Google Cloud

Ask the Expert ’21
～アプリケーション開発～

アジェンダ

1. 製品アップデート
2. アプリケーションをどこで動かすのか？
3. Anthos

製品アップデート

01

Google Cloud

最新機能を適用した Secured Cloud Run 2021

コンテナアプリを簡単にデプロイできる Cloud Run は、シンプル、かつスケーラブルな Serveless 基盤です。URL も生成され、利用勝手が良いですが、用途によって認証、IP 制限、VPC 内で動くような挙動を実現したい、などあるかと思います。本セッションでは、そういったネットワークの制限を入れたセキュアな Cloud Run の構成について紹介したいと思います。

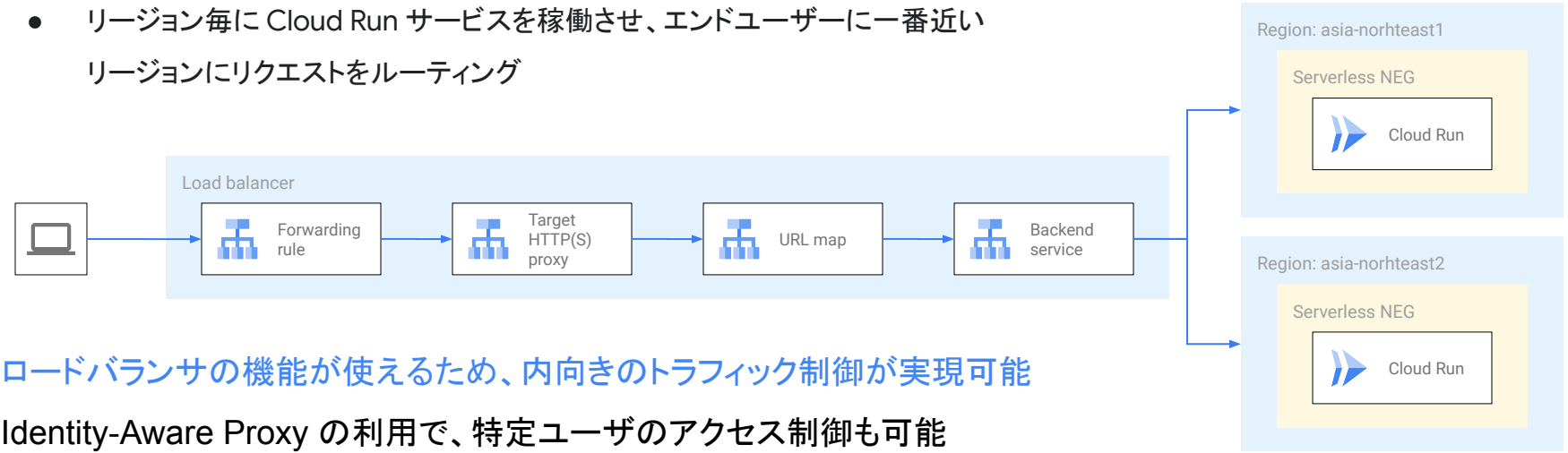
取り上げる主な Google Cloud 製品 / サービス

- Cloud Armor
- Cloud Load Balancing
- Cloud Run

Google Cloud 頼兼 孝幸
アプリケーション モダナイゼーション スペシャリスト

Serverless Network Endpoint Group（Serverless NEG）を利用し、ロードバランサ（GCLB）と連携

- Cloud Run などの Serverless サービスの、バックエンド エンドポイント グループ
- GCLB のバックエンド サービスとして、複数の Serverless NEG が設定可能
- リージョン毎に Cloud Run サービスを稼働させ、エンドユーザーに一番近いリージョンにリクエストをルーティング



ロードバランサの機能が使えるため、内向きのトラフィック制御が実現可能

Identity-Aware Proxy の利用で、特定ユーザのアクセス制御も可能

What's new in Google Kubernetes Engine.

本セッションでは Google Kubernetes Engine (GKE) の最新情報について、20 分間で出来る限り深く解説していきます。Live QA でのご質問もお待ちしています！

取り上げる主な Google Cloud 製品 / サービス

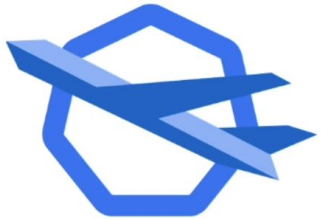
- Kubernetes Engine (GKE)

Google Cloud 篠原 一徳
アプリケーション モダナイゼーション スペシャリスト

GA

GKE Autopilot

GKE Autopilot は **GKE の新しいモード**。
従来の GKE は GKE Standard と呼ぶ。
基本的な仕組みは 従来の GKE と同じだが、以下の特徴がある。



