

用語集

クラウドネイティブの 用語

cloud-native container host container state DevOps developer service broker Gemfire cloud hyper-converged i EDAP Lightwave linkerd **M** M NOS OpenTracing opinionated **P** platform operator descriptive platform private cloud Prom repackaging runC **S** scheduler service service disco Cloud Data Flow State set swarm **T** tag the cloud traditional app ID Active Directory AKS API server Azure Container Registry **B** build cluster CNCF CNI Concourse CoreDNS container containerize co continuous integration continuous delivery continuous deployment co digital transformation Docker Docker Swarm **E** elastic ELK stack etcd **F** tes Engine Greenplum Database GRPC **H** Hadoop Harbor Helm Char re as a service (IaaS) ingress **J** Jaeger JSON **K** K8s KaaS Kafka kops segmentation Minikube MongoDB MySQL **N** namespace NodePort N S platforms platform as a service platform developer platform opera rabbitMQ RBAC Redis registry replica set repository refactoring repa (SDDC) software as a service (SaaS) Spanner Spark spec Spring Clo **W** workload **X** XML **Y** YARN YAML **Z** ZooKeeper 12-factor app **A** AC

はじめに

この用語集には、クラウドネイティブ分野の主な用語とその意味が記載されています。通常の辞書のように汎用的な定義を示すのではなく、用語の意味を平易な言葉で説明するとともに、関連する重要なテクノロジーも紹介しています。一部の用語は、使用法、状況、視点、文脈によって意味が異なります。



用語一覽



ACID 8
Active Directory 8
Admiral 8
agile software
development
(アジャイル
ソフトウェア開発)9
AKS 9
API-first
(API ファースト)9
API server (API サーバ)9
Azure Container Registry 9

A

B

build (ビルド)10
BOSH 10

Calico 10
Cassandra 10
CI/CD 10
Clarity 11
cloud computing
(クラウド
コンピューティング)11
Cloud Foundry
Container Runtime 11
cloud infrastructure
(クラウド
インフラストラクチャ)11
cloud-native applications
(クラウドネイティブ
アプリケーション)12
Cloud Spanner 12
cluster (クラスタ)12
CNCF 12
CNI 13
Concourse 13
container (コンテナ)13
Container as a Service 13
container host
(コンテナ ホスト)14
container registry
(コンテナ レジストリ)14
containerize (コンテナ化)14
containerized application
(コンテナ化された
アプリケーション)14
continuous delivery
(継続的デリバリ)14
continuous deployment
(継続的デプロイ)14
continuous integration
(継続的インテグレーション)14
controllers (コントローラ)14
converged infrastructure
(コンバージド
インフラストラクチャ)15
CoreDNS 15
CredHub 15

C

D

day one(導入初日)15
day two(導入後)15
desired state
(望ましい状態)15
developer-ready
infrastructure
(Developer-Ready
インフラストラクチャ)15
DevOps 16
Diego 16
digital transformation
(デジタルトランス
フォーメーション)16
Dispatch 16
Docker 17
Docker Swarm 17

E

elastic(伸縮自在な)17
ELK stack
(ELK スタック)17
etcd 17

F

fault tolerance
(フォルトトレランス)18
Flannel 18
Fluentd 18
Function as a Service 18

G

GCP Open Service
Broker 19
GemFire 19
Go 19
Google Cloud
Platform 19
Google Kubernetes
Engine 19
Greenplum Database 19
GRPC 19

H

Hadoop 20
Harbor 20
Hatchway 20
Helm Chart 20
Horizontal Pod
Autoscaler 21
hybrid cloud
(ハイブリッドクラウド)21
hyper-converged
infrastructure
(ハイパーコンバージド
インフラストラクチャ)21

I

image(イメージ)21
Infrastructure as a
Service(IaaS)21
ingress 22
Istio 22

J

Jaeger 22
JSON 22

K

K8s 23
KaaS 23
Kafka 23
kops 23
kubect1 23
kubelet 23
Kubernetes 24
Kubo 24

L

LDAP 24
Lightwave 24
linkerd 25

MANO 25
memcached 25
micro segmentation
(マイクロセグメンテーション)25
microservices
(マイクロサービス)26

M

Minikube 26
MongoDB 26
Multicloud
(マルチクラウド)26
MySQL 26

namespace
(ネームスペース)27
namespace isolation
(ネームスペースの分離)27
network functions
virtualization
(ネットワーク機能の仮想化)27

N

Node.js 27
NodePort 27
NoSQL 28
NSX 28

OCI 28
OpenStack 28
OpenTracing 28
opinionated platform
(固定型プラットフォーム)28
orchestration
(オーケストレーション)29
overlay network
(オーバーレイネットワーク)29

O

PaaS 29
PAS 29
PCF 29
Photon OS 29
PKS 29
platforms
(プラットフォーム)30
Platform as a Service
(PaaS)30
platform developer
(プラットフォーム開発者)31
platform operator
(プラットフォームオペレーター)31
platform reliability
engineer
(プラットフォーム信頼性エンジニア)31
pod(ポッド)31
PostgreSQL 31
prescriptive platform
(処方型プラットフォーム)32
private cloud
(プライベートクラウド)32
Prometheus 32
pull(プル)32

P

Q

quality of service
(サービス品質)33

RabbitMQ 33
RBAC 33
Redis 33
refactoring
(リファクタリング)34
registry(レジストリ)34
repackaging
(リパッケージ)34
replatforming
(プラットフォームの刷新)34
replica set
(レプリカセット)34
repository(リポジトリ)34
rkt 34
runC 34

R

S

scheduler (スケジューラ) 35
service (サービス) 35
service discovery
(サービス検出) 35
service mesh
(サービスメッシュ) 35
sidecar (サイドカー) 36
Software as a Service
(SaaS) 36
Software-Defined Data
Center (SDDC) 36
Spanner 36
Spark 37
spec (仕様) 37
Spring 37
Spring Cloud
Data Flow 37
StatefulSet 37
Swarm 37

T

tag (タグ) 38
Tern 38
Terraform 38
the cloud (クラウド) 38
traditional application
(従来型アプリケーション) 38
Twelve-Factor App 38

U

UID 39
underlay network
(アンダーレイ
ネットワーク) 39

V

Vagrant 39
virtual infrastructure
manager
(仮想インフラストラクチャ
マネージャ) 39
VMware Integrated
OpenStack 39
Volume (ボリューム) 40
vSphere Cloud
Provider 40
vSphere Integrated
Containers 40

W

workload (ワークロード) 41

X

XML 41

Y

YAML 42
YARN 42

Z

ZooKeeper 42

A

ACID

Atomicity (不可分性)、Consistency (一貫性)、Isolation (独立性)、Durability (永続性)の略。停電やシステム障害が発生した際、データベース トランザクションの信頼性を保証するために必要とされる特性です。

Active Directory

Microsoft Active Directory (AD) は、ユーザーを認証し、PC、サーバ、ストレージ システム、アプリケーション、その他リソースへのアクセスを制御するディレクトリ サービスの 1 つです。Active Directory ドメイン コントローラは、Kerberos Key Distribution Center (KDC) と LDAP サーバを組み合わせて認証と承認を行います。ユーザーの ID の認証には、安全性の高い Kerberos プロトコルまたは従来の NT LAN Manager (NTLM) が使用されます。リソースへのアクセスを承認するときは、通常、Privilege Attribute Certificate (PAC) が使用されます。PAC は Kerberos チケット内のデータ構造であり、グループ メンバーシップ、セキュリティ ID、ユーザーのプロファイルに関するその他の情報が記載されています。「LDAP」を参照。

