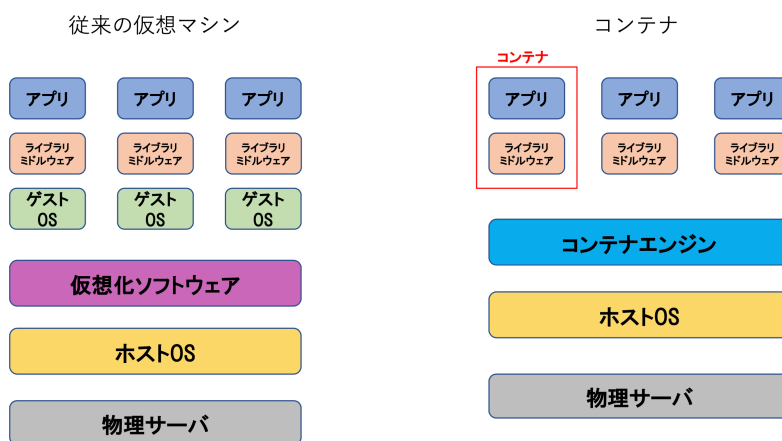


Docker, Kubernetes

1 仮想化マシンとコンテナ

仮想環境は、「安全で効率的なアプリの開発や運用」を行うために使用する。チームでアプリを開発する際、様々なケースでアプリが使われるが、1つのサーバやPC上に必要とされる仮想環境を構築しておくことで、様々な用途で自由にアプリを使用できるようになる。

コンテナとは、サーバ内を整理してアプリケーションや Web の開発・管理を効率的に行えるようにする仮想化技術である。従来の仮想化マシンは、1つの仮想環境の中に必要なものを OS から全て作っていたが、コンテナ環境では OS 周辺の環境は共通で使用し、アプリが使用する CPU、メモリ、ファイルなどは別々にまとめたコンテナとして管理する。これにより、OS 関連でリソース削減ができ、軽量のサーバでも仮想環境が作成できるという大きなメリットに繋がる。



1.1 Docker

Docker は LinuxOS の元でコンテナを使える仕組みの一つで、主にサーバで使われる。Docker はミドルウェアのインストールや各種環境設定をコード化 (Dockerfile) して管理するため、開発環境によるバージョンのずれ防止や開発環境準備の短縮化をすることが可能になる。Docker は基本 Linux で動くが、Windows や Mac でもデスクトップ版が存在し、仮想の Linux 環境を作って Docker Engine が使える。

1.2 Kubernetes

Kubernetes は Docker などのコンテナ仮想化ソフトウェアを管理、および自動化するためのオープンソースのプラットフォームである。コンテナには、アプリケーションを実行する機能はあるが、コンテナを管理したり、他のサーバと連携させたりする機能はなく、それを解決するのが Kubernetes である。Docker は 1 台の物理マシンの上にコンテナを作り実行していくのに対し、Kubernetes は複数の物理マシンを扱うことが前提である。Kubernetes 内で作ったマニフェストファイルに従い、複数の物理マシンにおけるコンテナの作成や管理の煩雑さを自動的に処理する。

2 現在の進捗

- ・ Docker と Kubernetes の基本を学んだ。(DockerDesktop でのコンテナの操作、Kubernetes のマニフェストファイルの記述など)
- ・ Kubernetes のチュートリアルとして、Redis を使用した PHP ゲストブックアプリケーションのデプロイを実行した。[1]

3 今後の課題

- ・ 研究分野への活用を考える (過去の論文や制作物などから知見を得る)。
- ・ Docker と Kubernetes を使ったアプリケーションなどや、使用例から更なる学習をする。
- ・ Kubernetes が使用されているマイクロサービスについて学習する。

参考文献

- [1] Kubernetes. <https://kubernetes.io/ja/docs/tutorials/>