- **Node も完全マネージド（Control Plane に加えて）**
- **Pod 単位の課金**
- **Pod 単位の SLA**

特徴 1 : Node も完全マネージド

つまり、「**Node の運用をしなくてよい**」ということ。

具体的には

- Node の作成、更新、削除は全て自動で行われる。
- バージョン アップグレードも自動で行われる。

ユーザーは **Kubernetes の「利用」に集中**できる。

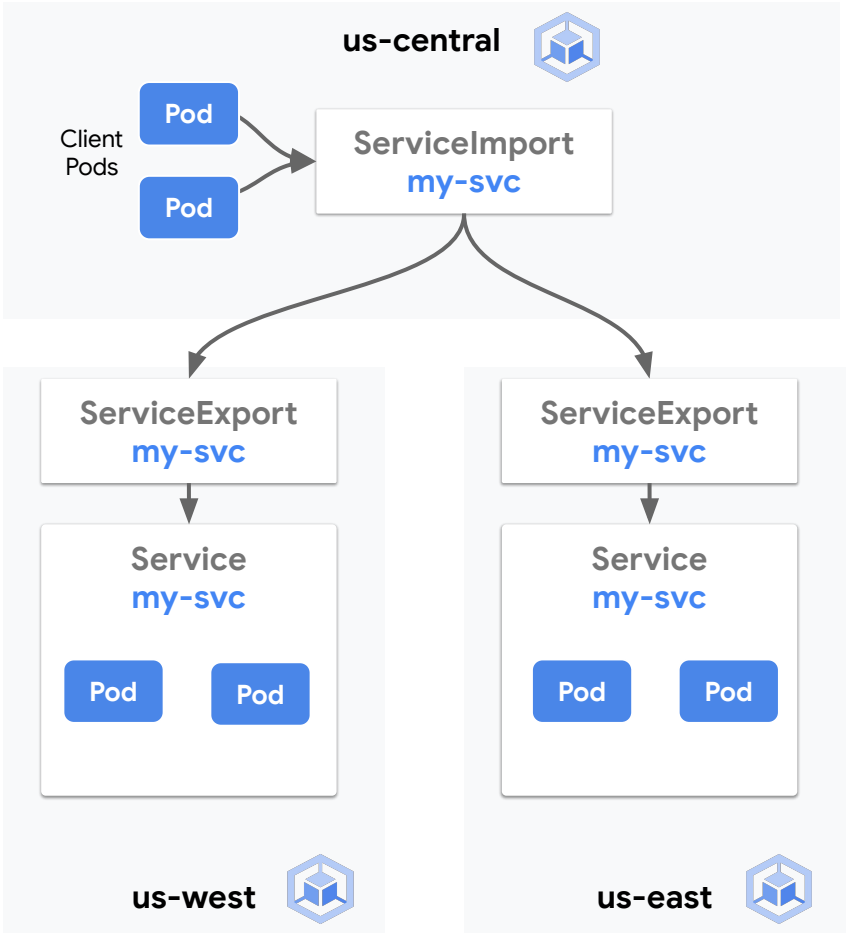
GA

Multi-cluster Services

クラスタ間でサービスディスカバリを提供

- Environ*1 に登録されたクラスタ間で有効
- 同一 VPC 内、Peering した VPC 内で利用可能
- Cloud DNS を使って Export した Service のレコードを持つ
- OSS の API を踏襲

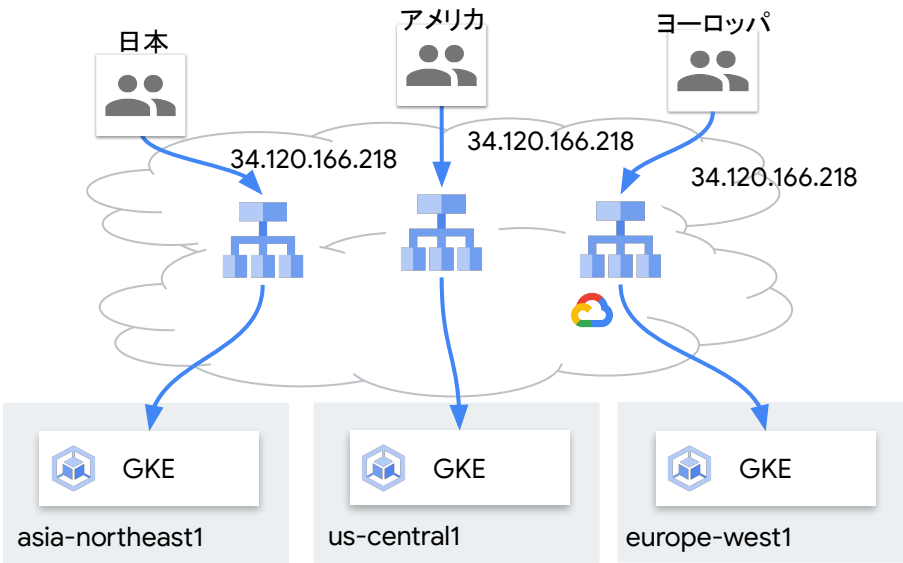
*1 : Environ とは GKE のクラスタを論理的にグループ化する機能です



GA

Multi-cluster Ingress

- グローバルに単一の VIP を提供し、最寄りの GKE へトラフィックを転送
- リージョンを跨いだクラスタ間での HTTP/HTTPS ロード バランシングが可能
- Environ に登録されたクラスタ同士で構成することが可能。
- Ingress for Anthos から rename



アプリケーションを
どこで動かすのか？

02

Google Cloud

アプリケーションはどこで動かすべきか、それが問題だ - Google Compute Engine から Kubernetes Engine, Cloud Run まで -

サービスを開発するときに必ずぶつかる壁、それがアプリケーションをどこで動かすか、です。様々な選択肢があり、迷われている方も多いのでは無いでしょうか。本セッションでは、どのような観点、要件を元にどの選択肢を選ぶかを、仮想マシンの Compute Engine から、マネージドな Kubernetes の GKE、そしてコンテナ＋ Serverless の Cloud Run まで最新の情報を交えながらご紹介します。

取り上げる主な Google Cloud 製品 / サービス

- Cloud Run
- Compute Engine
- Kubernetes Engine (GKE)

Google Cloud 長谷部 光治
ソリューションズ アーキテクト

サーバーレス

- 成熟度に関する**懸念もなくなりつつある**

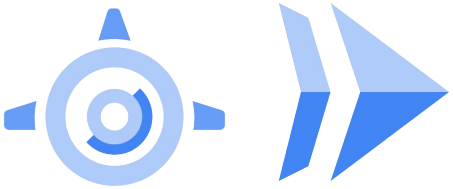
懸念点	Cloud Functions	Cloud Run
ドキュメント、サンプル、ツール、ベスト プラクティス情報が足りない	ベータリリースから 4 年が経過しアセットが揃ってきた	アルファ リリースから 3 年弱が経過、各種アセットが拡充中
IaaS、PaaS と比べるとデバッグが困難	• Cloud Debugger, Cloud Logging を利用し効率的にデバッグ • Functions Framework を使ったローカルでの開発	• Cloud Debugger, Cloud Logging を利用し効率的にデバッグ • Docker などを使ったローカルでの開発
アイドル状態が続くと次の実行が“コールド スタート”となる	コールドスタートをできる限り避けるベスト プラクティスを提供	最小インスタンス、同時実行を最適化しコールド スタートを避ける

サーバーレス

- **オープン性**を重視

懸念点	Cloud Functions	Cloud Run
複雑なケースではロジックの量に比べ、より多くの操作が必要になる	関数は複雑なロジックの構築に向いていない	アプリケーション単位で実行できるため、ロジックをまとめて対応
標準化、エコシステムが成熟していない	オープンな Functions Framework を利用し、エコシステムを拡充中	オープンソース の Knative コミュニティと共にエコシステムを拡充
提供プラットフォームにロックインする可能性	オープンなフレームワークを使うことで、ロックインを回避	