Admiral

VMware のオープン ソース プロジェクトの 1 つで、コンテナ管理ポータルです。DevOps チームなどがコンテナのプロビジョニングと管理を行うためのユーザー インターフェイスを提供します。Admiral では、Docker Compose を使用して複数のコンテナを 1 つのアプリケーションにまとめることができます。vSphere Integrated Containers 上で Admiral を使用すると、コンテナ ホストを管理したり、キャパシティの割り当てなど、ホストの使用状況に対してガバナンスを適用したりすることができます。Admiral では、コンテナ インスタンスに関するメトリックやその他の情報を表示することもできます。「vSphere Integrated Containers」を参照。

agile software development (アジャイル ソフトウェア開発)

組織横断型チームによる、イテレーション、インクリメンタル、コラボレーションを重視した開発の手法。チームを最適化することにより、フィードバック、問題点、新たな情報によって生じる要件変更にすばやく対応できます。

AKS

Azure Container Service (AKS) は Microsoft のマネージド Kubernetes サービスであり、Azure で実行されます。

API-first(API ファースト)

API をアプリケーションの基礎とするソフトウェア開発のガイドライン。ソフトウェア開発プロジェクトの初期段階から、クライアントアプリケーションやサービスで使用する API の構築に重点を置き、すべての機能要件が API によって満たされるようにします。

API server(API サーバ)

Kubernetes では、API サーバは、REST リクエストを受け付けて API オブジェクト(ポッド、サービス、レプリケーションコントローラなど)のデータを処理するフロントエンドとして動作します。

Azure Container Registry

ジオレプリケーション機能を含む、Microsoft のプライベート イメージレジストリ。

B

build(ビルド)

Docker では、Dockerfile を使用して Docker イメージを作成するプロセスを指します。CI/CD パイプラインの文脈では、アプリケーションを含むバイナリファイルのセットなどのアーティファクトを生成するプロセスを指します。

BOSH

大規模な分散システムのリリース エンジニアリング、展開、ライフサイクル管理を統合するオープン ソース システム。BOSH は、システムの監視と障害からのリカバリを実行し、ゼロまたは最小限のダウンタイムでソフトウェアをアップデートします。Kubernetes がコンテナ化されたアプリケーションの望ましい状態を維持するのと同様、BOSH は、Kubernetes 自体を含め、アプリケーションが実行されるインフラストラクチャの望ましい状態を維持します。

C

Calico

ソフトウェアベースのレイヤー 3 ネットワーク ファブリックをコンテナに提供するオープン ソース プロジェクト (旧称 Project Calico)。Kubernetes、OpenStack などのクラウド プラットフォームと統合できます。

Cassandra

NoSQL データベースの 1 つである Apache Cassandra は、汎用ハードウェアに分散された構造化データを管理します。一般的なユースケースには、推奨エンジンやパーソナライゼーションエンジン、製品カタログ、プレイ リスト、不正検出、メッセージ分析があります。

CI/CD

Continuous Integration (継続的インテグレーション) と Continuous Delivery (継続的デリバリー) のパイプライン、または Continuous Integration (継続的インテグレーション) と Continuous Deployment (継続的デプロイ) のパイプライン。多くの場合、どちらを表すかは文脈によって決まります。「continuous integration (継続的インテグレーション)」、「continuous deliver (継続的デリバリー)」、「continuous deployment (継続的デプロイ)」を参照。

Clarity

VMware のオープン ソース フレームワークの 1 つ(旧称 Project Clarity)。Web アプリケーションにおいて、統合されたインタラクティブなユーザー環境を創出するためのデザイン システムです。Clarity には、HTML と CSS のフレームワーク、ユーザー環境のガイドライン、Angular コンポーネントが含まれます。

cloud computing

(クラウド コンピューティング)

インターネット経由でサービス(通常は従量課金)として提供される、柔軟なオンデマンドの共有コンピューティングリソースとサービスの総称。計算処理、ストレージ容量、データベース利用、分析、ソフトウェア アプリケーションなどが含まれます。一般に、クラウドコンピューティングを提供する組織をクラウド プロバイダーといいます。クラウドコンピューティングの 3 つのサービスモデルである IaaS、PaaS、SaaS の定義も参照。クラウドコンピューティングの正式な定義と分類については、『The NIST Definition of Cloud Computing』(<https://csrc.nist.gov/publications/detail/sp/800-145/final>)を参照してください。

Cloud Foundry Container Runtime

Cloud Foundry Container Runtime (CFCR)は、BOSHを使用してKubernetesの展開と管理を行うためのオープンソースプロジェクト。旧称 Kubernetes on BOSH (Kubo)。CFCRの詳細については、CloudFoundry.org を参照してください。「BOSH」も参照。

cloud infrastructure

(クラウド インフラストラクチャ)

サーバ、仮想マシン、ストレージシステム、ネットワークなど、クラウドコンピューティングと IaaS を提供するために必要なコンポーネントで構成されます。クラウド インフラストラクチャは、ハイブリッドクラウドやプライベートクラウドでクラウドコンピューティングサービスを提供する際の基本的な構成要素となります。

C

cloud-native applications

(クラウドネイティブ アプリケーション)

一般的には、クラウドで分散アプリケーションとして実行するために開発され、最適化されたアプリケーション。より具体的には、Cloud Native Computing Foundation では、「モダン」アプリケーションとも呼ばれるクラウドネイティブアプリケーションの特性を次のように定義しています。

- コンテナ化により、再現性、透明性、リソース分離を実現。
- オーケストレーションにより、リソース使用率を最適化。
- 複数のマイクロサービスに分割することで、変更、管理、拡張を容易化。

ただし、実際の定義は組織によって異なります。たとえば Dell EMC は、クラウドネイティブ アプリケーションを、「オープンな共通の標準を使用し、動的な特性と優れた拡張性を備えた次世代の分散アプリケーション アーキテクチャ」と定義しています。

クラウドネイティブ アプリケーションは、通常、CaaS(サービスとしてのコンテナ)または PaaS(サービスとしてのプラットフォーム)上で開発、展開されます。「Twelve-Factor App」も参照。

Cloud Spanner

グローバルに分散され、一貫性に優れたデータベース サービス。リレーショナルデータベース構造のメリットと非リレーショナル データベースの拡張性を兼ね備えています。

cluster(クラスタ)

3 台以上の仮想マシンまたは物理コンピューターが互いに接続され、1つのシステムを構成している状態。クラスタを構成するコンピューターをノードといます。クラスタ上のアプリケーションは複数のノードで実行されるため、通常、分散アプリケーションとなります。高い可用性、フォルト トレランス、スケーラビリティを提供するクラスタは、クラウド コンピューティングの重要な構成要素です。

CNCF

Cloud Native Computing Foundation の略。Linux Foundation の傘下に設立されたオープン ソース プロジェクト CNCF は、Kubernetes をはじめ、Prometheus、OpenTracing、Fluentd、Linkerd などの重要なオープン ソースプロジェクトをホストしています。VMware は、Linux Foundation および Cloud Native Computing Foundation のメンバーです。

