

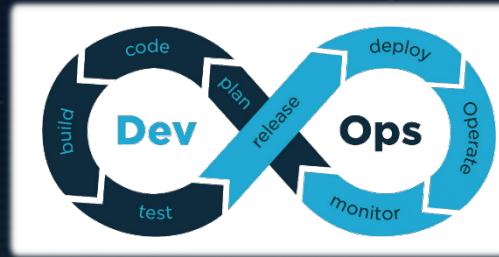
# Exastroで実現するDevOpsの開発効率化 (EPOCH × k8s)



# アジェンダ

- はじめに 約30分
  - クラウドネイティブ時代のアプリケーション開発とは？
    - DevOpsとは？
    - コンテナとは？Kubernetesとは？
  - Exastro EPOCHのご紹介
- Exastro EPOCHを活用したDevOpsデモ 約60分

## DevOpsとは



ソフトウェアの開発担当と導入・運用担当が密接に協力する体制を構築し、ソフトウェアの導入や更新を迅速に進めること。  
“Development”（開発）と“Operations”（運用）の略語を組み合わせた造語。

IT用語辞典より

- ✓ テクノロジーの進化によって、あらゆるものを取り巻く環境が複雑さを増し、**将来の予測が困難なVUCA(※)**の時代になっている
- ✓ 他社との差別化・追従に**サービス提供にスピード**が求められている。

※ VUCA:  
V (Volatility : 変動性)  
U (Uncertainty : 不確実性)  
C (Complexity : 複雑性)  
A (Ambiguity : 曖昧性)

実現にはクラウドネイティブの実践が必要！

クラウドネイティブのカラクリとは？  
システム初期提供時にどんなカラクリを仕込んでおけば、  
気持ちよく DevOps をまわせる？

## クラウドシフトで期待できること



最終的な目的  
ITリソースを活用し  
利益を向上する

### ROI(投資利益率)の引き上げ

投資の質を上げて  
売上につなげる

アプリケーションの  
生産性アップ

高速・スケーラブルな  
リリース

### OPEXの効率化

OPEX(運用費)を  
下げて投資にまわす

運用に必要な  
人件費の削減

運用に必要な  
保守費の削減

### 11のカラクリ (10の技 + 仮想化)

#### クラウドネイティブの実現手段

CI(継続的インテグレーション)

CD(継続的デリバリー)

マイクロサービスアーキテクチャ

サービスメッシュ

ブルーグリーンデプロイメント  
(immutable infrastructure)

宣言的API

オートスケーリング

分散トレーシング

メトリクス監視

メンテナスタスクの一元管理・自動化

仮想化(ハードウェア寿命からの解放)

## クラウドシフトで期待できること

11のカラクリ  
(10の技+仮想化)

### ROI(投資利益率)の引き上げ

投資の質を上げて  
売上につなげる

アプリケーションの  
生産性アップ

高速・スケーラブルな  
リリース

CI(継続的インテグレーション)

CD(継続的デリバリー)

マイクロサービスアーキテクチャ

サービスメッシュ

ブルーグリーンデプロイメント  
(immutable infrastructure)

宣言的API

最終的な目的

ITリソースを活用し  
利益を向上する

OPEXの効率化

OPEX(運用費)を  
下げて投資にまわす

特にこの6つの手段が  
「ROI(投資利益率)の引き上げ」という  
クラウドネイティブ最大のテーマに直結する

保守費の削減

オートスケーリング

分散トレーシング

仮想化(ハードウェア寿命からの解放)

# コンテナとは？

- ✓ Dockerに代表される仮想化技術で、非常に軽量なリソースで稼働できる



# なぜコンテナが必要なのか？

✓誤解されがちだが、コンテナはコスト削減のためのソリューションではない

コンテナとは：ビジネス・スピードを向上させる  
新しい仮想化テクノロジ

	デプロイ時間	主な導入目的
仮想マシン	数分～数日	コスト削減
コンテナ	数ミリ秒～数秒	ビジネス・スピード向上

コンテナ採用のビジネス目的トップ3

- 56% アジャイル開発によるサービス・レベルの向上
- 51% デジタル・ビジネス、DXの推進
- 48% タイム・トゥ・マーケットの改善

Source: [Market Guide for Container Management](#)

5 © 2021 Gartner, Inc. and/or its affiliates. All rights reserved. Gartner is a registered trademark of Gartner, Inc. and its affiliates.

