

RED HAT OPENSIFT APPLICATION RUNTIMES でクラウドネイティブ開発を効率化

主なメリット:

- モノリスからマイクロサービスへの移行中にクラウドネイティブ・アプリケーションの作成を開始することが極めて容易になる
- 既存の専門知識を活かし、厳選されたランタイムとフレームワークによりクラウド開発における課題を解決する
- クラウド・インフラストラクチャでもオンプレミス環境でも利用可能

はじめに

クラウドネイティブ・アプリケーションとマイクロサービスを作成するメリットが明確であっても、いくらかのコードを実行できるという段階から本番稼働可能な成果物の提供までには、多くの場合、ある程度の時間が必要となります。アプリケーションをクラウドに配置し、マイクロサービスに移行すると、互いに接続された要素が増えるため、複雑さも増します。接続速度が低速な場合、未処理の要求の数が増え続けるため使用可能なリソースが枯渇し、診断が困難なカスケード障害が発生するおそれがあります。さらに、レジリエンシー（回復力）やセキュリティなど、インフラストラクチャに関する懸念事項が存在すると、具体的なビジネス価値を提供するコードを書くための時間が奪われる可能性もあります。

レジリエンシーと運用上の応答性を念頭に置く場合、マイクロサービスを一から新規開発することも可能です。ただし、大半の組織は既存アプリケーションに依存しています。これらのアプリケーションは、最新のフレームワークと言語を使用して書き換えるにはコストがかかりすぎるものです。また、既存のアプリケーションをモダナイズする用途と、新たなマイクロサービスを作成する用途で、それぞれ異なるソリューションの導入を検討する組織も珍しくありません。しかし、モダナイズと新規作成という2つのアプローチに何の共通点もなければ、非効率的となる可能性があります。

開発者が必要とするものは、以下のとおりです。

- アプリケーション・コードとともにコンテナに組み込む準備ができているランタイム、フレームワーク、およびインフラストラクチャのコンポーネント
- クラウド上でビルドとデプロイが可能なランタイムの使用法を示す具体例
- 作業に適したツール: 既存アプリケーションと新規マイクロサービスの両方に応えるソリューションを備えた、複数のフレームワークと言語
- 複数のクラウドだけでなく、オンプレミス環境にもデプロイ可能なソリューション

Red Hat® OpenShift Application Runtimes は、これらのニーズを考慮して設計されており、クラウドネイティブ・アプリケーションの開発と提供を効率化します。

RED HAT OPENSIFT APPLICATION RUNTIMES

OpenShift Application Runtimes は、先進的でクラウドネイティブな、マルチクラウド対応のアプリケーション・ランタイムを提供します。これらのコンテナ化されたランタイムは、規範的な開発者エクスペリエンスを提供するため、新しいクラウドネイティブ・アプリケーションの作成と既存アプリケーションのクラウド対応をすぐに始めることができます。複数のフレームワークに加えて、レジリエンシー、セキュリティ、および運用のためのインフラストラクチャ・コンポーネントが統合されており、本稼働対応のコードの提供をサポートします。OpenShift Application Runtimes は、オープン・ハイブリッドクラウド用に設計されたコンテナ・アプリケーション・プラットフォームである Red Hat OpenShift をベースとしており、開発とデプロイのための極めて効率的なプラットフォームを実現します。