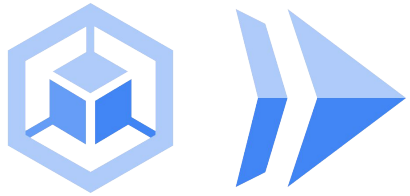
Platform as a Service（PaaS）



- PaaS のプロダクトも Cloud Run で拡充

懸念点	App Engine	Cloud Run
ランタイム バージョンに翻弄される	同じ様に提供ランタイムに影響を 受ける可能性がある	ランタイム バージョンは気にする 必要がない
“12 ファクター”のプラクティスに縛られて、アーキテクチャの柔軟性が失われる可能性	“12 ファクター” にあるような stateless なアプリケーションを 作るのに向いている	
提供プラットフォームに ロックインする可能性	ロックインになってしまう 可能性がある	コンテナ、そしてオープンな Knative をベースとすることで ロックインを回避

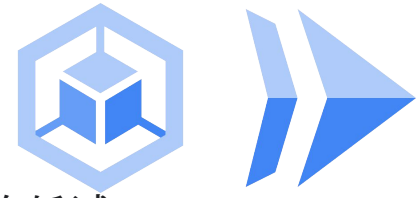
Container as a Service（CaaS）



- **マネージド**のメリット、**各種自動化機能**を活かし、利用者の負荷を低減

懸念点	GKE		Cloud Run
	Standard	Autopilot	
セキュリティ パッチ対応などの実行環境管理	コントロール プレーンは Google、ノードについては利用者と Google の共同責任	コントロール プレーン、ノード共にGoogle の責任	Google が責任を持ち、利用者は気にしなくて良い
ロード バランシングとスケーリング	利用者が設定を行い Google がその設定どおりに稼働させ続ける責任を持つ		自動で設定、スケーリングが行われる
キャパシティ管理	利用者がノードのキャパシティ、自動スケーリング設定を行う	要求されたリソースが確保される	キャパシティを意識する必要はない

Container as a Service（CaaS）



- Google Cloud 提供の**各種マネージド機能**を利用することにより作業負荷を低減

懸念点	GKE		Cloud Run
	Standard	Autopilot	
スタートアップに時間がかかる	イメージの事前展開などスタートアップの時間を短縮する各種プラクティス		各種コールド スタートを低減する設定、機能
アプリケーション構成に関して“ガードレール”の役割を果たすものが少ない	様々なプラクティス、ポリシー管理、セキュリティ管理機能を提供	ベスト プラクティス、セキュリティ設定が初めから組み込み済み	構成が抽象化されておりセキュリティを担保
ビルド、デプロイのメカニズムを用意	Cloud Build など Google Cloud が提供するサービスと簡単に連携		
モニタリング、ロギング、その他の共通サービスとの連携を管理	運用管理に利用する Cloud Operations、その他プライベート コンテナ レジストリ、プライベート Git リポジトリといったサービスと簡単に連携		

利用料金

- **コストの考え方**と**ワークロードの特性**を併せて評価する

	Compute Engine	GKE		Cloud Run	App Engine	Cloud Functions
		スタンダード	Autopilot			
料金を構成する要素	● 仮想マシン (vCPU, メモリ) ● ディスク ● イメージ ● ネットワーク転送量 ● GPU	● Compute Engine 利用料 ● クラスタ利用料 (1 クラスタごと)	● Pod (vCPU, メモリ, ディスク) ● クラスタ利用料 (1 クラスタごと)	● CPU ● メモリ ● リクエスト量 ● ネットワーク転送量	● インスタンス (CPU, メモリ) ● ネットワーク転送量	● 呼び出し回数 ● コンピューティング時間(CPU, メモリ) ● ネットワーク転送量
最小課金単位	最低 1 分、以降は 1 秒単位	最低 1 分、以降は 1 秒単位	1 秒	100 ミリ秒	15 分	100 ミリ秒
ゼロスケール可能か？	いいえ	いいえ	はい (クラスタ利用料は除く)	はい	はい	はい

- 実際に掛かるコストは、**実ワークロードを使って試算** すること

機能と技術要件

	Compute Engine	GKE		Cloud Run	App Engine	Cloud Functions
		スタンダード	Autopilot			
通信プロトコル	制限なし	制限なし	制限なし	HTTP/1.x, HTTP/2(gRPC 含む) over TLS, HTTPS, WebSockets	HTTP/1.x, HTTPS	HTTP/1.x, HTTPS
Google Cloud ソースのイベント連携	-	-	-	Eventarc を利用したイベント連携	-	ネイティブで様々なイベントと連携

新規事業「Bill One」による Google Cloud 活用術

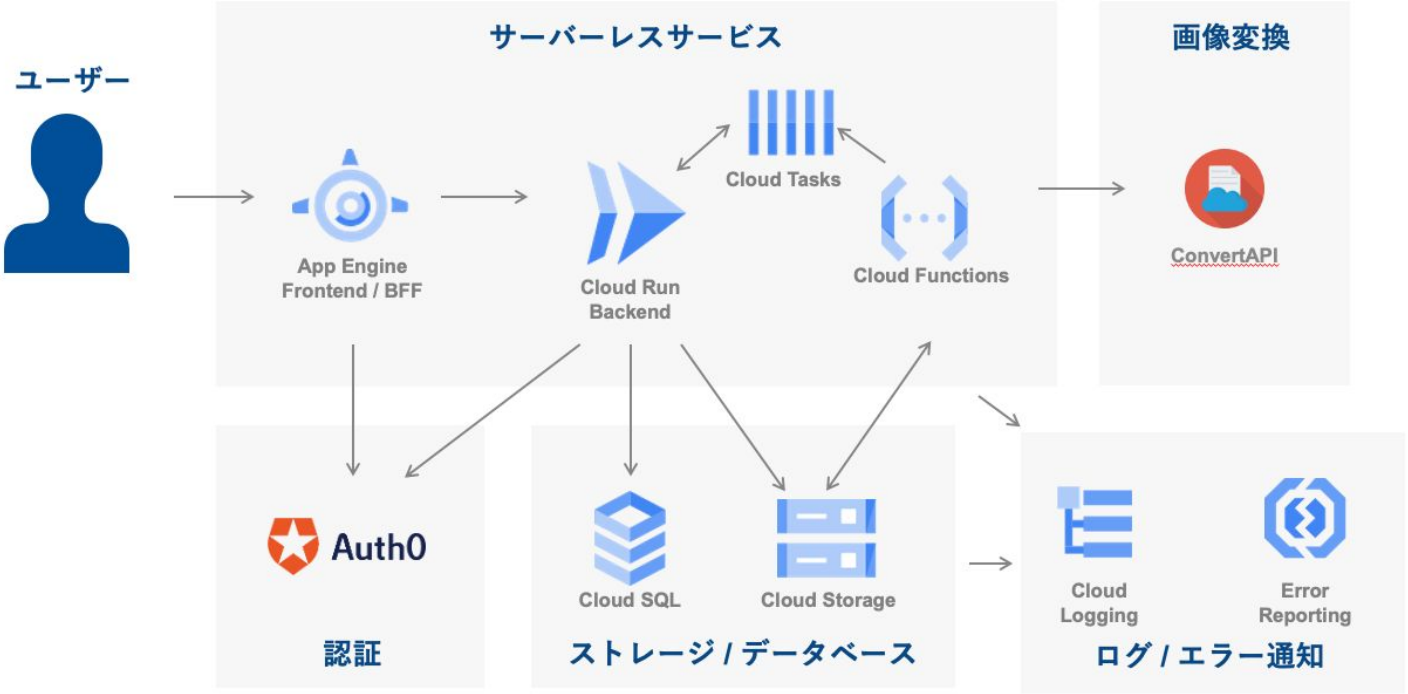
2020 年 5 月にローンチしたクラウド請求書受領サービス「Bill One」。Bill One 事業を爆速で推進するために Google Cloud 活用術を紹介します。

