

クラウドで実現する迅速簡便なシステム検証環境の構築

Rapid and simple setup of system evaluation environment by the Cloud

落合 寛彰 吉成 雅通 人見 瑛一 浜田 善夫

Abstract

At Yamaha Motor Co., Ltd. (hereinafter referred to as “the Company”), the IT Division is advancing strategic initiatives in IT, digital technology, and data utilization under the long-term vision “Art for Human Possibilities” toward 2030. Under the slogan “Yamaha Motor to the Next Stage,” the Company promotes three key digital transformation (DX) programs: Y-DX1 (Reforming the Management Foundation), Y-DX2 (Strengthening the Present), and Y-DX3 (Creating the Future)^[1].

Through new Yamaha experiences, services, and products that blend the real and digital worlds, the Company aims to strengthen customer engagement, enhance brand value, and foster lifelong Yamaha fans. To realize this goal, the IT Division’s role is to provide rapid and effective solutions that meet business needs promptly, while building a cloud-native environment as the foundation of DX activities.

This paper introduces one such initiative — the development of a rapid and easy-to-set-up system verification environment. This environment has already been adopted in several projects and is contributing as a core foundation for Yamaha’s DX initiatives.

1 はじめに

ヤマハ発動機株式会社(以下当社)IT 本部では、2030年までの長期ビジョン “Art for Human Possibilities” に向けて、IT・デジタル技術・データの戦略的活用を図るため、“Yamaha Motor to the Next Stage” を掲げ、Y-DX1 (経営基盤改革)、Y-DX2 (今を強くする)、Y-DX3 (未来を創る) から成る3つのデジタルトランスフォーメーション (DX) 活動を推進している^[1]。リアルとデジタルで創出するヤマハの新たな体験・サービス・製品を通じてお客さまとつながり、ブランド価値の向上を図り、生涯にわたるヤマハファンの創造を目指している。その実現に向け、ビジネス部門からの要求に迅速に応えるソリューションを提供することが IT 部門の役割であり、DX 活動の基盤としてクラウドネイティブな環境の実現に取り組んでいる。

本稿では、その取り組みの一つとして、迅速かつ簡便に構築可能なシステム検証環境を紹介する。この環境は既に複数のプロジェクトで活用されており、DX 活動の基盤として貢献している。

2 背景

企業は市場のニーズに迅速に対応し、競争力を維持することが求められており、その実現手段としてクラウド技術は重要な要素である。クラウド技術は急速に進化しているが、当社ではこれまでクラウド技術を活用したシステム検証環境を提供しておらず、プロジェクトごとにインフラやアプリケーションの構築・設

定を個別に行っていたため、多くの工数とリードタイムが発生していた。そこで、クラウド技術を活用し、迅速かつ簡便で安全に技術検証を行えるシステム検証環境を構築した。

3 解決すべき課題

当社が新しいシステムの検証に際して抱えていた課題として、従来の構成では次の3点が挙げられる(図1)。

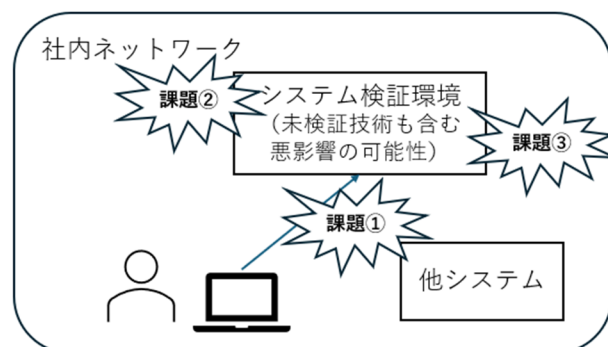


図1 システム検証環境構築の課題

課題①：安全性の確保

新しいシステムを稼働させる際、他の社内システムやネットワークに悪影響を及ぼさない安全性の確保が必要である。特に、セキュアな通信経路の確保や、検証環境が他システムと隔離された構成が求められる。

課題②：迅速かつ簡便な環境構築

オンプレミス(自社内に設置・運用する IT インフラ)で検証環境を準備する場合、最短でも2週間を要し、その後の環境設定やアプリケーション構築には1カ月以上を要する。これでは、冒頭で述べた市場のニーズに迅速に対応したい業務とシステム開発との間に乖離が生じる。したがって、システム稼働に必要なインフラやアプリケーションを迅速かつ簡便に構築することが求められる。

課題③：環境の再現性および変更管理の容易さ

本番環境構築時には、検証環境と同等の構成を用意する必要がある。一部設定に差異が生じる場合もあるが、安定した再現性の確保は、規模が大きくなるほど困難となる。また、構築済みの環境に対しても随時変更が発生するため、変更管理の容易さも重要な要件である。