facebook.com/redhatjapan
@redhatjapan
linkedin.com/company/red-hat

Red Hat OpenShift Application Runtimes の特長：

- 規範的：マイクロサービスの作成や、既存アプリケーションのリフト & シフト形式でのクラウド移行など、開発上の具体的な課題向けに推奨されるソリューションとして、厳選されたランタイムとフレームワークが採用されています。
- 複数フレームワーク、言語非依存：ブラウザやモバイルクライアントのリアクティブサービスの提供から、従来型のトランザクション処理の実行に至るまで、作業内容に適したツールを柔軟に使用できます。
- 既存の専門知識を活かせる設計：多くの組織が Java™ EE についての豊富な専門知識を有していることから、既存コードの再利用を可能とするランタイムオプションがいくつか用意されており、従来の Java EE からマイクロサービスまで幅広いアーキテクチャ・スタイルを網羅しています。
- 極めて簡単に使い始められる：OpenShift Application Runtimes のローンチサービスでは、任意のランタイムやサンプル・ユースケースに基づき、Red Hat Developers の Web サイトからオンラインで、実行可能かつ完全なプロジェクトを開始することができます。必要なコードはすべて、GitHub アカウントのプロジェクトにフォークされるか、zip ファイルとしてダウンロードできます。Maven を使用する場合は、プロジェクトの pom.xml ファイルに数行追加するだけで、Red Hat JBoss® Enterprise Maven リポジトリからランタイムを自動的に取得できます。
- テスト、検証、保守：OpenShift Application Runtimes のコンポーネントは、Red Hat によりテストおよび検証されているため、安心して利用することができます。Red Hat は、ランタイムに含まれるアップストリーム・オープンソース・プロジェクト群の重要なコントリビューターであるとともに、これらの多くのプロジェクトにおいてリーダー的役割を果たしています。一部のランタイムは、Red Hat JBoss Enterprise Application Platform (JBoss EAP) や Red Hat JBoss Web Server などの Red Hat 製品をベースとしています。Red Hat は、Red Hat JBoss Middleware のような他の Red Hat 製品を含む、OpenShift Application Runtimes の包括的なロードマップを提供しています。
- クラウド・インフラストラクチャを選ばず、DevOps 対応：任意のパブリックまたはプライベートクラウドで実行可能なアプリケーションを作成するための効率的な手段を提供します。アプリケーションは、セルフサービスの Web コンソールで構築、デプロイ、管理することができます。Red Hat OpenShift Online を使用すると、ソフトウェアや Red Hat Container Development Kit をインストールする必要なく操作を開始でき、ラップトップ端末上で完全なシングルノードの Red Hat OpenShift クラスタを実行できます。

Red Hat OpenShift Application Runtimes は、Red Hat OpenShift を使用して、開発だけでなくデプロイと運用を効率化します。Red Hat OpenShift は、大規模環境で分散システムの構築や実行を行う際の複雑さを単純化するために、多数の機能を提供します。提供される機能は以下のとおりです。

- サービスの検出、ルーティング、ロードバランシング：スケーリングやローリングデプロイのため、トラフィックを複数のインスタンスに分散します。
- 設定と機密情報の管理：コードベースと環境固有の情報を分離します。
- 一元的なロギング、メトリック、モニタリング：運用効率を向上します。

Red Hat OpenShift は開発者に、コンテナ化アプリケーションの構築と実行のためのセルフサービス・プラットフォームを提供します。Red Hat OpenShift を使用すると、新しいマイクロサービスのための環境をわずか数分で提供できます。Red Hat OpenShift では、数クリックの操作で強力な自動化されたビルド・パイプラインおよびデプロイメント・パイプラインを利用でき、本稼働に耐え得る継続的インテグレーションと継続的デリバリー (CI/CD) およびローリング・デプロイメントが実現します。デプロイの自動化機能は、マイクロサービスに移行し、より頻繁にリリースを行うにあたって、極めて重要となります。

Red Hat OpenShift の主要コンポーネントである Kubernetes は、Google が最初に開発を始めたコンテナ・オーケストレーション・プラットフォームです。Kubernetes は、業界におけるクラウド運用環境のデファクト・スタンダードであり、クラウドネイティブ・アプリケーションの作成と運用をクラウド規模で実現するためのインフラストラクチャ・サービスの多くを提供します。Red Hat OpenShift を使用しない場合、膨大な YAML ファイルを作成する必要があるため、Kubernetes の学習に必要な労力が非常に大きくなります。Red Hat OpenShift では、Kubernetes がもたらすあらゆるメリットを、開発者にも運用担当者にも使いやすい包括的な Web インタフェースにより利用できます。

OpenShift Application Runtimes を使用することで、わずか数分のうちに、必要な機能をすべて備えたエンタープライズ・グレードのクラウド・プラットフォーム上のコンテナ内で、コードの構築と実行を始めることができます。図1は、Red Hat OpenShift Application Runtimes の論理サービスレイヤーを示しています。