取り上げる主な Google Cloud 製品 / サービス

- Cloud Build
- Cloud functions
- Cloud Run

Sansan株式会社 大西 真央
Bill One事業部 プロダクト開発責任者

アーキテクチャ


ユーザー
サーバーレスサービス
App Engine
Frontend / BFF
Cloud Run
Backend
Cloud Tasks
Cloud Functions
画像変換
ConvertAPI
Auth0
認証
Cloud SQL
Cloud Storage
ストレージ / データベース
Cloud
Logging
Error
Reporting
ログ / エラー通知

損害保険ジャパンの次期アプリケーション実行環境の選定～PaaS 環境の利用標準化について～

損保ジャパンではパブリッククラウドの積極的な活用を進めてきましたが、IaaS 中心のためアプリ担当者とインフラ担当者間で非機能要件やシステム構成調整のワークロードが大きな課題となっておりました。そこで、PaaS を最優先に利用する規定を設けることにより、PaaS のメリットを最大限生かしインフラ環境構築を省きすぐにアプリケーション開発ができる環境を目標とし PaaS 環境の利用標準化策定に取り組みました。PaaS 選定にあたり、GCP、AWS、Azure、Heroku の PaaS 機能を比較評価した結果、Google を選択しました。「今後クラウドネイティブな Web アプリケーション開発、コンテナ開発推進にあたっては、GCP が提供する PaaS を採用することがなぜ唯一の選択であったのか」について解説します。

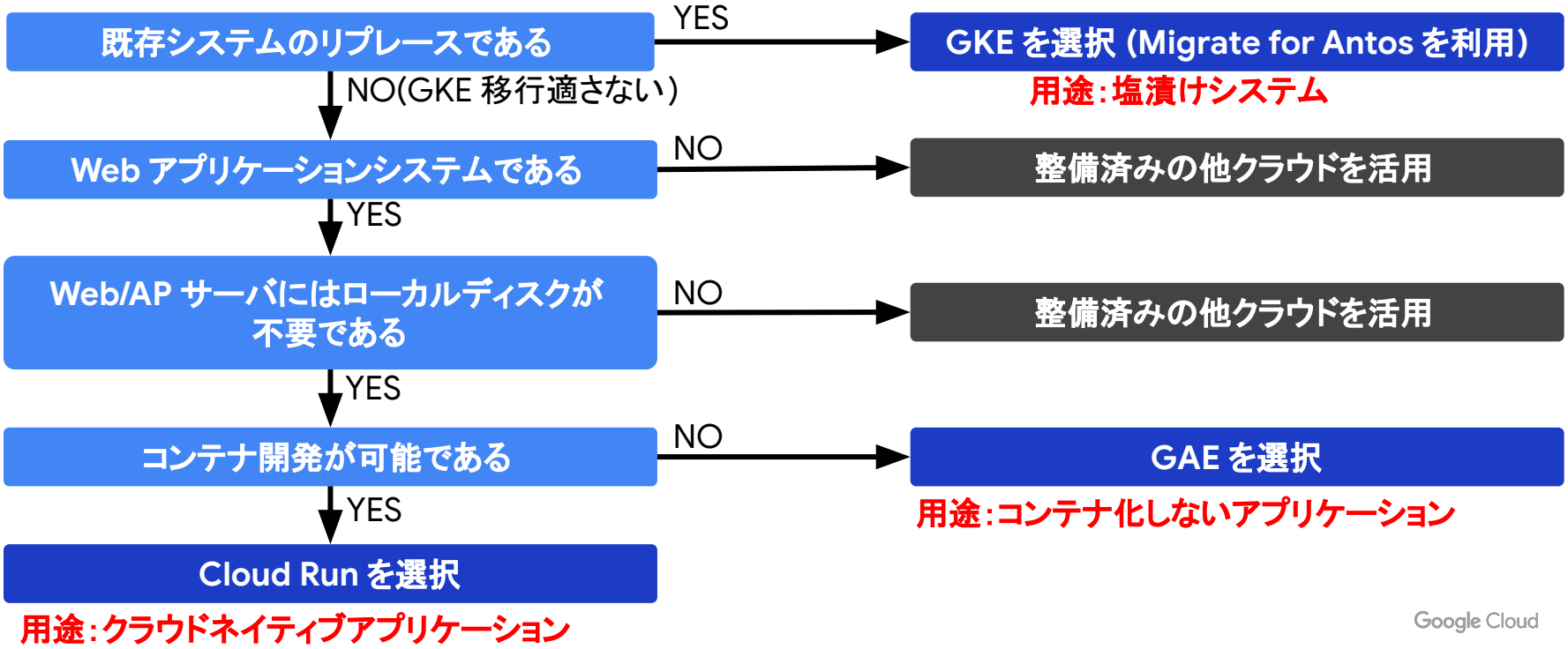
取り上げる主な Google Cloud 製品 / サービス

- App Engine
- Cloud Run
- Kubernetes Engine (GKE)

SOMPOシステムズ株式会社
野中 誠貴
アーキテクト部 チーフリーダー

Google Cloud ディシジョンフロー

用途に応じて最適なPaaS を選定できるようになることが重要があり、これからクラウドネイティブな開発に取り組むアプリチームに向けてディシジョンフローを用意した。



Cloud Run 利用方針

Cloud Run のプラットフォームは以下の3 種類ある。

1. Cloud Run (フルマネージド)
2. GKE 上で動かす Cloud Run for Anthos on Google Cloud
3. Anthos clusters on VMware 上で動かす Cloud Run for Anthos on-prem

