

vRealize Automation Code Stream スタート ガイド

2019 年 12 月 19 日

vRealize Automation 8.0

最新の技術ドキュメントは、VMware の Web サイト (<https://docs.vmware.com/jp/>)

VMware, Inc.
3401 Hillview Ave.
Palo Alto, CA 94304
www.vmware.com

ヴィエムウェア株式会社
105-0013 東京都港区浜松町 1-30-5
浜松町スクエア 13F
www.vmware.com/jp

Copyright © 2021 VMware, Inc. All rights reserved. [著作権および商標情報](#)。

目次

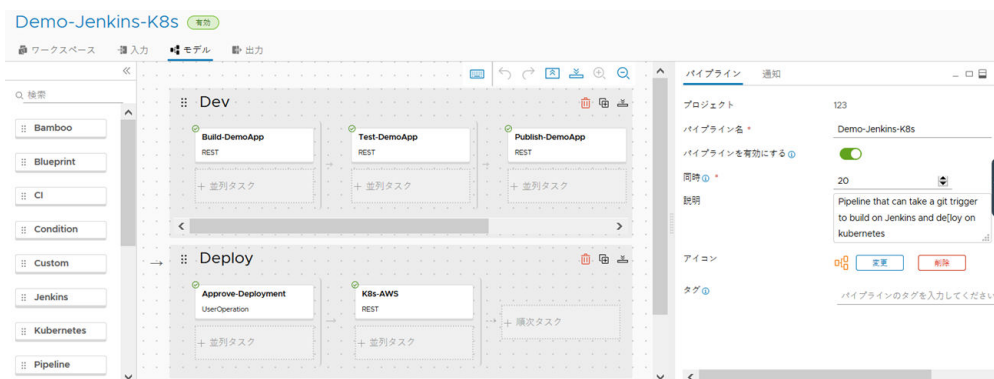
- 1** vRealize Automation Code Stream とは 4
- 2** vRealize Automation Code Stream の機能 5
- 3** はじめに 8
- 4** セットアップする方法 10
- 5** その他の機能 13

vRealize Automation Code Stream とは

1

vRealize Automation Code Stream™ は、連続統合および継続的デリバリー (CI/CD) ソフトウェアであり、ソフトウェアを迅速かつ確実に配信できます。しかも、オーバーヘッドがほとんどありません。vRealize Automation Code Stream は、モノリシック レガシー アプリケーションの展開をサポートし、複数のクラウドで Docker および Kubernetes コンテナを実行できます。

vRealize Automation Code Stream を使用すると、Git や Jenkins などの既存の開発ツールを使用しながら、DevOps のライフサイクル全体を自動化するパイプラインを作成できます。