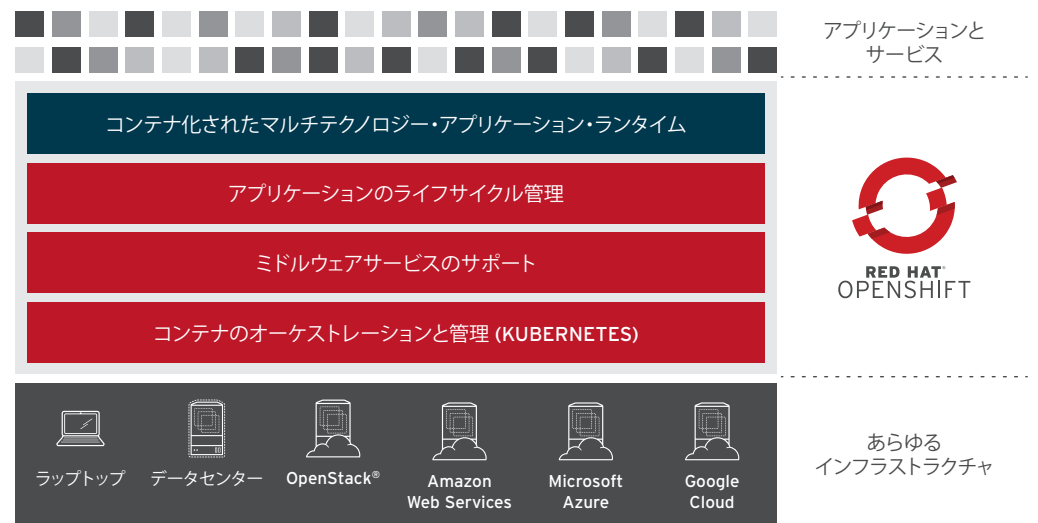


図1. Red Hat OpenShift Application Runtimes のサービスレイヤー

提供されるランタイム

Red Hat OpenShift Application Runtimes には、先進的なクラウドネイティブ・アプリケーションの提供に伴う課題を克服するための開発者支援に特化した一連のランタイムが含まれています。

- JBoss EAP**: 先進的なモジュール式のクラウド対応アーキテクチャをベースとした Java EE 7 認定プラットフォームであり、最小限の変更で既存の Java EE アプリケーションと Spring MVC アプリケーションのクラウド移行を実現するために理想的なソリューションです。JBoss EAP ランタイムは緊密に統合されており、Red Hat OpenShift のコンテナ機能を利用して、動的なスケールリング、構成、およびライフサイクル管理を単純化します。Red Hat OpenShift の機能を使用すると、アプリケーションを高速で動作するモノリスに移行することができ、デプロイやテストが容易になります。
- WildFly Swarm**: Java EE 開発者が慣れ親しんだ方法でマイクロサービスに移行するための手段を提供するほか、完全版の Java EE ランタイムよりも軽量な選択肢となります。Java EE のサブセットと Eclipse MicroProfile (エンタープライズ Java マイクロサービス用のオープンなコミュニティ指向の仕様) を組み合わせることで、アジリティに優れた標準ベースのプラットフォームが実現します。WildFly Swarm は、Java EE アプリケーションとマイクロサービスのビルドと実行に関する革新的なアプローチを提供します。このアプローチでは、Java EE アプリケーションとマイクロサービスを必要最小限のサーバー領域とともにパッケージ化し、`java -jar` コマンドだけで実行できるアプリケーションを作成します。これは、WildFly Application Server をフラクショナルと呼ばれるコンポーネント群に分解することによって実現します。実際にアプリケーションに必要なフラクショナルのみが、最終の JAR ファイルにアセンブルされます。WildFly Swarm は、既存の Java EE モノリスからマイクロサービスへ機能を移行したり、既存の Java EE の専門知識を使用して新しいマイクロサービスを開発するために役立ちます。
- Eclipse Vert.x**: Java 仮想マシン (JVM) 上でリアクティブ・マイクロサービスを構築するために設計されています。リアクティブ・プログラミング・モデルは、柔軟性、スケーラビリティ、応答性、およびレジリエンシーに優れた疎結合のアプリケーションの構築手段として人気の高いモデルです。このモデルは、高並行性・低レイテンシーのサービス、つまり非同期の非ブロッキング I/O を使用するイベント駆動型のサービスに最適です。Vert.x を使用すると、Java 開発者は既存の専門知識を活用しながら、リアクティブ・プログラミングやマイクロサービスに移行することができます。Vert.x 自体が多言語対応であり、開発者は JVM 上で動作する他の言語を使用することができます。ただし、OpenShift Application Runtimes の初回リリースでは、Java のみがサポートされています。
- Red Hat JBoss Web Server ランタイム**は、Spring Boot を使用するマイクロサービスや Spring フレームワークを使用するアプリケーション用に提供されています。JBoss Web Server には、Apache Tomcat、Apache HTTPD、Apache CXF、および Hibernate が含まれ、これらには Red Hat によるフルサポートが提供されます。このランタイムは、Spring、Spring Boot、Spring Cloud Kubernetes、および Netflix OSS (Hystrix and Ribbon) を組み合わせて使用します。これらのコンポーネントは、Red Hat が動作の連携をテストし、検証しています。インテグレーション (統合) を行うことで、OpenShift または Kubernetes のネイティブ・サービスを、構成やサービス検出といったサービス用の使い慣れた Spring アプリケーション・プログラミング・インタフェース (API) で使用することができます。ネイティブサービスを使用するメリットは、Red Hat OpenShift で管理できる点です。
- Node.js**: JavaScript ランタイムです。これはイベント駆動型のサーバーサイド・フレームワークであり、モバイルやブラウザー・バックエンドなど、「JavaScript Everywhere」のアプローチにおいてよく使用されます。このランタイムにより、JavaScript 開発者はすべてのランタイムで同様のサービスにアクセスできます。すべてのランタイムが、Red Hat OpenShift を介して、同じ管理機能と運用機能を共有します。OpenShift Application Runtimes の初回リリースでは、Node.js はテクノロジー・プレビューと見なされています。