CNI

Container Network Interface の略。CNCF がホストするオープン ソースプロジェクトの1つであり、Linux コンテナでネットワーク インターフェイスを設定する際に役立つ仕様とライブラリを提供します。

Concourse

Pivotal Cloud Foundry やその他のプラットフォームと連携し、継続的インテグレーションと継続的デリバリを実現するシステム。Concourse を利用することで、企業の開発チームは迅速かつ頻繁にソフトウェアをリリースできます。Concourse に関連する「CI/CD」の「D」は、「deployment (展開)」ではなく「delivery (デリバリ)」を表しています。Concourse は、コードを頻繁にコミットする際のテストとパッケージングを自動化します。「CI/CD」を参照。

container(コンテナ)

アプリケーションの依存関係および実行命令をアプリケーションとともにパッケージングする、可搬性のある実行可能フォーマット。イメージとも呼ばれます。コンテナ イメージを実行すると、分離された専用のアプリケーション、ファイルシステム、ネットワークを使用して、コンピューターまたは仮想マシン上でプロセスとして実行されます。柔軟性に優れ、ポータブルかつ予測可能な方法でアプリケーションをパッケージング、配布、変更、テスト、実行できることから、コンテナを使用するケースが増えています。コンテナを使用することで、ソフトウェアの開発と展開が迅速化されます。

Container as a Service

Container as a Service (サービスとしてのコンテナ) プラットフォームでは、Kubernetes やほかのオーケストレーション フレームワーク (Mesos、Docker Swarm など) を使用して、コンテナ化されたアプリケーションをビルド、展開、管理できます。



container host (コンテナ ホスト)

コンテナ実行用として最適化された Linux オペレーティング システム。CoreOS、VMware の Project Photon OS などがあります。

container registry (コンテナ レジストリ)

「registry (レジストリ)」を参照。

containerize (コンテナ化)

アプリケーションをコンテナにパッケージングすること。

containerized application (コンテナ化されたアプリケーション)

1 つ以上のコンテナで実行できるようにパッケージングされたアプリケーション。

continuous delivery (継続的デリバリ)

アプリケーションまたはアプリケーションの一部をパッケージ化して検証し、本番環境へリリースできるようにすること。

continuous deployment (継続的デプロイ)

アプリケーションまたはアプリケーションの一部を本番環境へ自動的に展開すること。

continuous integration (継続的インテグレーション)

複数の開発者やチームが作成したソースコードを結合してアプリケーションを作成し、テストすることを継続的に行うこと。

controllers (コントローラ)

Kubernetes では、コントローラーとは、クラスターの管理に関連付けられたルーチン タスクを実行するために Kubernetes Controller Manager によって開始されるプロセスを指します。

D

converged infrastructure (コンバージド インフラストラクチャ)

サーバ、データ ストレージ デバイス、ネットワーク機能、仮想化、管理ソフトウェア、オーケストレーション、アプリケーションなど、IT を支える各種のインフラストラクチャ要素を1つにまとめるテクノロジー。「hyper-converged infrastructure(ハイパーコンバージドインフラストラクチャ)」を参照。

CoreDNS

オープン ソース プロジェクトの1つ。Kubernetes、etcd、Prometheus、その他のソフトウェアを統合し、DNS とサービス検出をプラグインとして実行できるようにします。CoreDNS は CNCF にホストされています。

CredHub

Cloud Foundry のコンポーネントの1つで、パスワード、証明書、認証局、キーなどの認証情報を管理し、安全に保存します。CredHub は BOSH を使用して展開されます。「Pivotal Cloud Foundry」, 「BOSH」を参照。

day one(導入初日)

「deployment(展開)」を参照。

day two(導入後)

「post-deployment operations(導入後の運用)」を参照。

desired state(望ましい状態)

Kubernetes の大きなメリットは、アプリケーションの望ましい状態が自動的に維持されることです。この望ましい状態は、管理者またはプラットフォーム オペレーターが指定します。

developer-ready infrastructure (Developer-Ready インフラストラクチャ)

VMware vSphere、VMware NSX、VMware vSAN、VMware vRealize Operations は Software-Defined Data Center (SDDC) の基盤となります。たとえば、VMware SDDC 上で VMware Pivotal Container Service または Pivotal Cloud Foundry を実行すると、クラウドネイティブ アプリケーションの構築と運用に役立つセルフサービス対応の俊敏なインフラストラクチャ、つまり Developer-Ready インフラストラクチャを実現できます。

D

DevOps

適切な方法で確実かつ継続的にソフトウェアを提供するには、IT 部門と開発者が連携する必要があります。開発と展開を効率化し、自動化するため、開発者と IT 部門が密に連携して運用にあたることを「DevOps」といいます。DevOps は、クラウドネイティブアプリケーションの開発と展開を促進する重要な手法です。

Diego

Pivotal Cloud Foundry のコンテナ管理システム。

digital transformation

(デジタル トランスフォーメーション)

コンピューティング リソースの使用、組織のプロセス、ソフトウェア開発手法を最適化することで、企業の適応力、生産性、イノベーション、競争力を高め、グローバルな発展を促進すること。多くの場合、デジタル トランスフォーメーションは、クラウド コンピューティング、モバイル デバイス、ソーシャル メディア、ビッグデータ分析といった最新テクノロジーの導入を伴います。具体的には、

クラウドネイティブ関連のテクノロジーと手法(コンテナ、Kubernetes、マイクロサービス、コンテナ プラットフォーム、DevOps、CI/CD パイプラインなど)を統合してデジタル トランスフォーメーションを実現します。

Dispatch

VMware のオープン ソース フレームワークの 1 つ。機能を使用してビルドされたサーバレス アプリケーションとサービスを管理します。Dispatch はサーバレス ソリューションであるため、柔軟なドライバ インターフェイスを通じて 1 つ以上の Function as a Service (FaaS) の実装と統合されます。Dispatch は、Photon OS 上でビルドされた機能のランタイムとコンテナを管理します。「Function as a Service」を参照。

E

Docker

広く利用されているコンテナ フォーマットの1つ。Docker は、ソフトウェアをパッケージ化して移植するための標準フォーマットを定義しています。これは、貨物輸送の標準規格としてISO コンテナが定義されているのに似ています。Docker イメージのランタイム インスタンスとしてのコンテナは、3つのパーツで構成されます。

- Docker イメージ
- イメージを実行する環境
- イメージを実行するための命令のセット

Docker Swarm

スタンドアロンで使用できる Docker 用のネイティブなクラスタリング ツールの名前。Docker Swarm では、複数の Docker ホストを結合し、それらを1つの仮想 Docker ホストとみなします。標準の Docker API が利用できるため、すでに Docker で動作するツールであれば、複数のホストへ透過的に拡張できます。

elastic(伸縮自在な)

需要の変動に合わせて、リソースやサービスを動的に拡張または縮小できること。

ELK stack(ELK スタック)

ELKスタックはElasticsearch、Logstash、Kibanaで構成されます。これら3つのオープンソースプロジェクトを組み合わせることで、データを収集、検索、分析、可視化するためのプラットフォームを提供します。Elasticsearchは、非構造化データ、構造化データ、時系列データに対してクエリを実行できる分散型の検索/分析エンジンです。Logstashでは、非構造化データを収集して処理し、そのデータをElasticsearchなどほかのアプリケーションへ転送できます。Kibanaは、データをグラフィックやマップとしてダッシュボードに表示する可視化エンジンです。

etcd

キーと値の分散型ストア。Kubernetesはetcdを使用して、その状態と設定に関するデータを保存します。

F

fault tolerance (フォルト トレランス)