損保ジャパンにおいては、完全にGoogle がインフラを管理してくれるCloud Run (フルマネージド) を採用する。

Anthos は、GKE の管理をユーザーがする必要があり、2021 年 1 月時点で Kubernetes の利用が非常に少ないためコンテナ開発の成熟度に合わせて今後検討する方針とした。

Cloud Run (フルマネージド)の見解

損害保険ジャパンにおいては、Cloud Run (フルマネージド) を採用し、Kubernetes の学習コストを排除し、コンテナ開発に"**慣れる**"ことを重要視した。

- コンテナ化されたアプリケーションをシングルコマンドでデプロイできるためサービス固有の構成は不要
  ⇒ 現時点では Kubernetes が必要になるような複雑な要件（マイクロサービス）が無いと判断し、よりシンプルにコンテナを扱えるCloud Run のメリットを重視

- リクエスト数に応じ自動的にスケーリング
  ⇒ Kubernetes を使用したクラスターの構成・管理が不要となる。

- Kubernetes の名前空間やポッドでのコンテナ共存・ノードの割り当てや管理が不要になる。

GAE 利用方針

原則、従来のコンテナ化しないアプリケーションを維持せざるを得ないときの実行環境として利用する方針とする。
GAE は Flexible 環境と Standard 環境の 2 タイプ存在するが、Flexible 環境は、Cloud Run と機能重複すること、また VM の高速起動のメリットを重視し、スタンダード環境を採用する方針としました。

GKE 利用方針

コンテナ化されたアプリケーションであれば大きな制約はなく、GKE の仕様 (GKE
サービスとしてのクラスタ数、ノード数、コンテナ数の上限 の範囲で自由度が高く実装が可能である。
半面、Kubernetes 自体が複雑で学習コストが高いため、2021 年 1 月時点では利用範囲を絞って活用する方針とした。

損保ジャパン環境において以下に合致するシナリオで適用可能としている。

1. オンプレミス (vSphere) 環境上で稼働しているシステムにおいてリプレースが必要である。
2. 現行 OS・ミドルウェアバージョンを維持して、バージョンアップに伴うアプリケーション改修コストを抑止したい。

