

第四次産業革命と「Kubernetes」

クラウドネイティブ・コンピューティングに至る背景

第四次産業革命の入り口に到達していると言われている2020年。この変革点において、未来のソフトウェア開発とシステム運用の要件とは、どのようなものでしょうか。その要件に答えるかのように活動しているのが非営利団体「Cloud Native Computing Foundation」です。その革新的なアプローチの柱となるのが、「コンテナ」「イミュータブル・インフラストラクチャー」「Kubernetes(クーバネティス)」であり、これらの最先端のオープンソースソフトウェア(以下、OSS)を企業向けにチューニングしたのが「Red Hat OpenShift」です。

本稿では、時代背景からアプローチ方法、そして、OSSの特徴について解説します。

▶▶ 1. 第四次産業革命の推進力となる クラウドネイティブ・コンピューティング

第四次産業革命による変化を一言で表現するのは容易ではありません。日本政府が掲げる「Society 5.0」やドイツ連邦政府が掲げる「Industry 4.0」でも、第四次産業革命を標榜しています[1][2]。そして、経済産業省が2018年に発表したDXレポートは、日本企業のデジタルトランスフォーメーション(DX)が進まなければ、2025年以降、最大12兆円/年の経済損失が生じる可能性を示し、「2025年の崖」として警鐘を鳴らしています[3][4]。

第四次産業革命を特徴づけるのは、これまでとは比較にならないほど、多様な場所で利用できる高速で大容量のインターネット・アクセス、高性能で小型化されたセンサーを活用するIoT、非定型でさまざまなデータを取り扱うことができる人工知能技術、そして、低価格かつ時間単位で利用可能なクラウド・コンピューティングです。

この第四次産業革命の変化の片鱗を、われわれは日々感じながら生活しています。例えば、現代人はスマートフォンを手放せません。それはスマートフォンをコミュニケーションのツールとして利用するだけでなく、必要な情報へのアクセス手段や、電子決済のデバイスとしても利用しているからです。加えて、悪天候や地震などの

自然災害時には、気象情報の収集、避難場所と経路情報など、命を守る行動のための情報源にもなります。

スマートフォンの普及の次に変革の原動力として予測されているのがCASEです。この単語は、自動車のネットワーク接続(Connected)、自動運転の実用化(Autonomous)、自動車の共有やサービス化(Share & Service)、地球温暖化の対策としても期待される電気自動車化(Electric)の頭文字からきています。10年先には、世界の自動車台数は3億台増えて13億台に達し、そのうちの10億台以上がネットワークに接続されると予測されています。これにより、公共交通システムから社会インフラまでも連結した、超巨大なネットワークとなることは間違いありません[5]。

この変化は、自動車をモノとして所有することから、サービスとして利用する「MaaS(Mobile as a Service)」への変革をさらに加速させるでしょう。利用者は、スマートフォンから目的地を指定することで、現在の経路案内だけではなく、自動運転タクシーをはじめとするさまざまな交通手段と連携して目的地へ連れて行ってもらえる究極のモビリティ・サービスを受けられるようになることが予測されます[6]。

MaaSのようなシステムの開発・試験・運用では、1つの単一なシステムとして構築するよりも、今後の新し

い技術への適用や、さまざまなプログラミング言語、アップデートへの対応を考えると、分散したシステムを統合して形成するようなアーキテクチャーが適していると考えられます[7]。

この変革は、スマートフォンのサーバーサイド側アプリケーションやIoT端末化する自動運転車を支える交通システムなど、バックエンドのシステムのあり方にも大きな影響を及ぼします。理由は、人々の生活や企業活動に深く根ざした社会生活の基盤となるためです。そのような第四次産業革命を推進する情報システム基盤には、次のような特性が求められると考えられます。

- 社会活動に直結して24時間7日間無停止
- ラッシュ時の需要増に反応してスケール
- 拠点とクラウドの計算資源の運用を共通化
- ベンダー非依存の業界標準の実行基盤
- 継続して検証を重ね成果を積み重ねる継続的開発とデリバリー
- 大規模で複雑なマイクロサービスの安定運用
- 大量のデータが生成され蓄積・活用

このようなシステム基盤は、少数の市場支配的な大手ベンダーに依存することは許されません。業界横断のコミュニティの協力で開発されるOSSこそが、システム実現の鍵になると確信しています。