Gartner

<https://atmarkit.itmedia.co.jp/ait/articles/2107/30/news032.html>

# クラウドネイティブに欠かせないコンテナ・Kubernetesとは？



クラウドネイティブ

## コンテナ技術に必要な要素とは？

- ✓ Dockerコンテナ単体のビルドは容易だが、Dockerのみでは複数のコンテナでシステムを組み上げられない
- ✓ 複数のホストへ展開してスケールアウトし、ロードバランシングする仕組み

→コンテナオーケストレーション

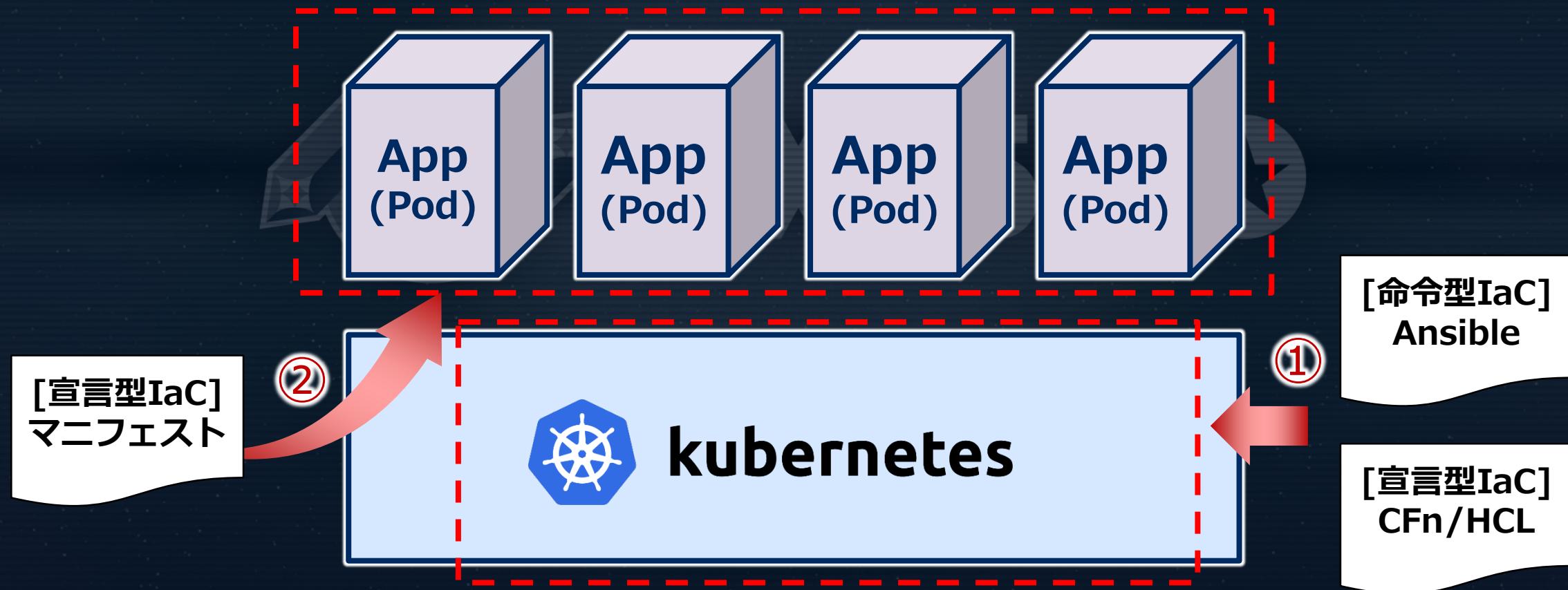


**kubernetes**

# Kubernetesを運用管理するために必要なIaCは？

✓コンテナ基盤の維持管理には2つのIaC(Infrastructure As a Code)が必要となる

- ① Kubernetes自体の構築やメンテナンスのための構築コード
- ② Kubernetes上にコンテナ(Pod)をデプロイするための“マニフェスト”