Anthos

03

Google Cloud

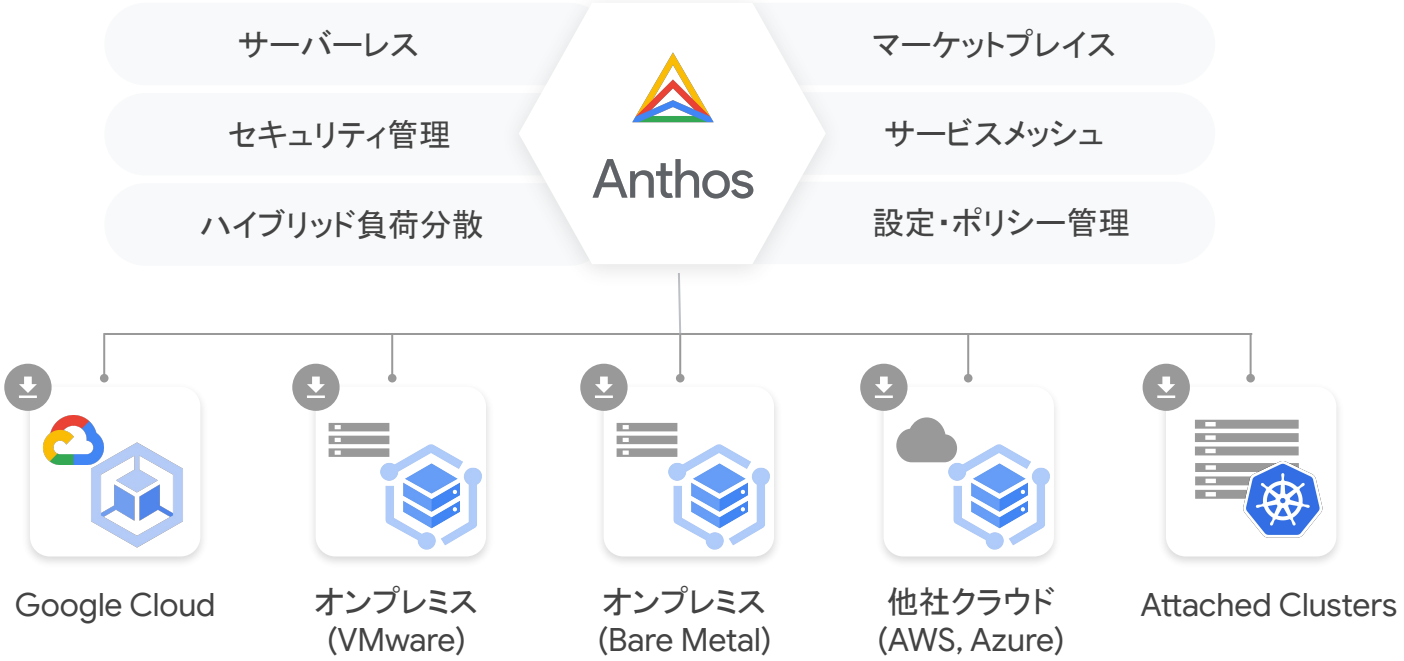
Anthos とは？

アプリケーションの モダナイゼーションのためのプラットフォーム

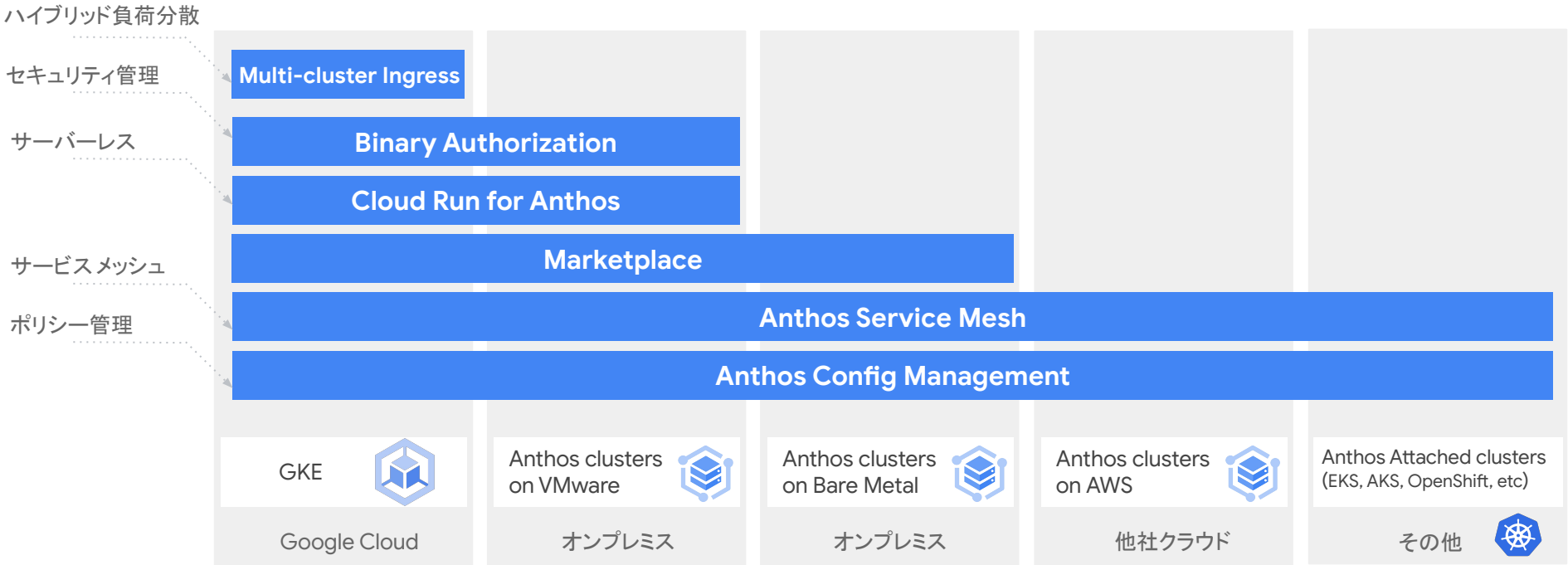
Anthos を導入することで **マネージドな Kubernetes** を
Google Cloud に加え、**オンプレミス、エッジ、他社クラウド** でも利用出来る他、
サービスメッシュ、サーバーレス、GitOps、コンテナセキュリティ など
モダナイゼーションに有用な機能が提供される

https://cloud.google.com/anthos/

ハイブリッド / マルチクラウド 環境に配置した Anthos クラスタに
エンタープライズで必要となる 機能群を提供



Anthos 1.7 サービス一覧

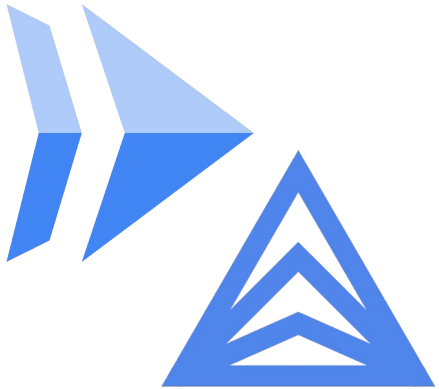


Anthos のデプロイ オプション - https://cloud.google.com/anthos/deployment-options

Cloud Run for Anthos

コンテナをサーバーレスに利用するためのサービス

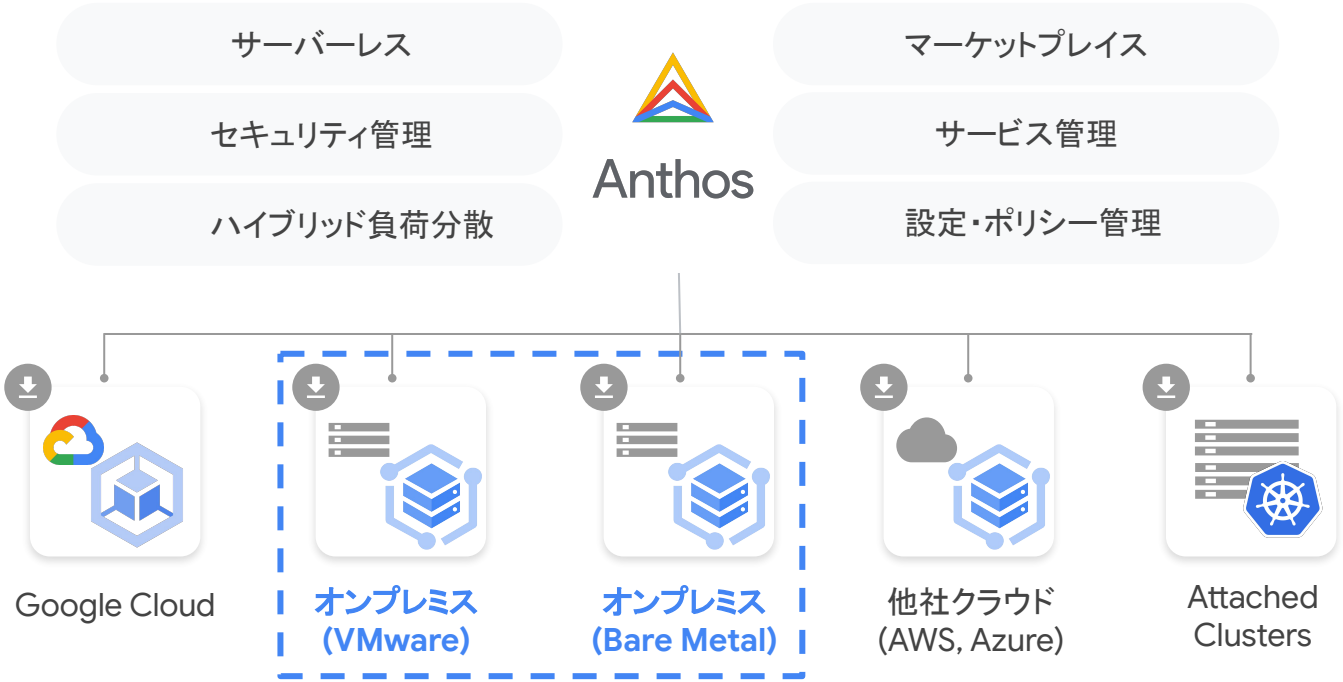
GKE / Anthos clusters 上で Add-on として提供



主な特徴

- 高速に 0 to N スケール
- 言語やライブラリの制約なし
- OSS の Knative 互換、ロックインの排除
- 外部公開やロギングの設定まで一括で自動設定
- for Anthos 以外 にフルマネージド版もあり

