は「今後も OpenShift の機能強化に取り組み、クラウドネイティブに取り組む企業へのパートナーシップや支援を強化していきます」と話す。

金城氏は「これまで日立は、ミッションクリティカルなシステムを抱える多くの大手企業を支援してきた実績があります。その経験や知見を生かし、IT 基盤をセキュアにして迅速なサービス提供や運用効率化を目指すお客さまの課題解決を支援する製品やサービスを提供していきます」と抱負を語った。

●お問い合わせ

株式会社 日立製作所 サービスプラットフォーム事業本部

〒244-0817 神奈川県横浜市戸塚区吉田町 292

<https://www.hitachi.co.jp/Prod/comp/soft1/openshift/>

※この冊子は、TechTarget ジャパン (<https://techtarget.itmedia.co.jp/>) に 2021 年 8 月に掲載されたコンテンツを再構成したものです。
<https://techtarget.itmedia.co.jp/tt/news/2108/20/news04.html>

copyright © ITmedia, Inc. All Rights Reserved.