図 2 は、Red Hat OpenShift Application Runtimes のコンポーネントの概要を示しています。

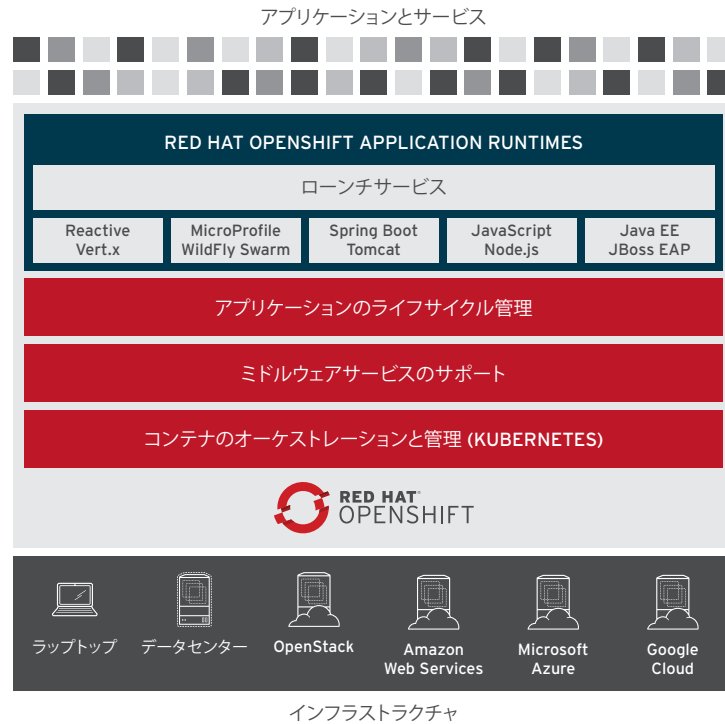


図 2. Red Hat OpenShift Application Runtimes のコンポーネント

RED HAT OPENSIFT APPLICATION RUNTIMES の使用

クラウド移行による既存アプリケーションのモダナイゼーション

アプリケーションをクラウドネイティブ対応させるために、最新のフレームワークやアーキテクチャ手法を使用して書き直すことは、必ずしも実用性やコスト効率に優れているとは限りません。しかし、既存アプリケーションにはいずれにせよメンテナンスが必要であり、実装が必要な改良のバックログも存在します。リリースサイクルが長い場合、バックログの消化が困難になりがちです。OpenShift Application Runtimes を使用して既存アプリケーションをクラウドにリホストすれば、CI/CD の実践に加えて、ローリングリリース（ブルーグリーン・デプロイメント、カナリア・デプロイメントなど）の実現も格段に容易になります。これにより、より低リスク・高頻度でのリリースが可能になります。クラウドでホストされているアプリケーションが「高速で動作するモノリス」となり、機能改良のバックログへの対応がより容易になります。

アプリケーションをクラウドにリホストすることは、マイクロサービスの実装に向けた効果的な第一歩にもなり得ます。クラウドは柔軟性に優れているため、新しいマイクロサービスを既存アプリケーションとともにコンテナにデプロイすることが、より容易になります。その後、モノリシック・アプリケーションから機能を移行するマイクロサービスを実装できます。