コンポーネントで障害が発生した場合、システムを引き続き適切に稼働させるための機能。

Flannel

Kubernetes で実行されるコンテナ向けにオーバーレイ ネットワークを提供します。「overlay(オーバーレイ)」を参照。

Fluentd

統合ログ環境向けのデータ コレクター。CNCF にホストされ、クラウドネイティブアプリケーションと連携します。

Function as a Service

Function as a Service (FaaS) は、従来のようにアプリケーションのバックエンドとしてサーバを管理することなく、アプリケーションの関数を実行、管理できるクラウド コンピューティングモデルです。「サーバレス」という言葉とともに利用が拡大しています。通常、関数はイベントに応答するため、IoT では FaaS が有効なイベント処理方法として用いられています。次に例を示します。朝、起床時にスマート ウォッチがあなたの血糖値をモニタリングします。血糖値が一定レベルに達すると、クラウド上でベンダーが管理するリモートサーバにスマート ウォッチがイベントを送信します。このサーバは FaaS フレームワークを使用しており、ある関数でイベントを処理し、ネットワークに接続されたあなたの電子レンジにリクエストを送信します。それがオートミールの調理を開始するトリガーになります。AWS Lambda は商用サーバレス プラットフォームの一例です。OpenFaaS は、Docker と Kubernetes を使用してサーバレス ファンクションを構築する FaaS フレームワークの一例です。「Dispatch」を参照。

G

GCP Open Service Broker

アプリケーションがどこからでも Google Cloud API へアクセスできるようにします。

GemFire

Pivotal Gemfire は、運用データを圧縮してメモリに格納し、大量のデータを扱う NoSQL アプリケーションへのスケーラブルで一貫したリアルタイムアクセスを実現する分散データ管理プラットフォームです。

Go

Google が開発したプログラミング言語の 1 つで、クラウドネイティブテクノロジーを構築する開発者に人気があります。Docker と Kubernetes は Go で記述されています。Go は golang と呼ばれることもあります。

Google Cloud Platform

GCP。

Google Kubernetes Engine

Kubernetes を利用してコンテナ化されたアプリケーションを展開および拡張するための管理環境。

Greenplum Database

シェアード ナッシングの超並列処理アーキテクチャを採用した、ACID 準拠のトランザクション データベース。Pivotal Greenplum は SQL 標準に準拠しています。業界標準のビジネスインテリジェンスツール、ETL ツール、Hadoop などと利用できます。分析機能のライブラリ、およびカスタム機能を作成するためのフレームワークを備えている Greenplum は、ビッグデータのデータウェアハウスとして利用できます。

GRPC

CNCF のプロジェクト。分散システムで使用するオープンソースのユニバーサルリモート プロシージャコール(RPC)フレームワークです。GRPC では、バイナリ シリアル化言語 Protocol Buffers を使用してサービスを定義できます。また、さまざまな言語でサービスを提供するためのクライアントスタブとサーバスタブを自動的に生成することもできます。

H

Hadoop

Hadoop は Hadoop Distributed File System (HDFS) と MapReduce で構成されます。HDFS は、Hadoop およびビッグデータに適したスケーラブルなストレージシステムです。MapReduce は、HDFS に保存されているファイルの膨大なデータを分析する処理フレームワークです。Apache Hadoop は Hadoop の無償バージョンとして公開されており、Apache Software Foundation が管理しています。このオープンソース版は、Hortonworks、IBM Open Platform、Cloudera など一部の商用バージョンの基盤となっています。サービスとしての Hadoop プラットフォームもあります。Microsoft は、パブリッククラウド Azure の一部として HDInsight を提供しています。Amazon Elastic MapReduce (EMR) は、Web サービスとしての Hadoop を AWS で提供しています。

Harbor

VMware のオープンソースプロジェクトの1つ(旧称 Project Harbor)。コンテナイメージのリポジトリを管理するセキュアなレジストリです。

Hatchway

VMware のオープンソースプロジェクトの1つ(旧称 Project Hatchway)。VMware vSAN を使用したハイパーコンバージドインフラストラクチャ(HCI)をはじめとする vSphere 環境で、コンテナ用のストレージインフラストラクチャのオプションを提供します。Hatchway を Kubernetes と統合することで、ストレージインフラストラクチャをコードとして使用できます。スナップショット、クローン作成、暗号化、重複排除、圧縮などのデータサービスを、コンテナボリュームのレベルで利用できます。<https://vmware.github.io/hatchway/> を参照。「vSphere Cloud Provider」も参照。

Helm Chart

事前に構成およびカスタマイズされ、再現可能な Kubernetes リソースのパッケージ。「チャート」と呼ばれるこのパッケージは、Helm ツールを使用して管理できます。チャートは Kubernetes アプリケーションの可搬性向上に役立っています。1つのチャートに、データベース、キャッシュ、HTTP サーバ、その他のリソースなど、ウェブアプリケーション全体を含めることができます。

Horizontal Pod Autoscaler

Kubernetes で、サービスに対するリクエストが増加し、管理者が設定したしきい値を超えたときにリソースを追加して対応するための仕組み。

hybrid cloud (ハイブリッド クラウド)

プライベート クラウド リソースとパブリック クラウド リソースなど、複数のデリバリ モデルを組み合わせる最新インフラストラクチャ。

hyper-converged infrastructure (ハイパーコンバージド インフラストラクチャ)

統合する主な IT コンポーネントの種類はコンバージド インフラストラクチャと同じですが、ハイパーコンバージド インフラストラクチャはスケーラブルなラックやアプライアンスを使用するため、管理が容易でパフォーマンスに優れ、柔軟に拡張できます。「converged infrastructure(コンバージド インフラストラクチャ)」を参照。

image(イメージ)

Docker では、コンテナの基本的な要素の1つ。イメージには、コンテナの実行時に使用するためのルート ファイル システムに対する変更内容とそれに対応する実行パラメーターを指定します。通常、イメージは、複数のファイル システムによって階層化されています。イメージには状態がなく、決して変化しません。

Infrastructure as a Service (IaaS)

ストレージ、ネットワーク、コンピューティングのリソースなど、基盤となるIT インフラストラクチャへのアクセスをオンデマンドで提供します。IaaS では、ユーザーが必要になった時点で IT サービスをプロビジョニングし、任意のソフトウェアの展開、実行に利用できます。通常、ユーザーは使用したリソース分のみを料金を支払います。ユーザーは、基盤となるクラウド インフラストラクチャの管理や制御を行いません。「cloud computing(クラウドコンピューティング)」を参照。

I

ingress

Kubernetes では、Kubernetes クラスタに属するサービス (HTTP や HTTPS など) への外部からのアクセスを制御する API オブジェクトを指します。ingress はロードバランシングを実行できます。

Istio

Kubernetes 上のマイクロサービスを接続、管理、保護するためのサービスメッシュを展開するプラットフォーム。Istio は、Kubernetes に展開されたコンテナ化されたアプリケーションを構成するマイクロサービス間のネットワーク通信を傍受して、相互に通信しているマイクロサービスを管理します。「microservices (マイクロサービス)」、「sidecar (サイドカー)」、「service mesh (サービス メッシュ)」も参照。

