

パブリッククラウド上のデータ活用システムの構築自動化に向けた
デプロイメント・パイプライン構成の検討

A Study on a Configuration of Deployment Pipeline for Automated Deployment of a Data Management System in a Public Cloud

南 豪介[†] 長谷 亮[†] 松浦 陽平[†]
Gosuke Minami Ryo Hase Yohei Matsuura

1. はじめに

近年、行政・民間組織におけるデータ活用の拡大が期待されている[1]。一例として、複数場所に保存されるデータの統合・分析のために、データを収集する機能を有したデータ連携基盤と分析アプリで構成されるシステムが様々な分野で期待される。本稿では、上記のようなシステムをデータ活用システムと記載する。データ活用システムの概要を図1に示す。データ活用システムの環境構築に、パブリッククラウド（以降、クラウド）の適用が広まりつつある。

しかし、データの処理・分析を取り扱う人材の不足が、データ活用システムの多分野適用の障害になっている[1]。データ活用システムの構築では、上記人材の役割としてシステム構築エンジニアが挙げられる。システム構築エンジニアはデータ連携基盤、分析アプリの構築・設定作業において通常、UIを用いてITインフラ、アプリの構築・設定を手作業で行う。上記UIでは、Webブラウザなどを利用して容易に構築・設定作業を行えるが、構築・設定対象が増加すると作業量も増加する。また、システムを構成するクラウドの機能追加、システムの要件変更・追加の度に、手作業で構築・設定作業を行う必要がある。そのため、システム構築エンジニアの作業労力を低減する手法を、システムの要件変更・追加を見越して検討する必要がある。

本稿では、システム構築エンジニアの作業労力の低減を目的として、クラウド上でのデータ活用システムの構築・設定を自動化するための、デプロイメント・パイプラインの構成を検討した結果を記載する。

2. 関連技術と課題

クラウド上でのデータ活用システムの構築を支援する手法としてPlatform as a Service（以降、PaaS）の利用が挙げられる。PaaSとは、クラウドベンダがOSやミドルウェアを含めて、利用者にシステムの実行環境を提供するサービスである。PaaSにより、ITインフラ、アプリを容易に構築できるが、設定項目は依然として多い。そのため、多くの設定項目を一度に設定し、構築する手法が求められる。

PaaSの構築・設定を自動化するための技術に、Infrastructure as Code（以降、IaC）がある。IaCとは、PaaSの構築・設定項目をコードとして記述したテンプレートを作成し、実行することで、PaaSの構築・設定を自動化する技術である。先行研究において、IaCを活用することで、データ連携に必要なITインフラとデータ連携機能を、クラウド上に自動構築できることが示されている[2]。しかし、ITインフラの変更・追加への対応を円滑にするには、各PaaSに対応するIaCを追加・管理でき、かつ、構築・設

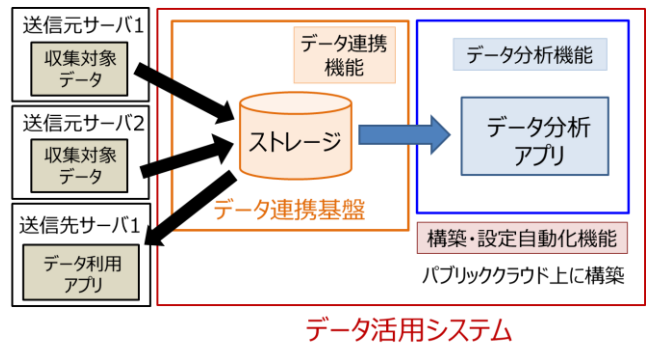


図1 データ活用システムの概要

定内容に応じてIaCを選択して自動実行できる仕組みを実現する必要があります。

また、アプリのデプロイ先に変更・追加が生じた場合にも、デプロイ先の環境に合わせて、ミドルウェアやライブラリ群、ITインフラの設定などを整備して、アプリをデプロイする必要があるため、デプロイ作業労力が大きくなる。