オンプレミス環境でのデプロイメント オプション



Anthos clusters on VMware

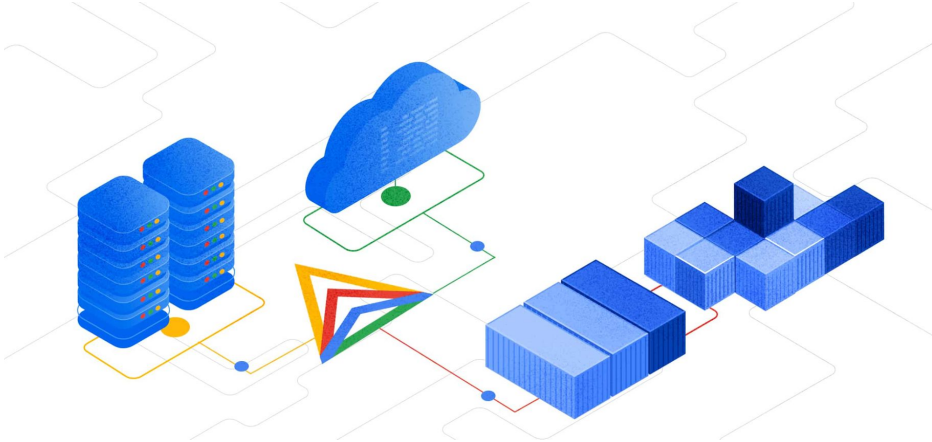
VMware vSphere® 環境上で稼働する
マネージドの Kubernetes

GKE と同様に Google が検証した
最新の Kubernetes を利用可能

Google Cloud のコンテナ エコシステムとの連携 :

Cloud Build, Container Registry, Audit Logs etc.

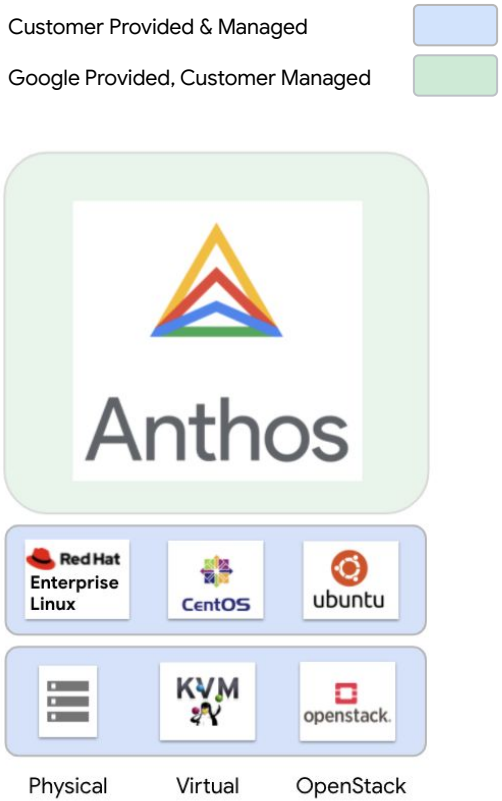
https://cloud.google.com/gke-on-prem/docs/



Anthos clusters on Bare Metal

Anthos コンポーネントをお客様提供のハードウェア / OS の上で稼働

インストール	• OS はお客様による管理 • 最小システム構成を満たすかは事前チェック機能で確認可能 • コンポーネントはすべて冪等かつ宣言的にインストール
動作環境	• 物理でも仮想環境でもデプロイ可能 • 特定バージョンのRHEL, CentOS, Ubuntu をサポート
ネットワーク	• Google 提供のオーバーレイネットワーク& L4 / L7 負荷分散
ストレージ	• CSI を通してのインテグレーション
アップグレード	• インプレースのアップグレードに対応
可観測性	• Operations によるメトリクスとログの収集 • お客様が選択するツールへのインテグレーションも可能