J

Jaeger

Uber Technologies により、オープンソース ソフトウェアとして公開されている分散トレーシング システム。マイクロサービスベースのアーキテクチャを監視できます。ユースケースとして、分散トランザクションの監視、根本原因の分析、サービスの依存関係分析、パフォーマンスの最適化などがあります。Jaeger は CNCF がホストしています。

JSON

JavaScript Object Notation の略。軽量のデータ交換フォーマット。通常、API 出力などのデータに注釈を付ける際に使用されます。

K

K8s

Kubernetes の略称。

KaaS

Kubernetes as a Service (サービスとしての Kubernetes)。

Kafka

Apache Kafka。流入する膨大なデータを分析できるようにするため、データストリームを分割した後、複数のマシンで構成された分散クラスタに配信します。正式には、パブリッシュ/サブスクライブ型の分散メッセージングシステム。Kafka の重要な役割は、Spark または同様のアプリケーションがデータストリームを容易に処理できるようにすることです。たとえば、Kafka は、データストリーム (複数台のサーバから受信するログ ファイルなど) を集約して「トピック」を生成し、それを Spark Streaming に渡します。Spark Streaming は渡されたデータをリアルタイムで分析します。

kops

Kubernetes Operations の略。

Kubernetes クラスタのインストール、管理、アップグレードを支援するコマンドライン ツールです。

kubectl

コンピューター上にインストールするコマンドライン インターフェイスであり、Kubernetes クラスタを制御、管理するコマンドを実行するのに使用します。

kubelet

ポッドを管理するために Kubernetes クラスタの各ノードで実行されるエージェント。kubelet の動作は PodSpec で指定します。PodSpec は、ポッドを記述する YAML または JSON オブジェクトです。kubelet はさまざまなメカニズム (主に API サーバ) を介して一連の PodSpec を受け取り、それらの PodSpec で定義されているコンテナが正常な状態で実行されるように管理します。

K

Kubernetes

コンテナ化されたアプリケーションの展開と管理を自動化するオーケストレーション システム。仮想マシンまたは物理マシンで構成された分散クラスタ上のコンテナでアプリケーションとそのサービスを実行する際、Kubernetes はすべての移動要素を調整します。これによって、コンピューティングリソースの使用状況の最適化、望ましい状態の維持、オンデマンドでの拡張/縮小を実行できます。Kubernetes は、オーケストレーション フレームワークまたはオーケストレーション エンジンとも呼ばれます。「desired state (望ましい状態)」、「orchestration (オーケストレーション)」を参照。

Kubo

「Cloud Foundry Container Runtime」を参照。

L

LDAP

Lightweight Directory Access Protocol の略。ディレクトリ サービスの情報(特にユーザー名とパスワード)を格納し、その情報にアクセスするための標準プロトコル。アプリケーションは LDAP サーバに接続して、ユーザーとグループを検証できます。

Lightwave

VMware のオープン ソース セキュリティ プラットフォーム Project Lightwave は、ディレクトリ サービス、Active Directory との相互運用、Kerberos 認証、証明書サービスを提供することにより、クラウド プラットフォームを保護します。Lightwave を使用することで、オンプレミス コンピューティング システムでの実績があるセキュリティ ポリシーとベスト プラクティスをクラウド コンピューティング環境に適用できます。具体的には、以下のサービスが含まれます。

- LDAP および Active Directory との相互運用性を備えたディレクトリ サービスおよび ID 管理

M

- Kerberos、SRP、WS-Trust(SOAP)、SAML Web SSO(ブラウザーベースの SSO)、OAuth/OpenID Connect (REST API)などのプロトコルによる認証サービス
- 証明書認証局と証明書ストアを備えた証明書サービス

linkerd

サービス検出、ルーティング、障害処理、クラウドネイティブ アプリケーションの可視化を追加するサービス メッシュ。linkerd は CNCF にホストされています。

MANO

NFV における MANO は、通信ネットワーク リソースの Management and Orchestration(管理とオーケストレーション)を略した言葉です。この意味では、MANO には vRealize Suite、すなわち vRealize Operations、vRealize Automation、vRealize Network Insight、vRealize Log Insight が含まれます。

memcached

コンピューター クラスタの分散メモリにデータをキャッシュするシステム。データベースに対する最近の呼び出しの結果を RAM(ランダム アクセス メモリ)に格納することで、Web アプリケーションのパフォーマンスを向上させます。

micro segmentation (マイクロ セグメンテーション)

VMware NSX のマイクロ セグメンテーション ポリシーでは、たとえば、コンテナ化されたアプリケーションを実行する Kubernetes の名前スペースできめ細かいトラフィックフローパターンを指定できます。マイクロ セグメンテーションを使用すると、ワークロードにセキュリティ要件を適用し、マイクロサービス単位でリソースを分離するルールを作成できます。

M

microservices(マイクロサービス)

アプリケーション開発で使用する「モダン」なアーキテクチャパターンの1つ。マイクロサービスアーキテクチャでは、アプリケーションの機能を目的ごとに小さなプロセスに分割し、各プロセスを個別に開発、テスト、展開、置換、拡張できるようにします。「cloud-native application(クラウドネイティブアプリケーション)」を参照。

Minikube

仮想マシン内または PC 上のローカル環境で、単一ノードの Kubernetes クラスタを実行できるツール。

MongoDB

分散 NoSQL ドキュメント データベース MongoDB では、変更に対応できるスキーマ不要のデータ モデルを使用してデータが保存されます。MongoDB には、セカンダリ インデックス、地理空間検索、テキスト検索が含まれます。一般的なユースケースとして、モバイル アプリケーションへのデータ送信、リアルタイム分析などがあります。

Multicloud(マルチクラウド)

クラウド コンピューティングに対するこのアプローチでは、1つの環境またはアーキテクチャ内で複数のクラウドプロバイダー、プラットフォーム、サービスを組み合わせることを指します。マルチクラウド戦略には、単一ベンダーへの依存度が軽減される、クラウドサービスを停止から保護する、ニーズに合わせてアーキテクチャをカスタマイズできる、ニーズの変化に応じてソリューションを柔軟に切り替えられる、という特徴があります。ただし、複数のクラウドプラットフォームやサービスを使用することで、セキュリティ、ガバナンス、コンプライアンスが複雑化する可能性があります。コンテナの可搬性により、マルチクラウド戦略の実行が容易になります。

MySQL

オープンソースのリレーショナルデータベース管理システム(RDBMS)。Web アプリケーションをはじめ、さまざまな種類のアプリケーションで広く使用されています。独立系ソフトウェアベンダー(ISV)やOEMが配布する多くのソリューションにも組み込まれています。名前の「SQL」は、Structured Query Language(構造化問合せ言語)の略。

N

namespace (ネームスペース)

Linux コンピューター関連でのネームスペースは、システム リソースを分離して仮想化するカーネルの機能。特定のネームスペースに限定されたプロセスは、同じネームスペースに属するほかのリソースやプロセスとのみやり取りできません。

Docker でのネームスペースは、ネットワークやストレージなどのシステムリソースを分離します。

Kubernetes では、多数の仮想クラスタが同じ物理クラスタを基盤としている場合、それらの仮想クラスタをネームスペースと呼んでいます。

namespace isolation (ネームスペースの分離)

コンテナが接続または通信できるリソースを制限します。

network functions virtualization (ネットワーク機能の仮想化)