# IaCはどのように運用するのか

## ✓ CI/CDを活用しましょう

アプリケーションの開発～リリースまでの一連の作業を自動化し、アプリケーション提供の頻度を高める手法です。

特に先頁の②（マニフェスト）のIaCに対しては、高頻度の変更(AP変更毎)が発生します

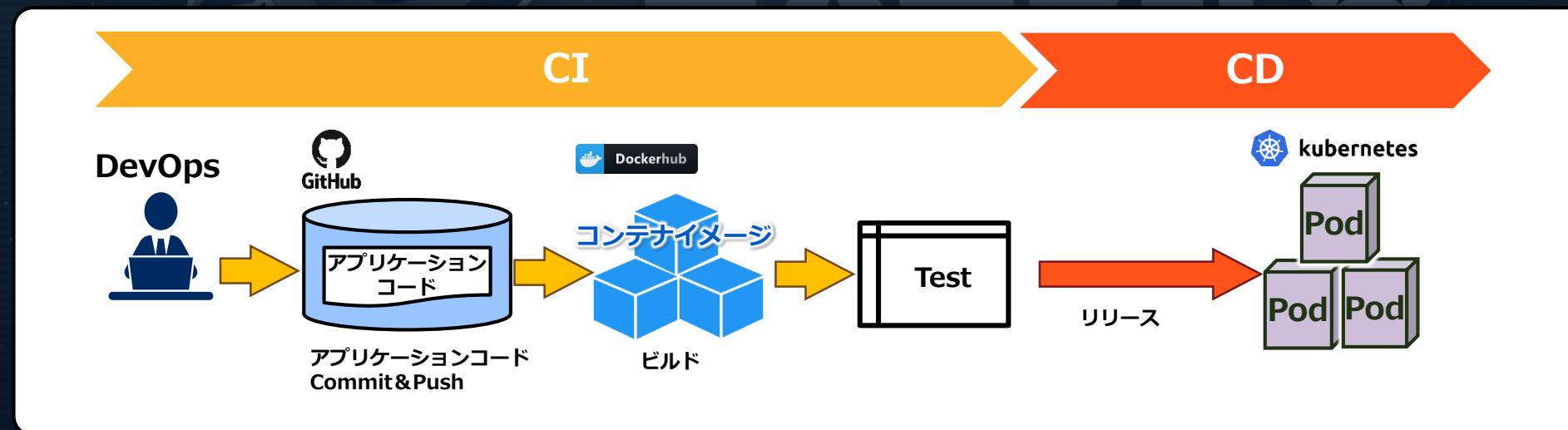