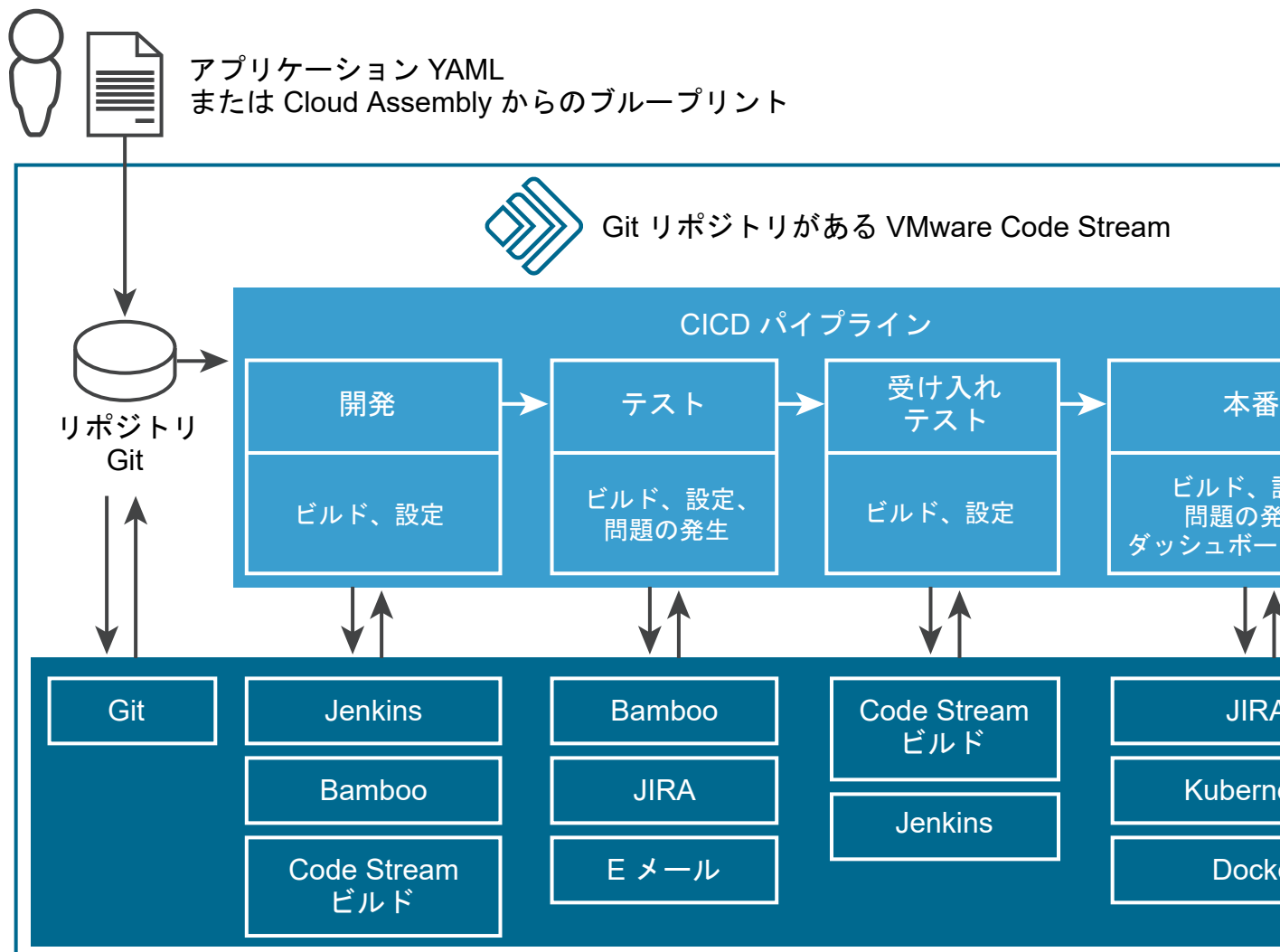
vRealize Automation Code Stream を使用すると、ソース コードが開発リポジトリからテストを経て本番環境に至るため、アプリケーションのビルド、テスト、展開を簡素化し、生産性を高めることができます。vRealize Automation Code Stream は、カスタム アプリケーションと商用アプリケーションをサポートするほか、ブループリントなどのオブジェクトもサポートします。

vRealize Automation Code Stream の機能

2

vRealize Automation Code Stream は、ソフトウェア リリース プロセスのタスクをモデル化し、開発者コードの開発とテストを自動化して、本番環境にリリースします。

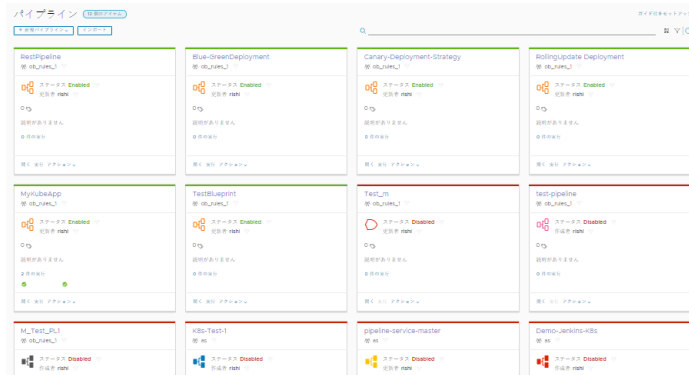
リリース プロセスを開発者ツールと統合して特定のタスクを実行し、コードのアーティファクトとバージョンをすべて追跡します。



ソフトウェアをビルド、展開、テスト、およびリリースするためのアクションを実行するパイプラインを作成します。vRealize Automation Code Stream は、本番環境へのリリース準備が整うまで、パイプラインの各ステージを通じてソフトウェアを実行します。