4 課題解決の利用技術

3章で述べた課題を解決するために、以下に AWS(Amazon Web Services[※])を中心とした6つのクラウド技術を説明する。

[※]Amazon Web Services, “Powered by Amazon Web Services” ロゴ、[およびかかる資料で使用されるその他の AWS 商標] は、米国その他の諸国における、Amazon.com, Inc. またはその関連会社の商標です。

利用技術①：クラウド(AWS)

インターネット経由でクラウド環境のサーバ、データベース、ネットワークなどの機能を利用できるサービス群である。社内ネットワークと隔離された環境であるため、社内に悪影響を及ぼすことなく、新しいシステムを安全に検証できる。

利用技術②：AWS Session Manager

操作端末からサーバへ高いセキュリティで接続可能な、AWS が提供する通信技術サービスである。従来の接続方式では、クラウドサーバにインターネットからの接続口を開ける必要があり、攻撃者による脆弱性の悪用が懸念されていた。

しかし、Session Manager では、接続口を設けることなく操作端末から接続できる。通信には暗号化された https プロトコルを用いるため、通信の盗聴や改ざんに対して高い安全性を有する。さらに、接続認証には AWS が標準提供する IAM (Identity and Access Management) を使用し、高い信頼性と堅牢性を確保している。

利用技術③：IaC(Infrastructure as Code)

クラウド上のサーバ、データベース、ネットワークなどの機能をプログラムコードとして定義・実行することで、インフラ環境を自動構築できる技術である。代表的な IaC ツールには

Terraform や AWS CDK がある。IaC コードを実行するだけで環境構築が可能となり、構築時間は約1～15分程度に短縮される。

また、類似のインフラ環境が必要な場合には、過去の IaC コードを再利用することで、最小限の修正で同一環境を構築できる。

利用技術④：コンテナ

システムと、その実行に必要なプログラムを独立したパッケージとしてまとめる技術である(図2)。

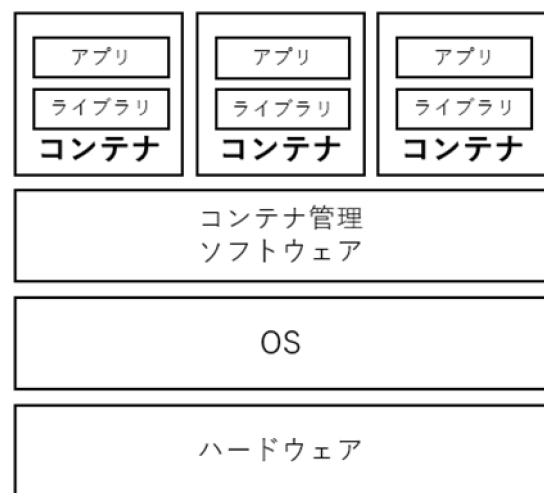


図2 コンテナの概念

システム間の独立性により、稼働の安定性とセキュリティの確保を両立し、起動の速さやメモリなどのリソース消費の抑制にも寄与する。Docker が最も広く利用されているコンテナ技術であり、“コンテナ”という用語は Docker コンテナを指す場合が多い。

コンテナは設計図(テキストファイル)から作成され、起動は簡単なコマンドで行う。実行時間は数秒から数分と比較的短時間である。

また、コンテナを複製しても環境は同一であり、同様に動作することが保証されるため、複数のメンバがそれぞれの環境でシステム技術进行评估する際にも、環境の差異に起因する問題が生じにくい。

利用技術⑤：DevContainers

Microsoft が提供するエディタ“VSCode”内で呼び出される機能であり、システム動作環境全体を Docker コンテナ内にカプセル化する技術である。複数のメンバでシステム検証を行う場合、設定ファイルを共有することで、全員が同一の環境に同一の方法で接続して使用できるため、環境差異による問題を抑止しつつ、開発スピードの向上が可能となる。

利用技術⑥：GitHub

ソフトウェア開発プロジェクトにおいて、主にバージョン管理とソースコード管理を提供する SaaS 型サービスである。コンテナの設計図(テキストファイル)、DevContainers の設定、IaC のプログラムコードの変更管理やバージョン履歴の管理に使用される。バージョン履歴を活用することで、過去の任意の時点のバージョンにいつでも戻すことが可能である。