ネットワーク機能の仮想化(NFV)とは、通信サービスを作成するためにネットワーク ノードの機能を仮想化するという概念を指します。通信業界では、これはクラウドネイティブ デザイン

戦略の一部になっています。NFV-I は、「ネットワーク機能の仮想化インフラストラクチャ」の略称です。NFV が使用されている通信環境では、ほかの機能と組み合わせて通信サービスを提供できる特定の機能として、仮想ネットワーク機能(VNF)があります。VNF マネージャは、VNF を NFV インフラストラクチャと統合してリソースを割り当てるなど、VNF 機能のコレクションを制御します。

Node.js

イベント駆動型の JavaScript ランタイム環境。サーバ上で JavaScript コードを実行し、Web アプリケーション内に動的なコンテンツを生成します。

NodePort

Kubernetes で の NodePort は、Kubernetes クラスタのノードで、外部アクセスを受け付けるポート上で稼動するサービス(Web サーバなど)を表します。

N

NoSQL

NoSQL データベースには、従来型リレーショナル データベースの表形式以外の方法で構造化されたデータが格納されます。NoSQL は、non-SQL、non-relational、not-only SQL とも呼ばれます。NoSQL データベースは、ビッグデータやリアルタイム データの処理で広く利用されています。NoSQL データベースのよく知られた例として、MongoDB、Cassandra、Pivotal Gemfire などがあります。

NSX

VMware NSX は、ソフトウェアベースでネットワークを仮想化するための製品ポートフォリオです。

O

OCI

Open Container Initiative の略。コンテナ関連の業界標準を策定している組織。OCI は、コンテナのフォーマットと実行に関するオープンな業界標準を策定するという明確な目的のもと、Linux Foundation の支援を得て発足されました。OCI は、ランタイム仕様 (runtime-spec) とイメージ仕様 (image-spec) の 2 つで構成されます。VMware は OCI のメンバーです。<https://www.opencontainers.org/> を参照。

OpenStack

クラウド コンピューティングを対象とするオープンソース プラットフォームの 1 つ。コンピューティング、ストレージ、ネットワークのリソース プールを、API とダッシュボードを通じて制御できます。「VMware Integrated OpenStack」を参照。

OpenTracing

分散トレーシングに関する、ベンダーに依存しない標準規格。CNCF にホストされています。

opinionated platform (固定型プラットフォーム)

「prescriptive platform (処方型プラットフォーム)」を参照。

P

orchestration

(オーケストレーション)

コンテナ化されたアプリケーションを自動的に展開、管理、拡張できることから、Kubernetes は「オーケストレーション フレームワーク」または「オーケストレーション エンジン」と呼ばれます。Kubernetes は、リソースの使用率、障害処理、可用性、構成、望ましい状態、スケーラビリティを自動的に管理します。

overlay network

(オーバーレイ ネットワーク)

多くのオーケストレーション システムには、オーバーレイ ネットワークと呼ばれる、ソフトウェアベースのネットワークコンポーネントが含まれます。オーバーレイはアンダーレイの上に配置され、コンテナやホストのライフサイクル全体にわたり、IP アドレスやポートなどのネットワーク機能を提供します。オーバーレイにより、同じ物理ネットワークを使用するアプリケーション間で通信を分離することができます。オーバーレイテクノロジーには、Flannel、Calico、VMware NSX などがあります。「NSX」、「underlay network(アンダーレイ ネットワーク)」、「orchestration(オーケストレーション)」も参照。

PaaS

Platform as a Service (サービスとしてのプラットフォーム)の略。

PAS

Pivotal Application Service の略。PAS(旧称 Elastic Runtime)は、Pivotal Cloud Foundry上でJava、.NET、Nodeのアプリケーションを実行します。

PCF

Pivotal Cloud Foundry の略。クラウドネイティブ アプリケーションの開発と展開を支援するプライベート PaaS(サービスとしてのプラットフォーム)。

Photon OS

VMware のオープン ソース プロジェクトである Project Photon OS は、コンテナの実行環境として最適化された Linux オペレーティング システム。

PKS

Pivotal Container Service の 略。Kubernetes ベースのコンテナ サービス。

P

platforms(プラットフォーム)

コンテナ プラットフォームを使用する際の重要なビジネス目標は、スケーラビリティに優れ、変更、拡張、運用、管理が容易なエンタープライズクラスのソフトウェアを速やかに開発、展開できるようにすることです。次の3種類のプラットフォームが、それぞれ異なるレベルでコンテナ テクノロジーをサポートします。

- 個々のコンテナ インスタンスを実行するためのプラットフォーム。コンテナ インスタンスを実行するプラットフォームでは、コンテナ化されたアプリケーションを開発し、テストできます。ただし、コンテナ化されたアプリケーションと Kubernetes とのオーケストレーションはサポートしていません。また、ツール、データベース、サービスとアプリケーションを統合するためのサービス ブローカーも提供しません。たとえば、VMware vSphere Integrated Containers はコンテナ インスタンス プラットフォームです。
- Container as a Service(サービスとしてのコンテナ)。
- Platform as a Service(サービスとしてのプラットフォーム)。

Platform as a Service(PaaS)

PaaS は、アプリケーションの開発、テスト、実行を行うためのクラウドベースの環境であり、プラットフォームのプロバイダーがサポートまたは提供するプログラミング言語、ライブラリ、サービス、ツールを使用できます。PaaS は、単に「アプリケーションプラットフォーム」とも呼ばれます。その場合、アプリケーション プラットフォームでは、コードを記述するだけでなく、ツールやサービス(データベースなど)を、たとえばマイクロサービスとしてアプリケーションに統合することができます。アプリケーション プラットフォームとも呼ばれるプライベート PaaS の例として、Pivotal Cloud Foundry があります。「Container as a Service」、「Infrastructure as a Service」、「cloud computing(クラウドコンピューティング)」を参照。

platform developer (プラットフォーム開発者)

担当プロジェクトや組織のニーズに合わせて Kubernetes プラットフォーム (またはその他のモダンなプラットフォーム) をカスタマイズするエンジニア。

platform operator (プラットフォームオペレーター)

Kubernetes などのプラットフォームを管理するエンジニア。

platform reliability engineer (プラットフォーム信頼性エンジニア)

Pivotal Cloud Foundry などのプラットフォームを操作するエンジニア。その業務には、ディザスタリカバリ、ネットワーク、ストレージ、モニタリングアプリケーション、サービスブローカーの管理などが含まれます。プラットフォーム信頼性エンジニアの重要な職務として、継続的インテグレーション(CI)と継続的デリバリー(CD)のパイプラインを確立し維持することがあります。「DevOps」を参照。

pod(ポッド)

Kubernetes では、ポッドは、1 つ以上のコンテナを管理できる展開可能な最小単位です。言い換えると、コンテナイメージはポッド内で実行されます。通常、ポッドは、コンテナとそのストレージリソース、IP アドレス、その他のオプションをアプリケーションインスタンスとして 1 つにまとめ、そのアプリケーションインスタンスが Kubernetes 上で実行されます。多くの場合、ポッドでは、コンテナランタイムとして Docker が使用されます。Kubernetes 管理者またはアプリケーション開発者は、YAML ファイルを使用してポッドを指定します。一般にポッドは展開単位で管理されます。「deployment(展開)」を参照。