▶▶ 2. CNCFとは

前述のような第四次産業革命が始まりを告げる中、約5年前に「Cloud Native Computing Foundation」(以下、CNCF)が設立されました。この組織は、非営利団体「Linux Foundation」の下部組織として、クラウドネイティブ・コンピューティングの普及と発展を目的として、約20社の創立メンバーで始めました。創立メンバーには、IBM、Google、レッドハット、VMwareなども含まれています[8]。つまりIBMは、創立時からCNCFに参加して活動を支援してきているのです。

2019年には、新たに173社のメンバーがCNCFへ参加し、2020年1月の時点で500社を超えました。これまでにAWSなど超大手のクラウド・ベンダーや日本の大手

ITベンダーも参加しています[9]。クラウドネイティブ・コンピューティングの市場規模が大きいと予測されることから、IT業界の主要な企業からの支持があるのです[10]。

このCNCFが主催するカンファレンス「KubeCon」のセッションには、第五世代携帯電話(5G)、交通システムに関連するもの、人工知能技術を使用したシステム構築などがあります。このような流れから、CNCFによって、第四次産業革命を推進する情報システム基盤が形づくられていくと確信しています。

▶▶ 3. クラウドネイティブ・コンピューティングとは

多くの支持を集めるCNCFには、「CNCF Cloud Native Definition v1.0」と呼ばれるクラウドネイティブ・コンピューティングが目指す方向性が掲げられています[11](表1)。この内容を確認して、CNCFの活動から提供されるOSSについて取り上げていきます。

表1. CNCF Cloud Native Definition v1.0

CNCF Cloud Native Definition v1.0

クラウドネイティブ技術は、パブリック・クラウド、プライベート・クラウド、ハイブリッド・クラウドなどの近代的でダイナミックな環境において、スケラブルなアプリケーションを構築および実行するための能力を組織にもたらします。このアプローチの代表例に、コンテナ、サービスメッシュ、マイクロサービス、イミューダブル・インフラストラクチャ、および宣言型APIがあります。

これらの手法により、回復性、管理力、および可観測性のある疎結合システムが実現します。これらを堅牢な自動化と組み合わせることで、エンジニアはインパクトのある変更を最小限の労力で頻繁かつ予測どおりに行うことができます。

Cloud Native Computing Foundationは、オープンソースでベンダー中立プロジェクトのエコシステムを育成・維持して、このパラダイムの採用を促進したいと考えています。私たちは最先端のパターンを民主化し、これらのイノベーションを誰もが利用できるようにします。

CNCF Cloud Native Definition v1.0の最初の段落では、想定される環境と推進手段(アプローチ)の違いが書かれ、中段で実現したい状態、そして下段にCNCFの実現方法が書かれています。中段に書かれたゴールとなる姿は、これまでもITベンダーやクラウド事業者が目指してきた方向と同じです。これまでと大きく変わる点は、上段の推進手段と下段の進め方です。推進手段では、コンテナをはじめとする近年急速に進化した新しい技術が掲げられ、これらの最新技術をOSSとしてベンダー・ニュートラルに推進して、普及させることが宣言されています。

このような宣言にIBMが参画することは、驚くべきことです。つまり50年前のアポロ計画の時代から、ハードウェアとオペレーティング・システムを開発し販売することで事業を展開してきたIBMからすれば、信じられない方向転換です。これは、第四次産業革命で予測される変化の大きさや影響を認識している証です。

ここまで、第四次産業革命として表現される変化と、それに応じた活動を推進するCNCF活動について触れてきました。ここからは、CNCFが推進する柱となる主要なアプローチについて触れていきます。

▶▶ 4. コンテナ

コンテナについて説明しているIT業界の記事などには、大型コンテナ船や海運コンテナの写真などが使われ、しばしば一般の読者を混乱させます。IT業界のコンテナの実態はこれとは大きく異なります。しかし、海運コンテナが国際貿易に与えた大きな変革と同様に[12]、IT業界のコンテナはアプリケーションの開発や運用にとって、大きなメ