## ✓ CI（継続的インテグレーション）

アプリケーションコードの変更を起点に、ビルドやテストの実行といった開発者の作業を自動化する手法を指します。

## ✓ CD（継続的デリバリー）

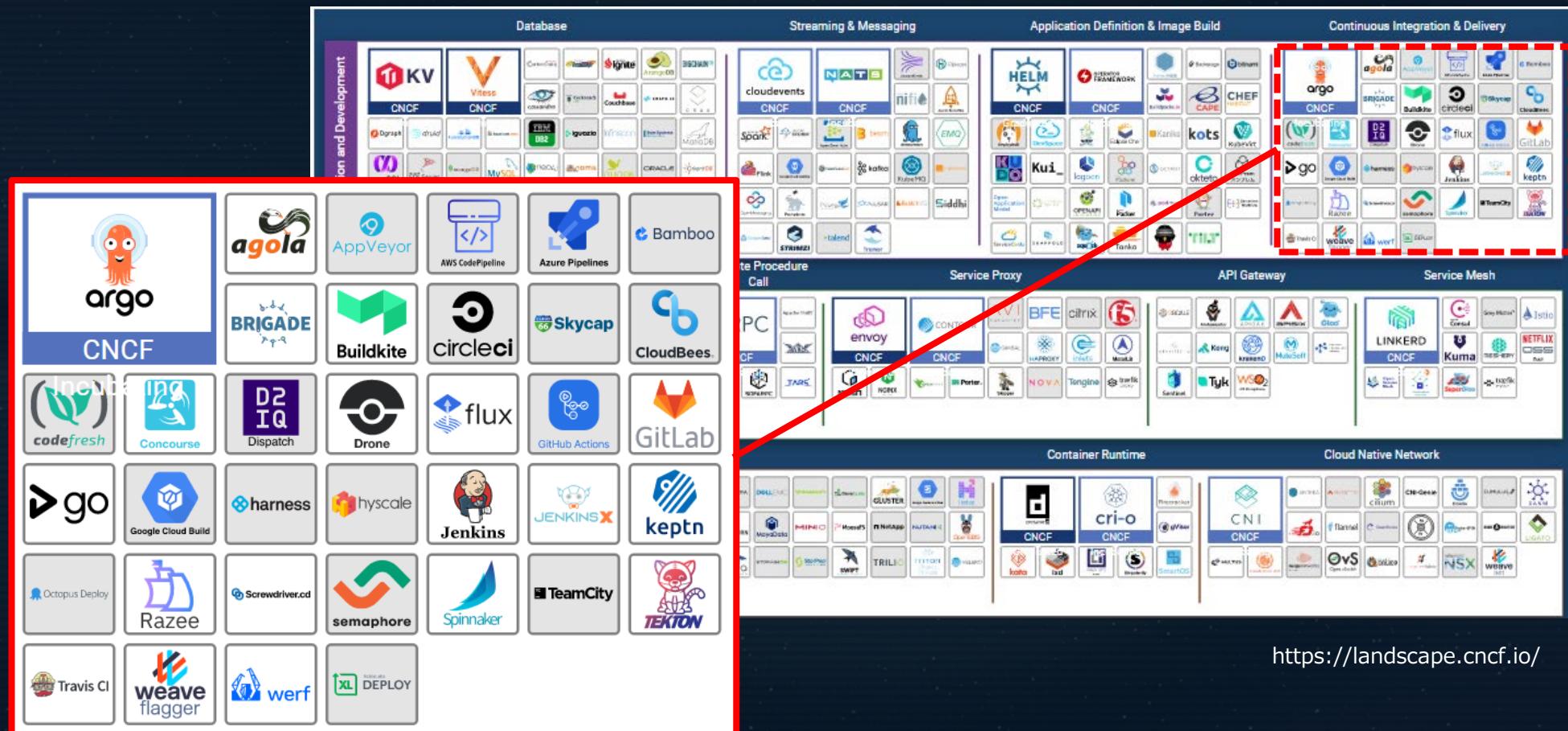
実行環境へのリリースまでを自動化する手法を指します。