PostgreSQL

Postgres とも呼ばれます。インターネットに接続する大規模なアプリケーションやデータウェアハウスのデータを安全に保存する、拡張可能なオブジェクトリレーショナルデータベース管理システムです。Postgres は ACID 準拠です。「ACID」を参照。

P

prescriptive platform (処方型プラットフォーム)

アプリケーション プラットフォーム 関連での「処方型プラットフォーム」とは、そのシステムで規定されたフォーマット、パイプライン、メソッドを使用し、複雑さを意識せずにアプリケーションを開発、実行できるプラットフォームを指します。たとえば、処方型コンテナプラットフォームには、スケジューラ、ランタイム エンジン、基盤となるインフラストラクチャとの連携、継続的デリバリーなどが備わっています。処方型プラットフォームは、「opinionated (固定型)」プラットフォームとも呼ばれます。

private cloud (プライベート クラウド)

完全に仮想化されたデータセンター。俊敏性を高める 2 つの重要な機能 (セルフサービスと自動化) を備えています。

Prometheus

Kubernetes で使用するオープン ソースの監視システム。Prometheus は CNCF にホストされています。

pull (プル)

レジストリからローカル キャッシュへコンテナ イメージをダウンロードし、そのイメージに基づいてコンテナを実行できるようにすること。

Q

quality of service (サービス品質)

QoS と表記されることや、ストレージ/ネットワークの流量制限の意味合いで用いられることもあります。

R

RabbitMQ

Advanced Method Queuing Protocol を実装した、オープンソースのメッセージブローカー。アプリケーションが中間プラットフォームを介してデータに接続したり、データを交換したりできるようにします。

RBAC

ロールベースのアクセスコントロール。Kubernetesに関連するRBACは、ロールに基づいてリソースへのアクセスを許可するモジュールです。RBACにより管理者は、Kubernetes APIを使用してアクセスポリシーを動的に設定できます。

Redis

ネットワーク接続されたメモリ内キャッシュにデータセットを保存できる、キーバリューストア。Redisのキーは、文字列、ハッシュ、リスト、セット、ソート済みセット、ビットマップ、ハイパーログを格納できることから、「データ構造サーバ」とも呼ばれます。たとえば、データサイエンティストは、これらのデータ型を操作して積、和、差などの集合演算やランク付けができます。

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

Y

Z

R

refactoring(リファクタリング)

アプリケーションを再設計したり、そのコードを変更したりして、改善すること。たとえば、アプリケーションを複数のマイクロサービスに分割する処理などがこれにあたります。

registry(レジストリ)

コンテナ イメージのリポジトリを含むホスト型サービス。VMware のオープンソース プロジェクト Harbor もレジストリの1つです。

repackaging(リパッケージ)

従来型アプリケーションをコンテナフォーマットに変更すること。

replatforming (プラットフォームの刷新)

より効率的な別のプラットフォームにアプリケーションを移植すること。コンテナを使用する新しいプラットフォームに従来型アプリケーションを移植する場合、プラットフォームの刷新にはリパッケージも含まれます。

replica set(レプリカ セット)

Kubernetes でのレプリカ セットは、ポッドのライフサイクルを管理するコントローラです。「controllers(コントローラ)」を参照。

repository(リポジトリ)

コンテナ関連では、一連のコンテナイメージを指します。リポジトリは、レジストリ サーバを介してほかのユーザーと共有できます。「repo」内のイメージにはラベルを付加できます。

rkt

標準に準拠した CoreOS のコンテナエンジン。「rocket(ロケット)」と発音します。

runC

コンテナを実行するコード モジュール。containerd の一部で、Open Container Initiative (OCI) によって管理されています。「OCI」を参照。

S

scheduler(スケジューラ)

さまざまな種類のワークロード(コンテナ、ジョブ、タスクなど)の展開スケジュールを設定して実行するシステム、またはそれに該当するソフトウェアコンポーネントのモジュール。Microsoft Azure など、ほとんどのパブリック クラウド サービスには、クラウドにジョブを追加できるスケジューラが備わっています。作成されたジョブは、データのバックアップ、ログの消去といったサービスやタスクを呼び出します。

service(サービス)

サービスの定義は文脈によって異なります。Kubernetes では、ポートやロードバランサーを使用してアプリケーション(一連のポッド)にアクセスする方法が記述された API オブジェクトです。

大規模なアプリケーションでは、マイクロサービスがサービスと見なされる場合もあります。たとえば、HTTP サーバはサービスの1つです。

service discovery(サービス検出)

マイクロサービスを使用して構築されたクラウドネイティブ アプリケーションでは、それが実行されているサービスまたはデバイスに動的に割り当てられたネットワーク情報が自動的に検出されます。

service mesh(サービス メッシュ)

コンテナ化されたアプリケーションがサービスまたはマイクロサービスのコレクションとして構築されている場合、そのアプリケーションはサービス メッシュを形成していることになります。サービス メッシュは、IP アドレスやポートの上位に、サービスに接続して通信を管理するためのレイヤーを作成します。サービス メッシュでは、たとえばロードバランシング、モニタリング、サービス間の認証を提供することがあります。サービス メッシュを提供するテクノロジーの例として、Istio と linkerd があります。

S

sidecar(サイドカー)

同じ場所にある別個のコンテナに配置することによってアプリケーションのコンポーネントまたはサービスを分離し、非依存性と柔軟性を実現するクラウドネイティブのアーキテクチャパターン。

Software as a Service (SaaS)

クラウドインフラストラクチャで実行されるアプリケーション。ダウンロードしてローカルマシンにインストールするのではなく、ネットワーク(通常はインターネット)経由で使用されます。サービスの利用者は、基盤となるクラウドインフラストラクチャまたはアプリケーションの機能の管理や制御を行いません。「Webアプリケーション」とも呼ばれます。

Software-Defined Data Center (SDDC)

インフラストラクチャが仮想化され、サービスとして提供されるデータセンター。SDDCのインフラストラクチャには、仮想化されたネットワーク、ソフトウェアベースのデータストレージと管理などがあります。SDDCでは、従来の環境より柔軟性、俊敏性、効率性、費用対効果に優れた方法でアプリケーションを実行できます。SDDCでは、インフラストラクチャのすべてのコンポーネント(コンピューティング、ネットワーク、ストレージ、セキュリティ、可用性)が抽象化され、自動化されたポリシーベースのソフトウェアとして提供されます。SDDCにより、手動プロセスが大幅に減少し、ITサービスのデリバリーが迅速化するとともに、コストが削減され、ROIが著しく向上します。

Spanner

「Cloud Spanner」を参照。

Spark

Apache Spark は、Python シェルからインタラクティブに使用できる大規模データ処理エンジンです。1 つのアプリケーション内に共存できる一連のツールを使用して、ストリーミング、SQL、複雑な分析を組み合わせます。Spark は Hadoop File System (HDFS) だけでなく、Cassandra や MongoDB などさまざまなデータソースにアクセスできます。Python の高度な数値処理ライブラリにアクセスできることから、多くのデータサイエンティストが Spark を好んで利用しています。

spec (仕様)

Kubernetes では、ユーザーが指定する構成など、望ましい状態を記述したものを指します。

Spring

Spring Framework は、Pivotal のオープンソース Java フレームワークであり、Java アプリケーションを開発するためのインフラストラクチャを処理します。Spring には、Spring Boot、Spring Cloud、Spring Cloud Data Flow が含まれます。