パイプラインをデータ ソースやリポジトリや通知システムなど 1 つ以上の DevOps ツールと統合して、パイプラインの実行に必要なデータを提供します。

たとえば、Kubernetes アプリケーションをダウンロードして GitHub から Kubernetes クラスタに展開するパイプラインがあって、そのパイプラインのエンドポイントをいくつか使用しなければならないことがあります。



vRealize Automation Code Stream は、さまざまなエンドポイント タイプと統合されています。

表 2-1. vRealize Automation Code Stream と DevOps ツールとの統合

DevOps エンドポイント	機能
Git	開発者がコードをチェックインしたら、リポジトリから開発者コードをプルし、Git トリガと連携してパイプラインをトリガします。
Kubernetes	コンテナ化されたアプリケーションを展開、スケール、および管理するための手順を自動化します。
vRealize Automation Code Stream ビルド	サードパーティ統合を使用するのではなく、連続統合用のネイティブ ビルドを作成します。
Jenkins	コード アーティファクトをビルドします。
E メール	ユーザーに通知を送信します。
JIRA	パイプラインが機能しなかったときにチケットを作成します。
Bugzilla	バグを作成および追跡します。

vRealize Automation Code Stream を使用する理由

開発者または DevOps 管理者として、既存の開発ツールを引き続き使用しながら、vRealize Automation Code Stream を使用して DevOps リリースのライフサイクル全体を自動化できます。vRealize Automation Code Stream の特徴は次のとおりです。

- 自動化が容易
- オープン ソースなしでプラグインをすぐに使える
- モデリングが容易でパイプラインをコード化できる

- VMware Cloud PKS との統合が容易
- レポートを作成して判断材料を得ることができる
- 詳細なダッシュボードで完全な可視化を実現できる
- ダッシュボードをカスタマイズできる
- DevOps メトリックと判断材料
- ガバナンス
- ロールベースのアクセス
- 非公開で制限された変数、および承認
- プロジェクト

vRealize Automation Code Stream では、ソース コードを開発段階とテスト段階で実行してから本番環境にリリースするため、ソフトウェア アプリケーションの展開が簡素化されます。また、Git トリガを使用しているので生産性が向上します。開発者がコードをチェックインすると、vRealize Automation Code Stream がパイプラインをトリガし、アプリケーションのビルド、テスト、および展開を自動化します。

vRealize Automation Code Stream は、他の vRealize Automation コンポーネントとともに使用できます。

- vRealize Automation Cloud Assembly ブループリントを展開し、そのブループリントが公開するパラメータ値を使用します。
- vRealize Automation Service Broker にパイプラインを公開して、チーム メンバーがそのパイプラインを各自のクラウド リージョンに展開できるようにします。

vRealize Automation Code Stream のその他の使用方法については、[5 章 vRealize Automation Code Stream のその他の機能](#)を参照してください。

vRealize Automation Code Stream の使用を始める前に

3

vRealize Automation Code Stream の使用を開始するには、DevOps 管理者または開発者として My VMware アカウントにサインアップし、ログインする必要があります。

ログイン

ログインすると、パイプラインを作成、実行、および管理できます。

表 3-1. vRealize Automation Code Stream を使用する準備

手順	アクション	必要な操作
1	My VMware アカウントを作成します。	https://my.vmware.com/ で会社のメール アドレスを使用して My VMware アカウントを作成します。
2	vRealize Automation にログインします。	送信トラフィックがファイアウォールを介して次の宛先にアクセスできるように HTTPS ポート 443 を開きます。 <ul style="list-style-type: none">■ *.vmwareidentity.com■ gaz.csp-vidm-prod.com■ *.vmware.com
3	vRealize Automation Code Stream サービスを開きます。	

DevOps 管理者のためのスタート ガイド

DevOps 管理者には Administrator ロールがあり、開発者が作業インスタンスを使用できるようにエンドポイントを作成することができます。パイプラインの作成、管理、トリガなどを行うことができます。

たとえば、開発者は場合によってはパイプライン タスクをデータ ソース、リポジトリ、または通知システムに接続する必要があります。これらのコンポーネントは、パイプラインを実行するためのデータを提供します。

また、vRealize Automation Code Stream を他の vRealize Automation コンポーネントと統合することもできます。

- ブループリントを展開するには、vRealize Automation Cloud Assembly を使用します。
- パイプラインを公開し、トリガするには、vRealize Automation Service Broker を使用します。

開発者のためのスタート ガイド

開発者がパイプラインをビルドおよび実行し、ダッシュボードでパイプライン アクティビティを監視するには、vRealize Automation Code Stream を使用します。User ロールが必要です。

パイプラインを実行したら、次のことを確認してください。

- パイプラインのどの段階でもコードが正常に完了している。結果が [実行] に表示される。
- パイプラインが機能しなかった場合、原因は何か。キー エラーが [ダッシュボード] に表示される。