リットをもたらすことになるでしょう。

IT業界のコンテナは、アプリケーションを実行するために必要なライブラリーなどのコードをパッケージ化したソフトウェアの一式です。例えばJavaアプリケーションでは、アプリケーションのコード本体、それを動作させるために必要なJava VMやサーブレット実行環境、データベース・サーバーへ接続するためのプラグイン、LinuxのOSライブラリー、そして、設定ファイルの一式をまとめて、コンテナの中に収めます[13][14][15]。

コンテナは、コンテナの実行環境のLinuxディストリビューションに関係なく、さまざまなディストリビューションのコンテナを実行することができます。つまり、アプリケーションの実行環境をコンテナ上に移すことで、アプリケーションは、ハードウェアだけでなくOS環境からも隔離され自由度が増すことになります。

コンテナ上のアプリケーションは、OS環境を占有するかのように振る舞うことができます。これまでアプリケーションとデータベースは、それぞれワークロードの特性が異なるために、物理的または仮想的にOS環境を分離して構成してきました。しかし、コンテナの場合も同様に、それぞれはIPアドレスを持ったサーバーと同じように扱うことができるため、アプリケーションのコンテナとデータベースのコンテナをつないで構成します。

しかし、利点はそれだけではありません。コンテナはLinux OSのネームスペース機能によって、アプリケーションから、サーバーに同居する他のプロセスから隔離します。そのため、これまでのように仮想化ソフトウェアによってハードウェアの振る舞いを再現するといった

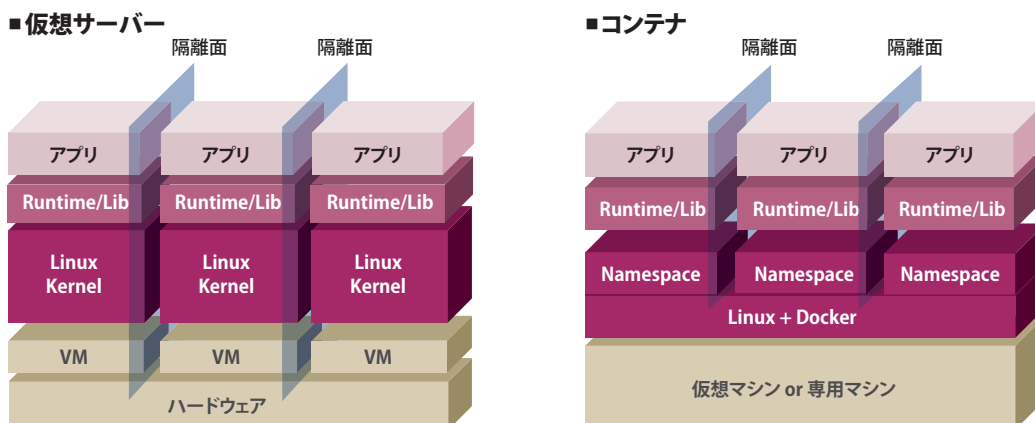


図1. 仮想サーバーとコンテナの違い概念図

複雑でCPU時間を消費する処理を行いません(図1)。このことは、CPU時間のロスがなく、ファイル換算でサイズが小さく、プロセスのように起動が速く、少ないメモリ容量で実行できるという特徴を生み出します。

このコンテナの特性は、異なるサーバー環境への移動を簡単にします。つまり、アプリケーションが依存するソフトウェア一式をコンテナに内包しているため、インストール作業を不要として、コンテナを配置するだけで完全に動作します。このことは、開発、テスト、本番などの環境で、アプリケーションの実行環境が均一となることを意味し、後述のイミュータブル・インフラストラクチャーを取り入れた開発と運用サイクルを実現する基礎となります。

この画期的な発明とも言えるIT業界のコンテナは、OSSとして開発され、だれでも無償で自身のパソコン上で使用できます。これは、たいへん大きな変化と言えます。

▶▶ 5. イミュータブル・インフラストラクチャーによる開発と運用サイクル

第四次産業革命を支えるシステムは、生活基盤として深く根ざすことから、電気や水道の供給と同じように停止が許されません。そして、継続的な改善や保守も必要となります。このような要件に対応する方法が、コンテナ技術の活用とイミュータブル・インフラストラクチャーの実践です。