アプリの実行環境を構築する手法としてコンテナ技術が挙げられる。代表的なコンテナ技術であるDockerでは、アプリの実行に必要な要素をコンテナイメージとしてパッケージ化することで、ローカルPCやクラウド上など、異なる環境でも動作するコンテナアプリを作成できる。先行研究において、Dockerコンテナのビルドからリリースまでを自動化したパイプラインが検討されている[3]。これにより、継続的なアプリのリリースを円滑に行えるが、アプリの実行基盤などのITインフラに変更・追加への対応を容易にするパイプラインの検討は未だ行われていない。

以降では、データ活用システムのITインフラを構成する各PaaSに対応するIaCを追加・管理可能であり、上記IaCをもとに構築したITインフラを選択してコンテナアプリをデプロイ可能な、デプロイメント・パイプラインの構成を検討した結果について説明する。

3. デプロイメント・パイプラインの構成

3.1. デプロイメント・パイプラインの概要

今回は、以下の用途のデプロイメント・パイプラインを構築した。

- ・インフラ構築パイプライン

データ連携に必要なITインフラや、分析アプリの実行に必要なITインフラの構築・設定を自動化する。

- ・アプリデプロイメントパイプライン

コンテナアプリのビルドとデプロイを自動化する。

[†]三菱電機株式会社 情報技術総合研究所 Information Technology R&D Center, Mitsubishi Electric Corporation.

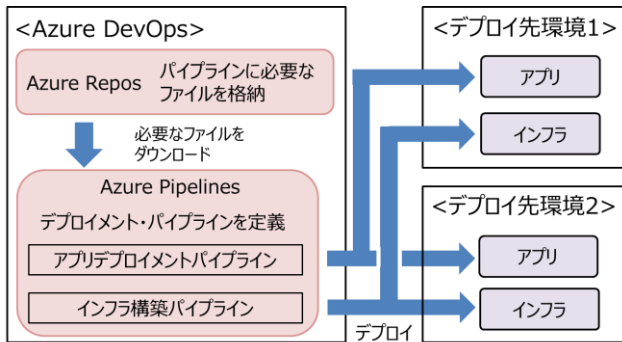


図2 パイプラインの全体構成

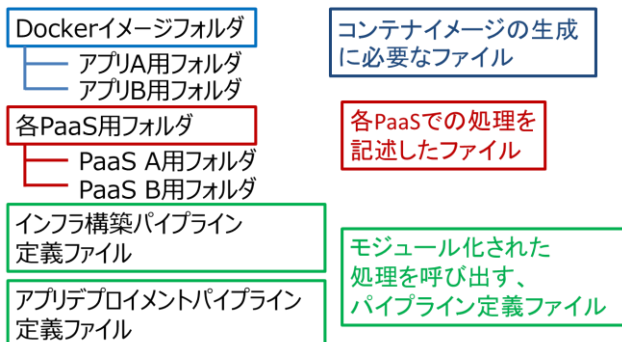


図3 リポジトリ構成の例

今回は、クラウドとして Microsoft Azure を検討対象とした。デプロイメント・パイプラインの実行に必要なファイルは、Git リポジトリの機能を提供する Azure Repos に格納する。ビルドやデプロイなどの処理は、デプロイメント・パイプラインの実行基盤である Azure Pipelines で実行される。Azure Pipelines での処理内容を記載する定義ファイルは、YAML ファイルとしてリポジトリに格納する。IT インフラの構築・設定自動化には、Microsoft Azure で提供される IaC である Azure Resource Manager テンプレート（以降、ARM テンプレート）を利用した。ARM テンプレートは、JSON ファイルとしてリポジトリに格納する。

デプロイメント・パイプラインの全体構成を図2に示す。図2では割愛したが、コンテナイメージの管理を行うコンテナレジストリとして、Azure Container Registry も利用する。