アプリケーション・サーバーからクラウドランタイムへの移行

従来のアプリケーション・サーバーの役割は、クラウドネイティブ・プラットフォームに移行する時点で変わります。従来のアプリケーション・サーバーは、ランタイムとデプロイメント環境、およびシステムクラスタで実行される中央ドメインの管理サービスを提供していましたが、移行後には、クラウドネイティブ・アプリケーションのランタイム環境として機能することができます。ただし、組み込まれている運用ツール（管理コンソールなど）の多くは不要となり、OpenShift と Kubernetes で提供される自動化管理機能の動作を妨げるおそれもあります。

OpenShift Application Runtimes には、Java EE アプリケーションの移行に対応できるよう、JBoss EAP をベースとしたクラウドネイティブ・ランタイムが含まれています。このランタイムは、JBoss EAP の先進的なモジュール式アーキテクチャを使用することで、動的スケーリングや CI/CD といった OpenShift の機能を簡単に活用できます。Red Hat Application Migration Toolkit では、既存の従来型アプリケーション・サーバー上で引き続き実行されるアプリケーションのために、プロジェクトを分析し、JBoss EAP に移行するために変更が必要であると見なされる領域を強調表示することができます。

現在多くの組織が、Java EE の小規模なサブセットしか使用していないアプリケーション・サーバーにアプリケーションをデプロイしています。OpenShift Application Runtimes は、これらのアプリケーションのための、より軽量な選択肢を提供します。WildFly Swarm ランタイムにより、アプリケーションを、実際にアプリケーションで使用されるアプリケーション・サーバーのコンポーネントのみとともに、単一の実行可能な JAR ファイルにアセンブルすることができます。Red Hat JBoss Web Server ランタイムは、Web アプリケーションや、ごく小規模なコンテナしか必要としないアプリケーションのために、Apache Tomcat と Apache HTTPD サーバーをベースとした、フルサポート付きのクラウドネイティブ・ソリューションを提供します。

マイクロサービスの構築

OpenShift Application Runtimes には、マイクロサービスを構築する際に使用できるランタイムが複数含まれています。

- Java EE の専門知識を持つ開発者にとって、使い慣れた Java EE 標準のサブセットと、進化するマイクロサービスの世界に適した俊敏な Eclipse Java Microprofile 標準とを融合させた WildFly Swarm ランタイムは、魅力的な選択肢です。
- リアクティブ・マイクロサービスの構築には、Eclipse Vert.x ランタイムが提供されます。さらに Vert.x は多言語開発に対応できるように設計されており、JVM 上で動作する他の言語の使用が可能となります。
- Spring Boot マイクロサービスは、JBoss Web Server ランタイムを使用して構築することができます。このランタイムには、サポートされる Apache Tomcat コンテナや、Spring Cloud Kubernetes を含むその他のコンポーネントが含まれます。Spring Cloud Kubernetes は、Spring API と Red Hat OpenShift の運用管理機能を連携させます。

マイクロサービス・アーキテクチャは、ソフトウェア・エンジニアリングの多くの課題を解決する能力に極めて優れたアプローチですが、一方で複雑さが増すという課題を抱えています。マイクロサービスを使用する場合、サービス検出やクライアントサイドのロードバランシングなど、追加のインフラストラクチャ・サービスが必要です。独立的にデプロイされるサービスの数が増えると、サービス間の分散トレースのような監視やメトリックに関わる機能が、運用上の主要な要件となります。

OpenShift Application Runtimes は、これらのニーズに対応するように設計されています。このランタイムには、Netflix Hystrix や Ribbon などの多数のインフラストラクチャ・コンポーネントが含まれており、カスケード障害を回避しながらマイクロサービス間でプロセス間通信を堅牢な方法で処理します。OpenShift Application Runtimes のローンチサービスは、これらのコンポーネントを使用してサーキットブレーカーのようなパターンを実装する方法を示す実行可能なコードを提供して、低速または応答しないサービスの問題が複数の障害へと発展することを防ぎます。

不要な複雑さの回避

OpenShift Application Runtimes には、開発者に多くのメリットを提供するソフトウェア・インフラストラクチャ・コンポーネントが含まれています。これらを利用することで、インフラストラクチャにまつわる問題を、アプリケーションではなくランタイムによって、より典型的に処理できるようになります。開発者は、独自のコンポーネントを構築したり、使えるコンポーネントを検索したりする代わりに、Red Hat が提供するコンポーネントとサンプルコードを使用することができます。この場合、サードパーティ製のコンポーネントを組み込む場合に発生する保守上の負担とリスク（バグやセキュリティ脆弱性の情報を常に把握し、対応し続ける必要）も避けることが可能です。