イミュータブル・インフラストラクチャーは、ITサービス運用におけるアプリケーションのデプロイ、およびサービス管理の手法です[16][17]。この手法では、ア

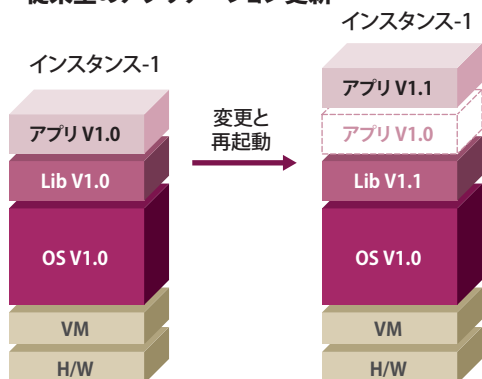
プリケーションの構成要素に変更が発生した場合、本番環境の中で変更するのではなく、変更済みのアプリケーションのコードと依存するソフトウェア一式、すなわち、コンテナを置き換えます。

従来のITサービス運用では、アプリケーションのコードなど構成要素に変更の必要がある場合、本番サービスを中断して行うか、または、サービス継続中にソフトウェアの構成要素の変更が実施されてきました。しかし、イミュータブル・インフラストラクチャーでは、稼働中であっても、前述の一式丸ごとをコンテナとして置き換えます。例えば、ソフトウェアのセットの中で1つの要素に変更が必要であった場合でも、アプリケーションのコードのコンパイルからコンテナのビルドまでの工程、および、テスト工程を再実行して置き換えます(図2)。

つまり、イミュータブル・インフラストラクチャーでは、アプリケーションが依存する基盤となるソフトウェアが随時変更されない、不変であることを意味します。このような手法は、次に挙げるようなさまざまなメリットをもたらします。

- 実行環境の一部を変更するのではなく、変更時は一式を作り直して再デプロイするため、構成のドリフト(揺らぎ)を軽減
- 脆弱性が発見されたソフトウェアの変更と品質の担保が容易となることから、脆弱性による攻撃の脅威を軽減
- 複数の固有設定や複数バージョンのファイルからリストアするのではなく、一式を置き換えるため、予定外の停止時間を軽減

■従来型のアプリケーション更新



■イミュータブル・インフラストラクチャーのアプリケーション更新

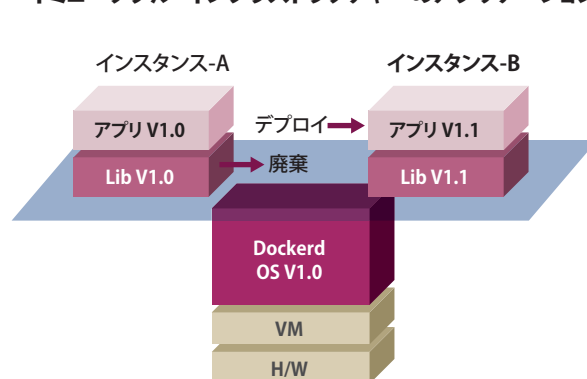


図2. 従来型とイミュータブル・インフラストラクチャーの違い

- 繰り返し一試をデプロイする結果として、十分にテストされ検証済みの共通イメージを構築
- イミュータブル・インフラストラクチャーは、アプリケーションのコードと依存するソフトウェア一試の運用に必要な自動化を促進
- イミュータブル・インフラストラクチャーは、ITの複雑さを軽減して、障害発生を減少させる
- アプリケーションが更新されるたびにテスト済みの最新のインスタンスが開始されるため、サーバーのパッチ適用や構成の変更が不要となり、変更を追跡して管理する必要がない
- アプリケーションの更新後に問題が発生した場合、以前の既知の正常なインスタンスにロールバックが簡単
- 個々のサーバー内のコンポーネントを変更しないため、予測できない動作やコード変更の意図しない結果が生じる可能性を削減

イミュータブル・インフラストラクチャーを導入する効果は、アプリケーションのコードと依存するソフトウェア一試を形成し、デプロイのサイクルを繰り返すことから得られます。言い方を変えると、アプリケーションをコンテナへ移行するだけではメリットを受けることは少なく、むしろ、それまでのサーバー運用と同じことを実施しているのは欠点となることが多いのです。つまり、アプリケーションのコンテナへの移行は、イミュータブル・インフラストラクチャーの概念を取り入れた開発と運用サイクルを回すことによって、真の恩恵を受けることができます。