## ✓ CI/CDのイメージ



## CI/CDツール

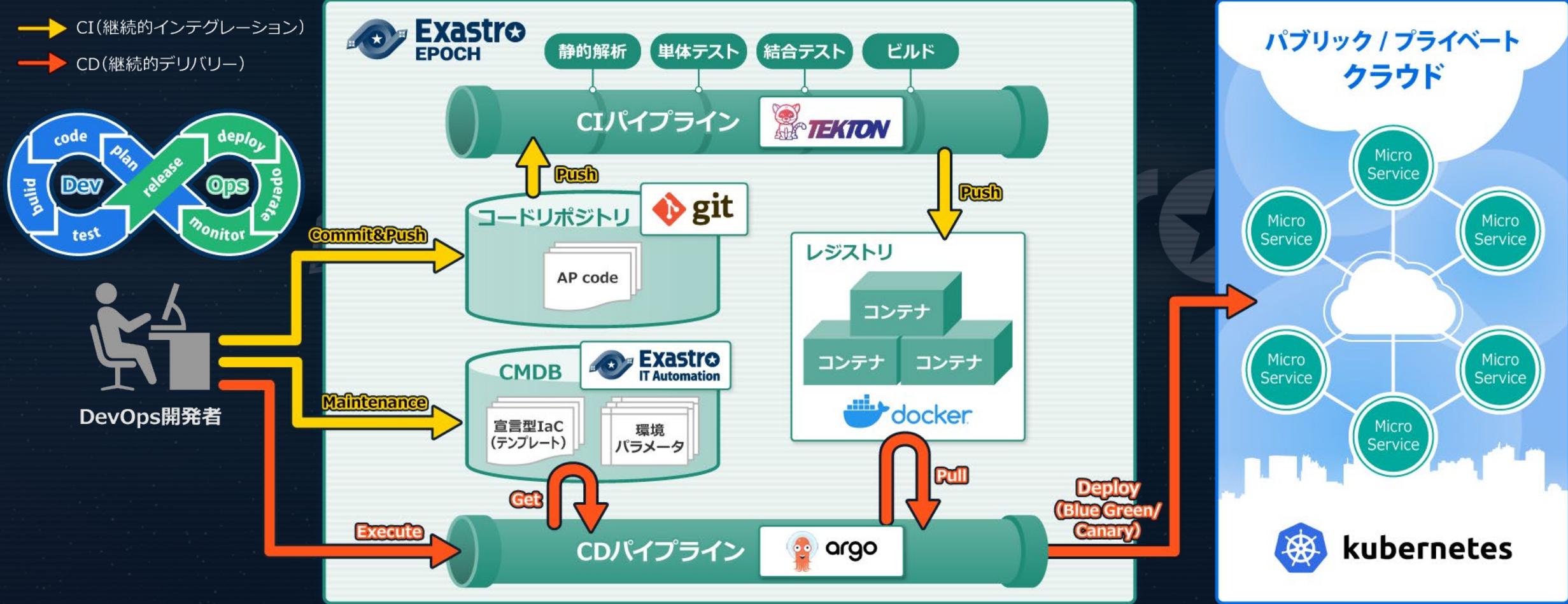
✓ 様々なCI/CDツールが存在し、短期間での選定・準備には苦労します。



- ✓ 新技術要素が多く難易度が高い・・・
- ✓ 幅広いスキルやノウハウが求められる・・・

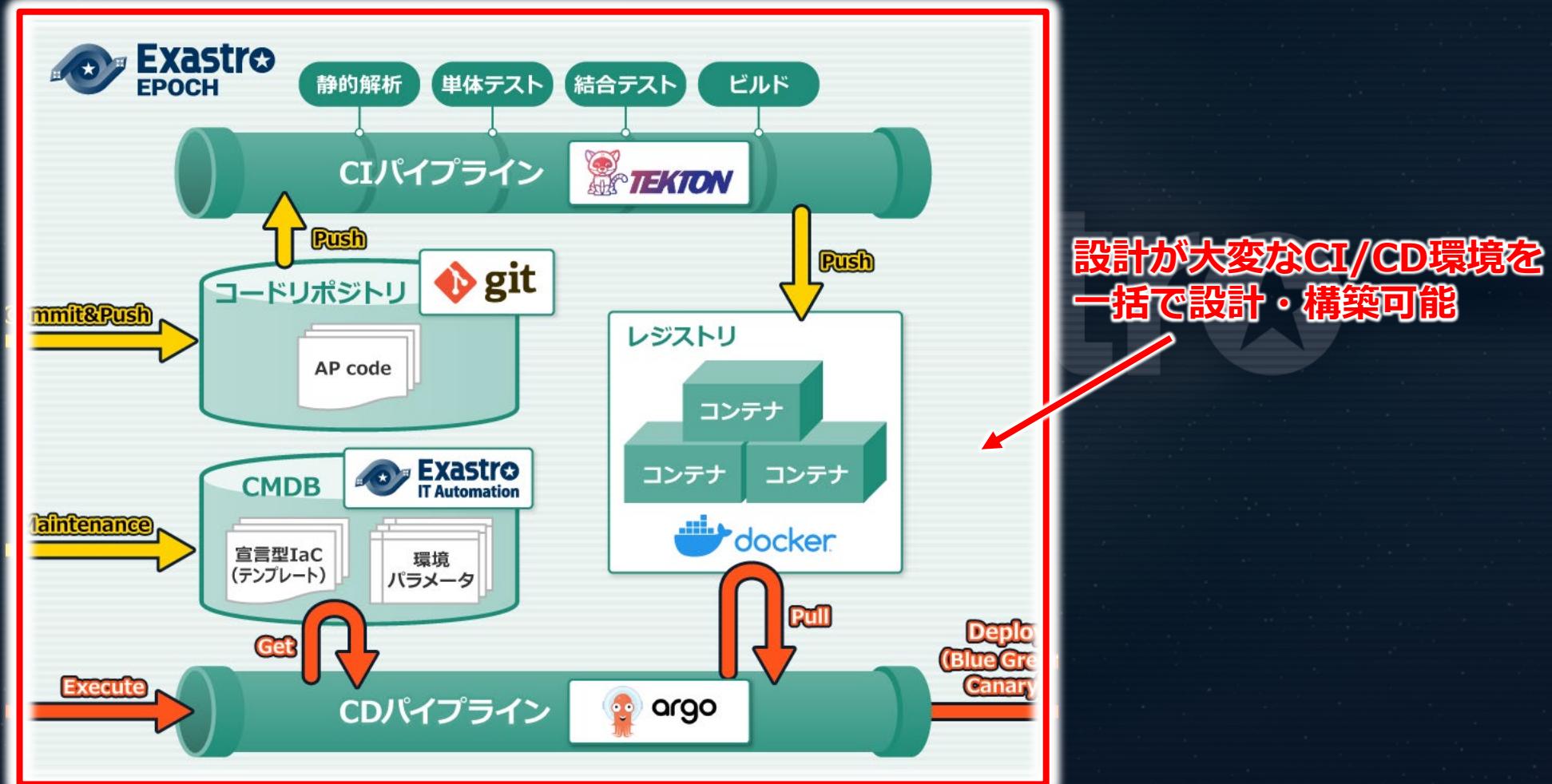