スマート テンプレートの使用

アプリケーションをネイティブにビルド、テスト、および展開するパイプラインを作成するときには、スマート テンプレートを使用すると時間を節約できます。テンプレートごとに質問がいくつか用意されており、その回答に基づいてパイプラインが作成されます。

- ビルド目的、環境、およびソース コードの場所。
- 展開目的、およびアプリケーションの展開先。たとえば、スマート テンプレートは既存の Kubernetes クラスターを識別します。これにより、アプリケーションのビルドおよび展開の際に使用するクラスターを選択できます。

テンプレートを利用してパイプラインを作成した後、さらにニーズに合うようにパイプラインを変更できます。

vRealize Automation Code Stream をセットアップする方法

4

vRealize Automation Code Stream をセットアップする管理者は、ログイン後、エンドポイントの追加、パイプラインの作成と実行、および結果の表示を行うことができます。

vRealize Automation Code Stream はエンドポイントに接続し、パイプラインを実行するためのデータを取得します。この使用事例では、VMware Code Stream を GitLab リポジトリに接続し、パイプラインで Kubernetes ファイルをダウンロードできるようにします。

スタート ガイド プロセスは、vRealize Automation Code Stream ユーザー インターフェイスのガイダンスありのセットアップとしても利用できます。[ヘルプ] アイコンをクリックして、[ガイダンスありのセットアップ] をクリックします。

前提条件

- オンプレミスの GitLab または GitHub リポジトリが配置され、パイプラインで使用されるコードが含まれていることを確認します。

手順

- 1 vRealize Automation Code Stream をオンプレミスの GitLab リポジトリに接続するための Git エンドポイントを追加します。
 - a [エンドポイント] をクリックします。
 - b Git エンドポイント タイプを選択し、名前と説明を入力します。
 - c その他の情報を入力します。
 - d エンドポイントへの接続をテストするには、[検証] をクリックし、エンドポイントを保存します。

- 2 [パイプライン] をクリックしてパイプラインを作成し、Git エンドポイントを使用するタスクを追加します。必要に応じて E メール通知を追加できます。



- 3 パイプラインを保存し、[有効] をクリックして、パイプラインを実行できるようにします。



- 4 パイプライン カードが有効である場合は、[実行] をクリックします。
- 5 [実行] をクリックして、パイプラインの実行を監視します。



- 6 パイプラインが機能しなかった場合は、問題を修正してから再度実行します。
- 7 [ダッシュボード] をクリックし、パイプライン アクティビティを監視できるようにパイプライン ダッシュボードを選択します。

結果

パイプラインが実行され、GitLab インスタンスから開発者ファイルがダウンロードされました。パイプライン タスクにより、アプリケーションが Kubernetes クラスタに展開され、パイプライン ダッシュボードでアクティビティすべてを監視しました。

次のステップ

ここで必要な情報が見つからない場合は、製品内にあるヘルプを参照できます。



- ユーザー インターフェイスの Signpost およびツールチップをクリックして表示し、任意のタイミングと場所でコンテキスト固有の必要な情報を取得します。
- [製品内サポート] パネルを開き、[アクティブなユーザー インターフェイス] 画面に表示されるトピックを参照します。パネルで検索して、質問に対する回答を取得することもできます。

vRealize Automation Code Stream のその他の機能

5

DevOps 管理者または開発者は、vRealize Automation Code Stream を使用して、DevOps のリリース ライフサイクルのステップをモデル化して自動化する、より複雑なパイプラインを作成できます。

詳細を知りたい内容	使用と管理についての参照先
概要と実行可能な操作	概要と仕組み
リリース プロセスのモデリング	リリース プロセスをモデリングするための設定
CICD、CI、または CD ネイティブ ビルドの計画とパイプラインの作成	コードのネイティブな構築、連携、配布の計画
使用事例	コードを my GitHub または GitLab リポジトリから自分のパイプラインに継続的に統合する方法 その他

vRealize Automation Code Stream で実行できる操作の詳細については、vRealize Automation Code Stream のヘルプ パネルにある製品内ヘルプを参照してください。



製品内ヘルプを使用することで、以下の機能などについて理解できます。

- エンドポイントとの通信
- 制限されたリソースおよび変数の操作
- プロジェクトの操作
- 開発者がコードを更新するときのパイプラインのトリガ
- カスタム ダッシュボードにおけるパイプラインに対する KPI の追跡