Spring Cloud Data Flow

データ統合パイプラインおよびリアルタイムのデータ処理パイプラインを構築するためのツールキット。Spring Cloud Data Flow サーバは、Spring Cloud Deployer を使用して、Pivotal Cloud Foundry、Mesos、または Kubernetes とパイプラインを統合します。Spring Cloud Data Flow では、収集、リアルタイム分析、バッチ処理、データのエクスポートを統合する分散システムを使用して、分析パイプラインを開発できます。

StatefulSet

Kubernetes で使用する StatefulSet は、ユーザーが指定した望ましい状態に基づいて、一連のポッドの展開とスケールリングを管理します。たとえば、ステートフルポッドのパーシステントストレージやその他のリソースを管理します。

Swarm

Docker で使用される swarm は、swarm モードで動作する 1 つ以上の Docker エンジンのクラスタです。ただし、Docker Swarm は、Docker エンジンの swarm モード機能と同じではありません。「Docker Swarm」を参照。

T

tag(タグ)

Docker でのタグは、リポジトリ内のほかのイメージと区別するためにユーザーが Docker イメージに適用するラベルです。

Tern

VMware のオープン ソース ツールの 1 つであり、Docker コンテナにインストールされたパッケージのソース、バージョン、ライセンスを識別します。結果として得られるマニフェストは、コンテナ内のパッケージがライセンス要件を順守していることを確認するのに役立ちます。<https://github.com/vmware/tern> を参照。

Terraform

HashiCorp のソフトウェアで、Google Cloud Platform などのクラウド サービス プロバイダーの環境でインフラストラクチャをコードとして作成するためのものです。

the cloud(クラウド)

インターネット経由で利用できるコンピューティング リソース。「cloud computing (クラウド コンピューティング)」を参照。

traditional application

(従来型アプリケーション)

一般に従来型アプリケーションは、データベース、アプリケーション、Web サーバで構成される n 層アプリケーション アーキテクチャに基づくモノリシック設計を採用しています。これらのコンポーネントはインフラストラクチャと密に結び付いており、その可用性はインフラストラクチャに依存します。

Twelve-Factor App

サービスとしてのソフトウェア(SaaS)アプリケーション、つまり Web アプリケーションを開発し、サービスとしてのプラットフォーム(PaaS)またはサービスとしてのコンテナ(CaaS)に展開するための手法。

U

UID

文脈またはシステムに応じて、ユーザー識別子、ユーザー ID、一意の識別子のいずれかの略。たとえば Kubernetes での UID は、オブジェクトを一意に識別する文字列です。

underlay network (アンダーレイ ネットワーク)

アンダーレイ ネットワークは、従来型のハードウェアベースのネットワーク接続手法か、ハードウェアとソフトウェアの組み合わせを使用して、仮想マシンまたは物理マシンを接続します。「overlay network(オーバーレイ ネットワーク)」、「service mesh(サービス メッシュ)」、「NSX」も参照。

V

Vagrant

HashiCorp の Vagrant は、マシンの構成をもとにテンプレートを作成し、そのテンプレートを配布して予測可能なアプリケーション開発環境を構築するツールです。

virtual infrastructure manager (仮想インフラストラクチャ マネージャ)

仮想インフラストラクチャ マネージャ (ViM) は、ネットワーク機能の仮想化インフラストラクチャ (NFV-I) に関連付けられたリソースを管理します。「network functions virtualization (ネットワーク機能の仮想化)」を参照。

VMware Integrated OpenStack

VMware がサポートする OpenStack のディストリビューションの 1 つであり、VMware Infrastructure 上で OpenStack を実行できます。「OpenStack」を参照。



volume (ボリューム)

Docker での ボリューム (データボリューム) は、1 つまたは複数のコンテナ内にあり、Union File System をバイパスする特別なディレクトリを指します。ボリュームは、コンテナのライフサイクルにかかわらずデータが存続するように設計されています。

vSphere Cloud Provider

VMware のパーシステント ストレージソリューションであり、データベースを使用するものを含め、大量のデータを扱うステートフルなコンテナ化されたアプリケーションをサポートします。Project Hatchway によって Kubernetes の一部となる vSphere Cloud Provider を使用することで、VMware Pivotal Container Service は vSphere ストレージ上で次のストレージ プリミティブをサポートすることができます。ボリューム、パーシステント ボリューム (pv)、パーシステント ボリューム クレーム (PVC)、ストレージクラス、ステートフル セット。vSphere Cloud Provider は、ポリシーベースのストレージ管理など、エンタープライズストレージ機能も提供します。「Pivotal Container Service」、「Hatchway」を参照。

vSphere Integrated Containers

VMware の製品の1つであり、コンテナ化されたアプリケーションを仮想マシン上で実行できるように、VMware vSphere 内にコンテナ インフラストラクチャを作成します。このソリューションには、コンテナを実行するためのエンジン、コンテナを管理するためのポータル、コンテナ イメージを安全に保存するためのレジストリ、Docker 互換のインターフェイスが含まれます。vSphere Integrated Containers の主なメリットは、コンテナの可搬性と、vSphere のセキュリティ、可視性、管理機能を組み合わせたとこにあります。

W

workload(ワークロード)

アプリケーション関連のタスク(コンピューティング、ネットワーク、ストレージ)を処理する際に発生する演算またはトランザクション処理の負荷。同じテクノロジーとツールを使用している同種のアプリケーションでも、状況や時間帯によってワークロードが大きく異なる可能性があります。通常、ワークロードは、CPU またはメモリの消費量、ネットワークトラフィック、リクエスト数、データベースのクエリ件数、トランザクション数などで測定されます。基本的な用語を大まかに定義すると、「アプリケーション」はなにかを処理するもの、「ワークロード」はそこで実行される処理、「ユースケース」はそれを実行する理由です。クラウドコンピューティングおよび Kubernetes クラスターでは、アプリケーションのインスタンスまたはアプリケーションの一部が一定時間内に実行する作業の量をワークロードと呼んでいます。

X

XML

Extensible Markup Language の略。データの構造定義と交換で使用する、柔軟でありながら詳細な記述が可能なフォーマットです。XML は、構成ファイルの作成やデータ交換などさまざまな目的で、レガシー アプリケーション、Java アプリケーション、Web アプリケーションで使用されます。XML がクラウドネイティブ アプリケーションで使用される場合もありますが、データフォーマットとしては JSON または YAML の方が適しています。「JSON」、「YAML」を参照。

Y

YAML

人間が理解できるデータ シリアル化のための標準フォーマット。通常、YAML を構成ファイルで使用して、情報やコマンドを構造化します。Kubernetes では、仕様ファイルを YAML で記述します。

YARN

Apache Hadoop のサブプロジェクト。演算処理からリソース管理を切り離し、HDFS に保存されるデータに対して、MapReduce よりも相互作用パターンを拡張できます。YARN は、MapReduce や Storm などの Hadoop アプリケーションが処理を実行する時点でリソースを割り当てます。データオペレーティングシステムとプラグイン対応アーキテクチャをほかのアプリケーションに提供する YARN は、Hadoop 環境の中心となるツールです。

Z

ZooKeeper

Apache ZooKeeper は、分散アプリケーションを調整するツールです。アプリケーションの名前用のレジストリを提供し、それらを構成して、同期します。分散アプリケーションの誤動作を防ぎます。

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

Y

Z