▶▶ 6. コンテナ・オーケストレーター「Kubernetes」

イミュータブル・インフラストラクチャーを実践しながら、コンテナを本番運用するためのオープンソースのソフトウェアが、CNCFが開発プロジェクトを支援するKubernetesです[18]。この単語には、ギリシャ語で船を操船するための舵輪やキャプテンといった意味があります。Googleの社内システムであるBorgに触発されてOSSとして開発されたもので[19]、コンテナ・オーケストレーターとも呼ばれます。これまで複数のコンテナ・オーケストレーターが提案されてきましたが、パブリック・クラウドの環境ではKubernetesが最も普及しています[20]。

Kubernetesは、コンテナ化されたアプリケーションを本番運用することを目的として開発されました。従来は、サーバーに高可用性やスケラビリティなどのクラスター化のためのソフトウェアを導入して構成していましたが、それらは不要です(図3)。さらに、監視やログ収集を集中管理するための構造を提供します。以下に、Kubernetesの提供する主要な機能を列挙します。

- 複数のコンテナ・ホストをクラスター化して高可用性とスケール
- ストレージ・システムと連携して永続ストレージを動的プロビジョニング
- 複数のコンテナを連携させ組み合わせ運用
- 負荷に応じてコンテナ数を増減するスケール

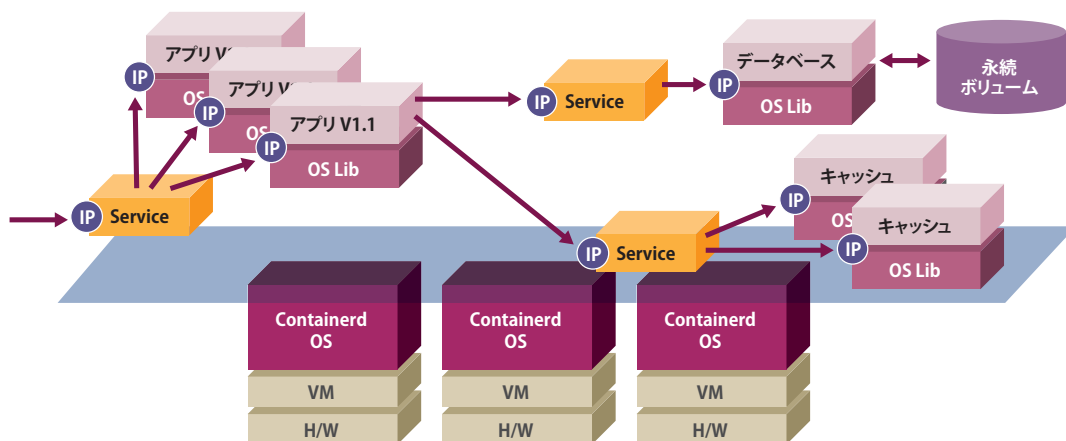


図3. コンテナ・オーケストレーターの動作概念

- 無停止でアプリケーションを更新するロールアウトとロールバック
- 障害等によって失ったコンテナを自己修復する可用性
- Kubernetesのクラスターを仮想化して運用
- メトリックスとログの集中管理アーキテクチャー

ここに挙げたような機能を提供するコンテナ化アプリケーションのプラットフォームがKubernetesです。

CNCFが提供するOSSであるKubernetesは、ITベンダーが製品化する前の上流を意味して、「アップストリームKubernetes」と呼ばれます。最先端の機能を利用できる一方で、リリースサイクルが3カ月ごとで保守期間が9カ月と短く、企業内での本番運用には高度なスキルが要求されます。これに対してレッドハットが製品化したKubernetesが「Red Hat OpenShift Container Platform」(以下、OpenShift)であり、以下の特徴が加えられています。

- OpenShiftは、イミュータブル・インフラストラクチャーを実践する機能が統合
- OpenShiftは、企業が運用するのに適した長期のサブスクリプション・サポート
- レッドハット認定コンテナなど企業利用に適する脆弱性対応をしたセキュリティーを提供