これらのクラウド技術を組み合わせたシステム検証環境(図3)により、図1に示した課題①～③を解決することができた。

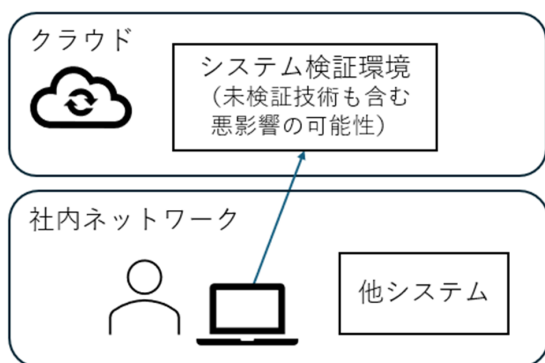


図3 システム検証環境の解決策

5 活用事例

この活用事例として、システム開発支援 AI エージェントシステムの検証を目的とした環境構築を紹介する(図4)。

検証対象のシステム(オープンソースソフトウェア)は、自然言語による指示でプログラムコードの作成、デバッグ、起動および動作確認を行うものであり、エディタ(VSCode)内でツールを呼び出して使用する。

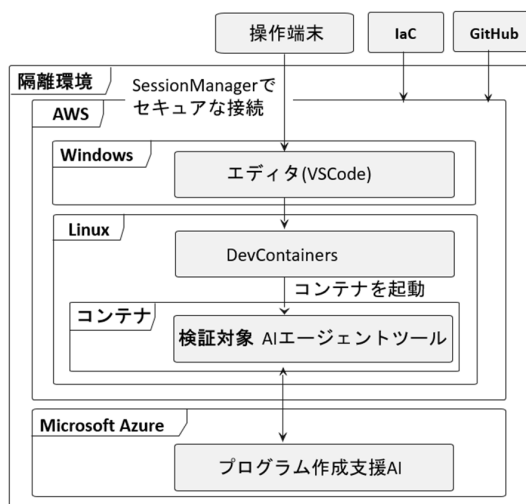


図4 活用事例の環境構成

対応①：安全性の確保

・AWS、AWS Session Manager

検証対象システムに有害なバグやセキュリティリスクが含まれる可能性を考慮し、操作端末は画面情報のみをやり取りするリモート操作端末として使用し、実際の処理はすべてクラウドの隔離環境で検証を行うことができた。

対応②：迅速かつ簡便な環境構築

・コンテナ、DevContainers、IaC

検証対象システムを動作させるコンテナを作成し、VSCodeから接続した。手作業の場合、手順書に基づき毎回作成から接続後の設定までを行う必要があるが、自動化により手順のミスなく容易に環境構築が可能となった。従来では1カ月以上を要していた環境構築を、今回は約2日間で完了し、93%以上の納期短縮を達成した。これにより、多くの工数とリードタイムのストレスから解放され、迅速に検証を開始できた。

対応③：環境の再現性および変更管理の容易さ

・GitHub、IaC

AWS 上の Windows や Linux、ネットワーク構成の設計図は、250行のテキストで記述されている。今回の活用事例では、試行錯誤を繰り返しながら環境変更を行ったが、混乱することなく一貫性のある変更管理が可能であった。

また、検証環境を一度削除し、別のメンバが再度検証を行う際には、手作業であれば半日要するところ、数分で環境構築を完了できた。

6 おわりに

本稿ではクラウド技術を活用し、迅速かつ簡便に構築可能なシステム検証環境を紹介した。これにより、従来のオンプレミス環境に比べて、環境構築のリードタイムを93%以上短縮し、安全で効率的なシステム検証が可能となった。

冒頭に述べたとおり、クラウド技術は日々進化しており、IT エンジニアはこれに継続して追従する必要がある。当社 IT 本部およびヤマハモーターソリューション株式会社では、クラウド活用を推進する組織体として CCoE (Cloud Center of Excellence) を両社合同で立ち上げ、AWS を中心にクラウド技術の相談・支援および人材教育をミッションとして活動している。この CCoE 活動を通じて、本稿で紹介した技術が社内において標準技術として浸透することが、クラウドネイティブな環境の実現につながり、DX 活動に寄与すると考える。

また、DX 推進に向けた基盤機能の拡張として、現在は生成 AI を活用したプログラムコードの自動生成、テストの自動化、グローバルでのコード共有などに挑戦している。これらの活動

は、システム開発におけるリードタイムを劇的に短縮する可能性を有しており、研究開発を継続することで、お客さまへのサービス即時提供を目指している。

■参考文献

[1] ヤマハ発動機株式会社:「中期経営計画 2025-2027」,
<https://global.yamaha-motor.com/jp/profile/mtp/pdf/2025/2025medium-plan.pdf>

■著者



落合 寛彰
Hiroaki Ochiai
IT 本部
プロセス・IT 部



吉成 雅通
Masamichi Yoshinari
IT 本部
プロセス・IT 部



人見 瑛一
Eiichi Hitomi
IT 本部
プロセス・IT 部



浜田 善夫
Yoshio Hamada
IT 本部
プロセス・IT 部