目的に適したツール

かつては、すべての開発プロジェクトに単一のフレームワーク、言語、アーキテクチャ・スタイルを使用することも現実的だったかもしれませんが、状況は変わりました。Red Hat が最近実施した調査では、お客様の半数近くが、目的に適したツールを選択することが優先事項であると回答していました。Red Hat OpenShift Application Runtimes には、複数のフレームワークと言語をサポートする複数のランタイムが含まれています。

これらのランタイム同士には多くの共通点があるため、複数のフレームワークと言語を使用することが現実的となっています。これらのランタイムで構築されたアプリケーションは、Red Hat OpenShift を使用して同じ方法で設定、デプロイ、および管理することができ、開発者と運用チーム両者の負担が軽減されます。すべてのランタイムは、Red Hat Single Sign-On で保護することができます。さらに、ローンチツールは利用価値の高い学習ツールであり、各ランタイムでの実装例が示されます。

オンプレミスとマルチクラウド・サポート

大半の IT 組織は、1 つだけのパブリッククラウド・インフラストラクチャ・プロバイダーに拘束されなことを望んでいます。また、多くの組織が、数年後においても、アプリケーションの半数以上は依然としてオンプレミスで実行されるものと考えています。Red Hat OpenShift のデプロイメントは極めて柔軟であり、Red Hat OpenShift Online などのパブリッククラウド（主要なクラウド・インフラストラクチャ・プロバイダーで利用可能）、Red Hat OpenShift Dedicated などのプライベートクラウド、Red Hat OpenShift Container Platform による自社インフラを利用できます。

今すぐ試す

Red Hat OpenShift Application Runtimes に含まれるローンチサービスで、開発者視点での使い心地を今すぐ試すことができます。developers.redhat.com/launch では、実行可能なサンプルプロジェクトを作成して OpenShift Online にデプロイすることができます。プロジェクトを zip ファイルとしてダウンロードし、ローカルで構築、デプロイすることもできます。

ローンチサービスでは、まずミッションを選択します。ミッションとは、先進的アプリケーション開発の基本的な側面を示す、事前構成済みの、実際に機能するアプリケーションです。選択できるミッションは以下のとおりです。

- **CRUD**: HTTP REST エンドポイント上のリレーショナル・データベースに対する作成、読み取り、更新、および削除操作を行います。
- **サーキットブレーカー**: サーキットブレーカー・パターンは、正常に回答していないサービスへのアクセスを制限することによって、カスケード障害を防ぎます。
- **外部設定**: 環境設定をアプリケーション・コードから分離する ConfigMap の使用例です。
- **ヘルスチェック**: ヘルスチェック・パターンは、アプリケーションが使用可能であり、要求を処理できる状態になっているかどうかをチェックします。これは、アプリケーションやサービスが複数のインスタンスを介してスケールアウトできるようにするために必要な手順の1つです。
- **REST API レベル 0**: HTTP を介したリモート・プロシージャ・コールのための REST エンドポイントの基本的な例です。

ミッションを選択したら、Eclipse Vert.x、Node.js、Spring Boot、Wildfly Swarm のいずれかの使用可能なランタイムをプロジェクトに使用するように選択できます。同じミッションをさまざまなランタイムに使用できるため、実際に試して比較することができます。図 3 は、ローンチサービスのフローを示しています。

OpenShift Online にデプロイする場合は、コードが GitHub アカウントのプロジェクトにフォークされます。Red Hat OpenShift プロジェクトは、GitHub リポジトリのマスターブランチへ push がある度に構築およびデプロイするように構成されています。このプロセスのメリットは、プロジェクトのコードを Red Hat OpenShift で実行するために必要なすべての設定、構築、およびデプロイメントの手順を処理できる点です。これにより、Red Hat OpenShift が開発とデリバリーのプロセスをどのように加速できるかを体験することができます。