このような理由から、オンプレミスでKubernetesを運用する際に、OpenShiftが最も利用されています[20]。

7. まとめ

第四次産業革命の入口であるいま、将来のソフトウェアやシステム運用についての要件を考察し、CNCFが推進する革新的なアプローチの柱となる部分について解説しました。そのアプローチは、コンテナとイミュータブル・インフラストラクチャーです。そして、コンテナ化アプリケーションの本番運用環境がKubernetesであり、企業向けに強化とサポートされたのがOpenShiftです。IBMはクラウドネイティブ・コンピューティングの将来展望に注目して、レッドハットをIBMのグループに迎え入れ、今後のお客様のビジネスの成功を支援するべく力を注いでいます。

[参考文献]

- [1] クラウス・シュワブ:第四次産業革命 タボス会議が予測する未来、日本経済新聞社 pp. 148 / 2779 - 265 / 2779 (2016)
- [2] 日立東大ラボ:SOCIETY(ソサエティ)5.0 人間中心の超スマート社会、日本経済新聞社 pp. 480 / 3660 - 557 / 3660 (2018)
- [3] 藤井保文、梶原和啓:アフターデジタル オフラインのない時代に生き残る、日経BP社 pp. 120 / 2453 - 164 / 2453 (2019)
- [4] デジタルトランスフォーメーションに向けた研究会:https://www.meti.go.jp/shingikai/mono_info_service/digital_transformation/20180907_report.html 経済産業省 (2018)
- [5] 中西孝樹:CASE革命 2030年の自動車産業、日本経済新聞出版社 pp.1403 / 3844 - 1431 / 9999 (2018)
- [6] 日高洋祐、牧村和彦、井上岳一、井上佳三:MaaS モビリティ革命の先にある全産業のゲームチェンジ、日経BP社 pp.178 / 3779 - 188 / 3779 (2018)
- [7] 日高洋祐、牧村和彦、井上岳一、井上佳三:MaaS モビリティ革命の先にある全産業のゲームチェンジ、日経BP社 pp.2780 / 3779 - 2792 / 3779 (2018)
- [8] CNCF:New Cloud Native Computing Foundation to Drive Alignment Among Container Technologies(2015), https://www.cncf.io/announcement/2015/06/21/new-cloud-native-computing-foundation-to-drive-alignment-among-container-technologies/
- [9] CNCF:Annual Report(2019), https://www.cncf.io/cncf-annual-report-2019/
- [10] CNCF:Cloud Native Interactive Landscape (2019), https://landscape.cncf.io/
- [11] CNCF:Cloud Native Definition v1.0(2018), https://github.com/viktorklang/cncf-toc/blob/master/DEFINITION.md
- [12] JSA日本船主協会:コンテナ船:国際海上物流を一変させたユニークなアイデア 海運雑学ゼミナール https://www.jsanet.or.jp/seminar/text/seminar_177.html
- [13] docker:What is a Container?(2020), https://www.docker.com/resources/what-container
- [14] Red Hat:What's a Linux container?(2020), https://www.redhat.com/en/topics/containers/whats-a-linux-container
- [15] Google:What are Containers and their benefits(2020), https://cloud.google.com/containers/
- [16] O'REILLY:An introduction to immutable infrastructure(2015), https://www.oreilly.com/radar/an-introduction-to-immutable-infrastructure/
- [17] Josha Stella:Immutable Infrastructure Considerations for the Cloud and Distributed Systems, O'REILLY (2016)
- [18] CNCF:Kubernetes Production-Grade Container Orchestration (2020), https://kubernetes.io/
- [19] InfoQ:GoogleがBorgの詳細を公開 (2015), https://www.infoq.com/jp/news/2015/04/google-borg/
- [20] SCSK:Sysdig 2019 Container Usage Report: Kubernetesとセキュリティーの新たな洞察 (2019), https://www.scsk.jp/sp/sysdig/blog/sysdig/sysdig_2019_container_usage_report_kubernetes.html



日本アイ・ビー・エム株式会社
クラウド & コグニティブ・ソフトウェア事業本部
テクニカル・セールス & エバンジェリスト 担当部長

高良 真穂
Takara Maho

2002年、日本IBM入社。IBM Cloudを活用したコンテナとKubernetes / OpenShiftの普及のため研修講師、勉強会、活用支援、執筆活動を進めている。著書「15Stepで習得 Dockerから入るKubernetes コンテナ開発からK8s本番運用まで」(StepUp!選書)。