## Exastro EPOCHは 「クラウドネイティブシステム開発を加速するためのフレームワーク」です



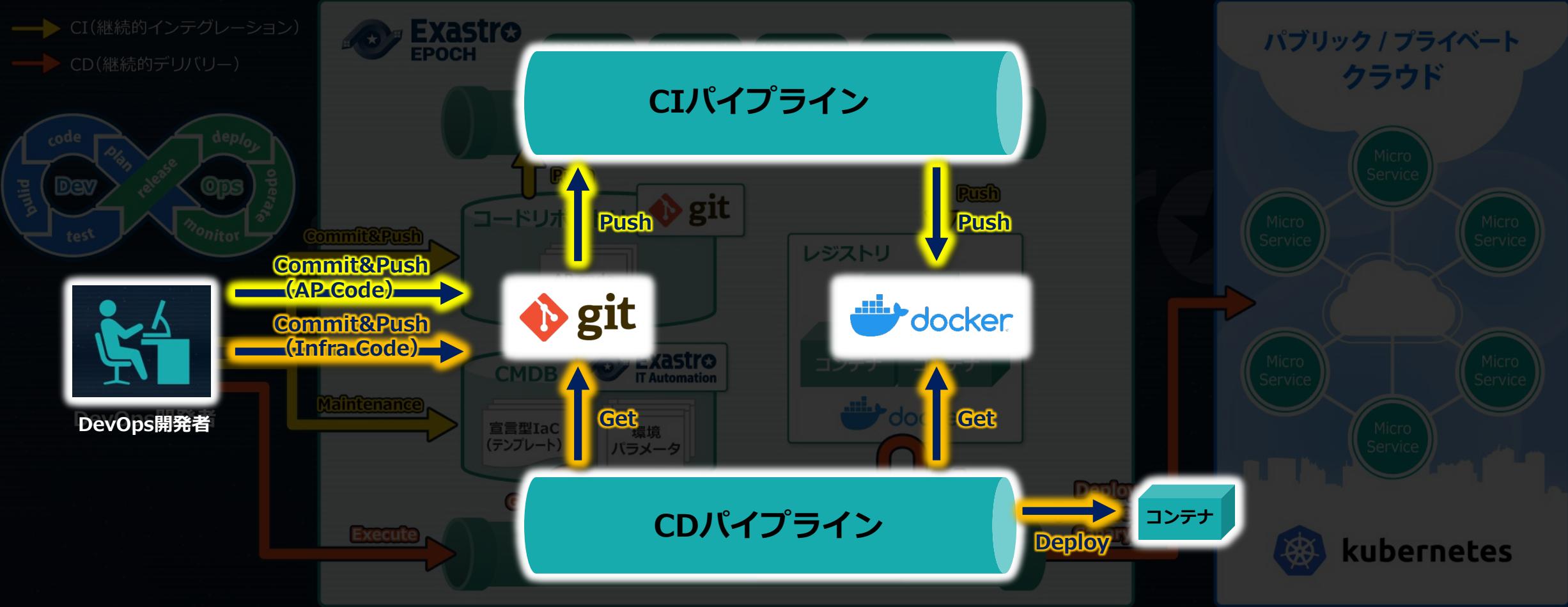


## 高速開発に必要なCI/CD環境を一元的に提供する



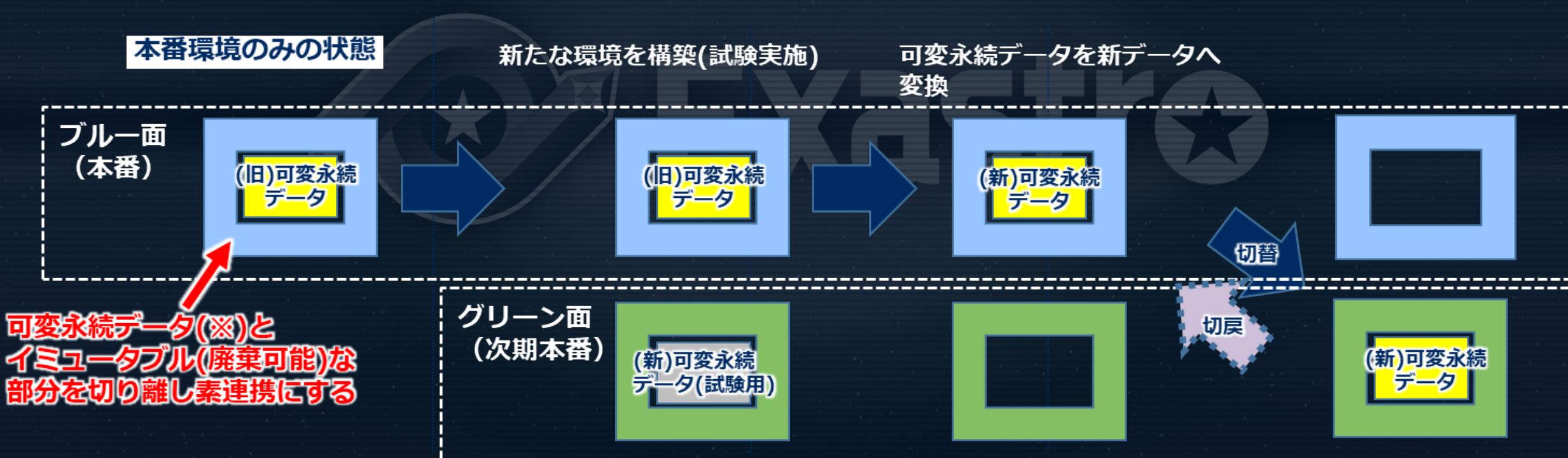
## 2つめの特徴 – GitOpsに対応した環境の展開

アプリケーションコードのコミット&プッシュだけで、CI/CDパイプラインを実行可能 (GitOps)



### 3つめの特徴 - カナリアリリース/ブルーグリーンデプロイに対応

コンテナデプロイでは迅速な切り替え・切り戻しに有用な“カナリアリリース”と“ブルーグリーンデプロイ”が利用可能  
(例: ブルーグリーンデプロイ)