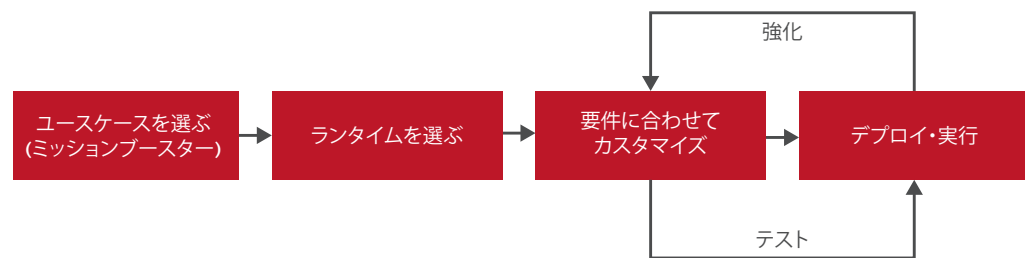


図 3. Red Hat OpenShift Application Runtimes のローンチサービス

次のステップ

Red Hat では、以下のように、先進的なクラウドネイティブ・アプリケーションを構築するために役立つさまざまな資料を提供しています。

- Red Hat OpenShift Application Runtimes と Red Hat OpenShift について詳しくは、[OpenShift ブログ](#)をご覧ください。また、OpenShift Commons の Red Hat OpenShift Application Runtimes に関するブリーフィングビデオをご覧ください。
- [developers.redhat.com](#) から [Red Hat Development Suite](#) をダウンロードして、ラップトップまたはデスクトップに完全な開発環境をインストールすることができます。Red Hat Development Suite には以下が含まれています。
 - [Red Hat JBoss Developer Studio](#) : Eclipse をベースとした包括的な統合開発環境 (IDE) であり、OpenShift やコンテナ開発で利用できる幅広いツールを備えています。
 - [Red Hat Container Development Kit](#) : Windows または macOS を実行している場合でも、ローカルマシン上でシングルノードの OpenShift クラスタを実行できます。
- OpenShift Online 上で動作する開発者向け SaaS (Software-as-a-Service) プラットフォームである [Red Hat OpenShift.io](#) を使用すると、ソフトウェアをインストールすることなく、開発チームの誰もが使える完全なクラウドネイティブ開発環境を用意する方法を学ぶことができます。注: OpenShift.io は現在、制限付きで開発者プレビュー中です。
- 既存アプリケーションの移行を開始するための分析ツールは、[Red Hat Application Migration Toolkit](#) から入手できます。このツールキットは、プロジェクト・ソース・ディレクトリやアプリケーション・アーカイブなどのアプリケーション成果物を検査して、詳細な HTML レポートを生成します。レポートでは、移行を実現するために変更が必要な領域が強調表示されます。この結果は、レガシー Java アプリケーション・サーバーから [JBoss EAP](#) にアプリケーションを移行する場合に特に役立ちます。JBoss EAP は、OpenShift Application Runtime に含まれるランタイムの1つです。
- チームのための参加型ワークショップである [Red Hat Open Innovations Labs](#) を使用すると、先進的アプリケーションの開発を迅速にスタートできます。
- メッセージング、インメモリ・データグリッド、Red Hat OpenShift 上で動作するビジネスプロセスやルール管理など、[Red Hat JBoss Middleware](#) で提供される機能をご確認ください。
- Red Hat が作成した[無料の e ブック](#)をご覧ください。この e ブックでは、マイクロサービスや先進的な Java EE デザインパターンなど、関連するトピックを扱っています。

RED HAT について

オープンソースソリューションのプロバイダーとして世界をリードする Red Hat は、コミュニティとの協業により高い信頼性と性能を備えるクラウド、Linux、ミドルウェア、ストレージおよび仮想化テクノロジーを提供、さらにサポート、トレーニング、コンサルティングサービスも提供しています。Red Hat は、お客様、パートナーおよびオープンソースコミュニティのグローバルネットワークの中核として、成長のためにリソースを解放し、ITの将来に向けた革新的なテクノロジーの創出を支援しています。

アジア太平洋 +65 6490 4200	インドネシア 001 803 440224	ニュージーランド 0800 450 503	ベトナム 800 862 6691
オーストラリア 1 800 733 428	日本 03 5798 8510	フィリピン 800 1441 0229	中国 800 810 2100
ブルネイ / カンボジア 800 862 6691	韓国 080 708 0880	シンガポール 800 448 1430	香港 852 3002 1362
インド +91 22 3987 8888	マレーシア 1 800 812 678	タイ 001 800 441 6039	台湾 0800 666 052



facebook.com/redhatjapan
@redhatjapan
linkedin.com/company/red-hat