3.2. パイプライン処理のモジュール化

デプロイメント・パイプラインの定義ファイルにおける各処理の記述仕様は、IT インフラを構成する PaaS によって異なる。そこで本研究では、PaaS ごとに特化した処理をモジュールとして分割して、個別に管理するリポジトリ構成を検討した。検討したリポジトリ構成の例を図3に示す。また、アプリデプロイパイプラインにおける、処理のモジュール化の例を図4に示す。

システム構築エンジニアは、各 PaaS 向けの処理に対応するモジュールを作成し、各 PaaS のフォルダに格納する。PaaS に依存しない共通の処理フローを記述したパイプライン定義ファイルは、パイプラインの用途ごとに作成する。

3.3. デプロイメント・パイプラインの試作

検討した構成の実現性検証のため、デプロイメント・パイプラインの試作を行った。具体的な PaaS として、コンテナ実行基盤の PaaS である Web Apps for Containers と Azure

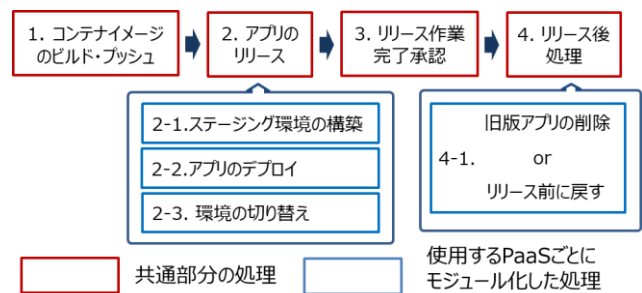


図4 アプリリリース処理のモジュール化

Kubernetes Service のモジュールを作成した。試作の結果、全般的なデプロイメント・パイプラインの処理を共通化しつつ、各 PaaS の構築自動化と、PaaS に対するアプリデプロイの自動化を実現できることを確認した。

4. 考察

IT インフラを構成する PaaS の処理をモジュールとして作成し、分類して保管・利用するデプロイメント・パイプラインの検討を行った。その結果、利用する PaaS の変更・追加時は、対象となる PaaS のモジュールを作成し、パイプラインに組み込むことで、変更・追加対象外の PaaS に影響を与えることなく、自動構築・設定を行うことが可能である。そのため、システム構築エンジニアは、PaaS の変更・追加に対して、円滑に対応することが可能となった。

コンテナ化したアプリを、複数の実行基盤から選択してデプロイできるデプロイメント・パイプラインの検討を行った。その結果、システム構築エンジニアは、デプロイ先の環境に合わせて利用するモジュールを選択すれば、デプロイメント・パイプラインのデプロイ先を容易に変更可能となった。そのため、アプリのデプロイ先の環境に変更・追加が発生した場合の、システム構築エンジニアの作業労力を低減可能と考えられる。

5. おわりに

本研究では、クラウドを活用しデータの連携・分析を行うデータ活用システムにおいて、システムの要件変更・追加への対応を容易にするデプロイメント・パイプラインの構成を検討した。結果として、IT インフラの構築・設定や、コンテナアプリのデプロイにおける PaaS に特化した処理をモジュール化したデプロイメント・パイプラインの検討を行い、システムを構成する IT インフラや、アプリのデプロイ先に変更・追加が発生した際の作業労力を低減できる可能性を示した。今後の課題として、デプロイメント・パイプラインによるシステム構築エンジニアの労力低減について、定量的な評価を行うことが挙げられる。

参考文献

- [1] 総務省.情報通信白書令和2年版,
<https://www.soumu.go.jp/johotsusintokei/whitepaper/r02.html> (閲覧: 2021年6月1日) .
- [2] 南 豪介, 長谷 亮, 松浦 陽平, “パブリッククラウドを活用したデータ連携基盤の Infrastructure as Code による自動構築方式の検討” 情報処理学会 第 83 回全国大会講演論文集, pp. 1-141~1-142, 2021.
- [3] N. Raičić and M. Savić, "Architecting Continuous Integration and Continuous Deployment for Microservice Architecture," 2021 20th International Symposium INFOTEH-JAHORINA, pp. 1-5, 2021.