## 4つめの特徴 - ワークロードを統一

CI/CDのステータスを各CI/CDツールを介さずに一括で確認可能  
スムーズな状況確認・DevOpsを実施



## 5つめの特徴 – 環境一致を強力に支援する

宣言的IaCの環境依存パラメータをシステム上で管理  
CD時に自動的にパラメータ反映したIaCをコミットする

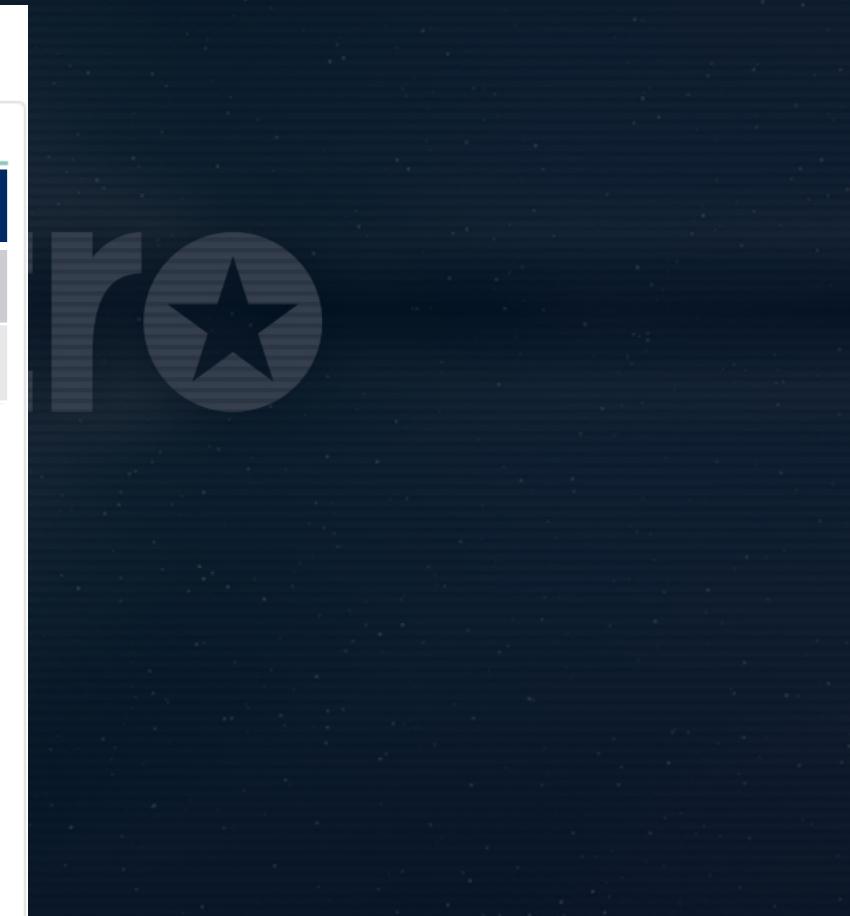
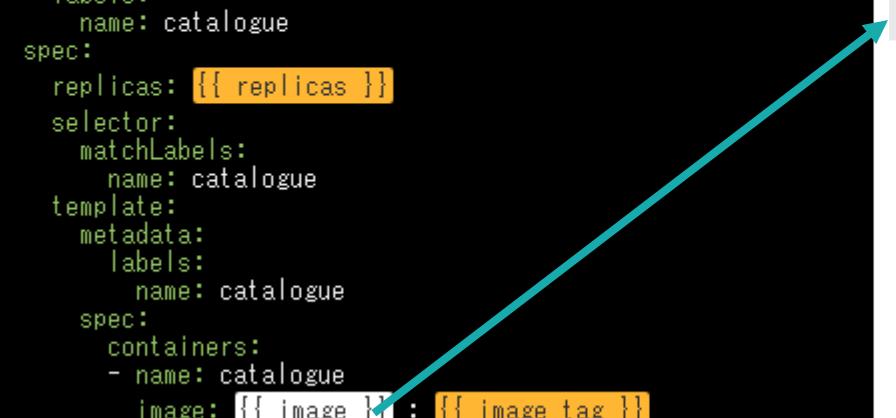
パラメータ入力

text.yaml

```
# epoch-template => file001
apiVersion: apps/v1
kind: Deployment
metadata:
  name: catalogue
  labels:
    name: catalogue
spec:
  replicas: {{ replicas }}
  selector:
    matchLabels:
      name: catalogue
  template:
    metadata:
      labels:
        name: catalogue
    spec:
      containers:
        - name: catalogue
          image: {{ image }} : {{ image_tag }}
          ports:
            - containerPort: 8000
      nodeSelector:
        beta.kubernetes.io/os: linux
```

環境

環境	Develop
環境①	パラメータA
環境②	パラメータB



✓ Exastro EPOCHを活用したDevOpsデモ

- ・インストール編
- ・DevOpsデモ

[https://exastro-suite.github.io/epoch-docs/learn\\_ja.html#quickStart](https://exastro-suite.github.io/epoch-docs/learn_ja.html#quickStart)



## Exastro EPOCH で DevOps に必要なプロセスや環境を一括で提供！

- ✓ DevOps でサービス提供がスピードアップ！
- ✓ 実現には様々な課題（ノウハウ・スキル）があるが、  
Exastro EPOCH の活用でショートカットが可能！



## Exastro EPOCH の関連情報を準備しています

✓ コミュニティサイト

[https://exastro-suite.github.io/epoch-docs/index\\_ja.html](https://exastro-suite.github.io/epoch-docs/index_ja.html)

✓ Issue

<https://github.com/exastro-suite/epoch/issues>



**Exastro** 