ユースケース: ハイブリッド クラウド構成

Google Cloud とオンプレミスをワークロードの特性に応じて使い分け

Google Cloud

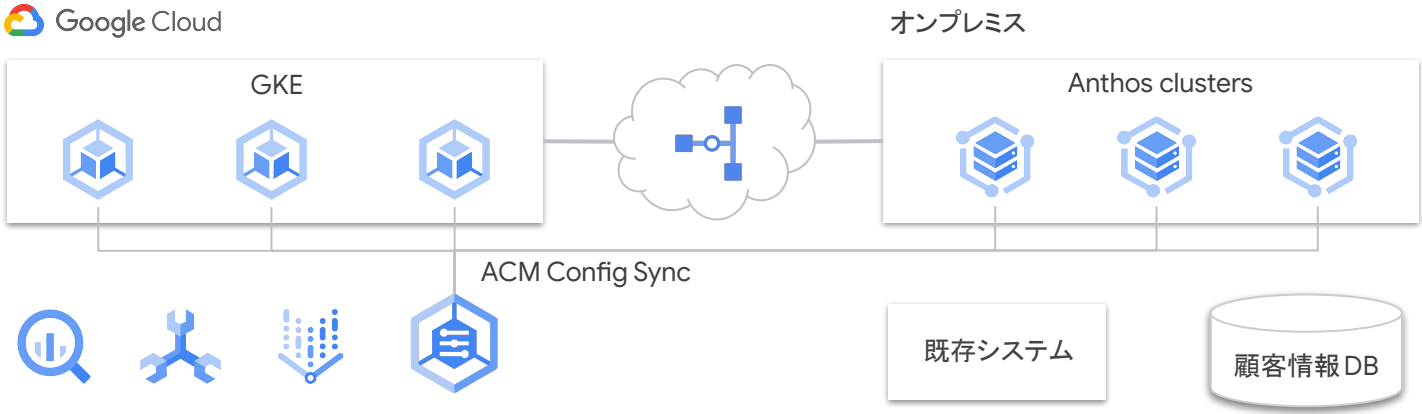
スケーリングを柔軟かつ迅速に行いたい

AI / ML や BigQuery など Google Cloud 固有のサービスと連携するアプリケーション

オンプレミス

Confidential な情報を扱うためクラウドに移行できない

既存システムと低レイテンシーでの接続が必要なアプリケーション

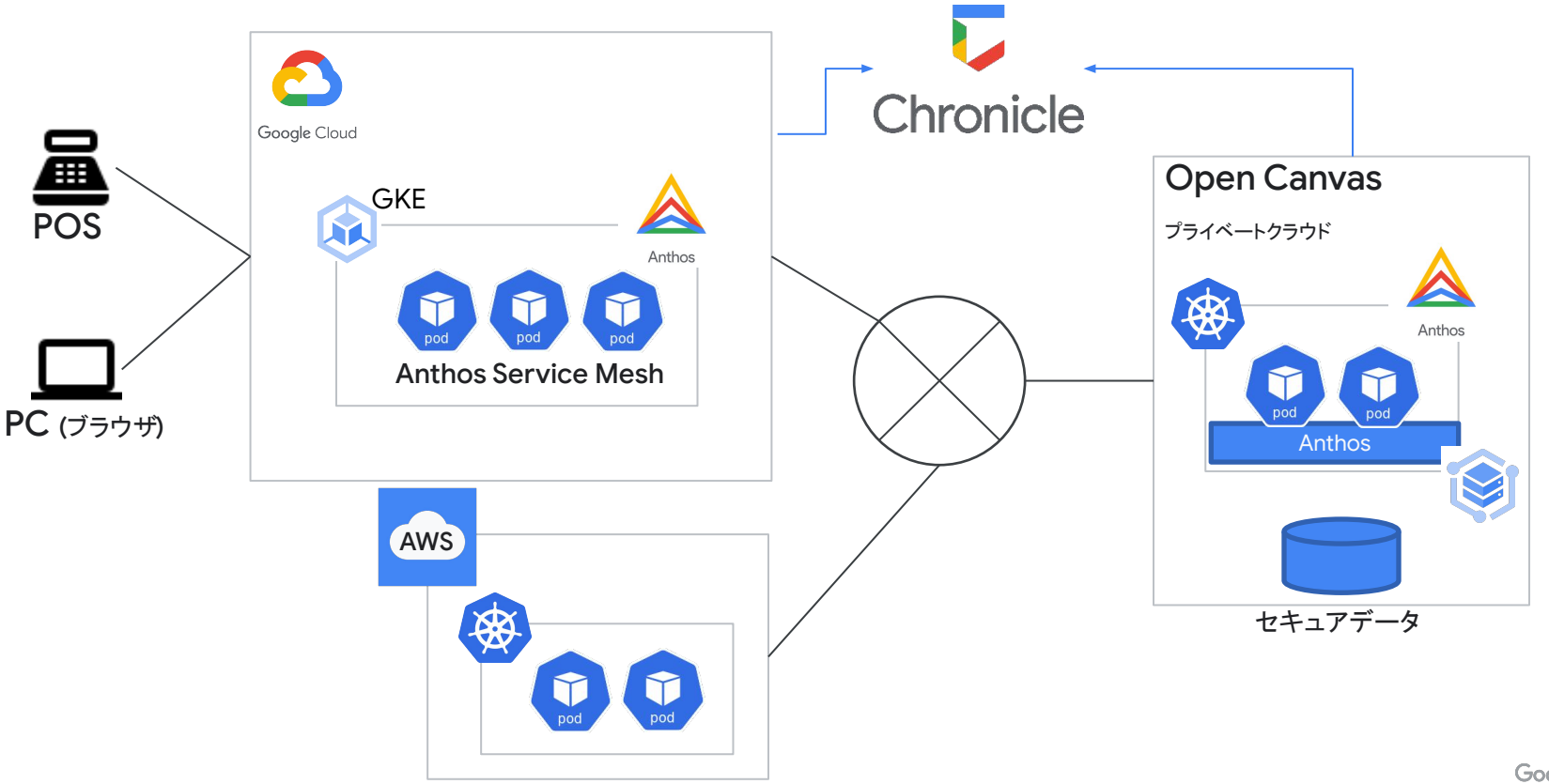


株式会社エヌ・ティ・ティ・データ 様

セッションのダイジェスト

- Digital CAFIS は CAFIS サービス(クレジットカードを中心とした決済基盤) を安定に提供しつつ、事業環境の変化に追随し新たな価値を提供するためのプログラム
- Digital CAFIS を支える ”Omini Platform” は基本機能 / 分析データ / ダッシュボードを統合された API で提供
- 3 つの観点から自社のプライベートクラウドと Google Cloud を組み合わせた “ハイブリットクラウドを採用”
    - クラウド障害時の業務継続性 / セキュアデータのコントロール / クラウドの拡張性を有効活用
- プライベートクラウドを含むマルチクラウドでシステム管理が可能な Anthos を導入

3.4 Omni Platform のアーキテクチャ



4.4 Anthos 検証結果（可用性・拡張性・運用性）

ケース No	観点	結果
2-1	クラウド障害時の別クラウドへの切替	• クラウド / オンプレとの両系アクティブ構成も可能 • オンプレへの AP デプロイを行うことで、障害切替可能 (注) DB 同期や AP の継続性の検討は必要
2-2	オンプレ環境でのオートスケール、自動復旧	• Kubernetes の機能で従前どおり可能
3-1	システム監視およびアラート機能	• Cloud Monitoring を利用することで、監視およびアラートが可能
3-2	障害時の解析情報取得	• Anthos Service Mesh, Cloud Logging の利用で可能

ソフトバンク株式会社様の事例

セッションのダイジェスト

- 対象設備数 100 万ホストを超えるソフトバンクの通信事業設備を監視運用するためのシステム
- 現状は OpenStack を利用して自社データセンター内にプライベートクラウドを独自ネットワーク内に構築
- Kubernetes / Cloud 導入の狙いはリリース工数の短縮・運用削減 / スピード・柔軟性の向上
- Anthos を含む他社製品でオンプレミス製品選定を実施
- 殆どの項目で 3 製品クリアしたが、運用性と構築時の製品の安定性で Anthos が勝利

PoC 評価

オンプレミス向け Kubernetes 3 製品で
正常性 / 機能 / 運用 / 障害試験を実施

ほとんどの評価基準は 3 製品ともにクリア

運用性と製品の安定性で大きく分ける結果となった (※2)

※1 Anthos の NG 項目は、PoC 実施時 vSphere 7 (vSAN 7) 未対応のため PV への RWX / ROX が不可 (現在 vSphere / vSAN 7.0 Update1 対応済みのため解決済)
※2 PoC 時点での評価であり、各社機能追加、改良が盛んで製品評価は実施時期で変動すると思われます

評価項目	Anthos GKE On-Prem	B 社	C 社
IaaS	11/11	11/11	11/11
コンテナ基盤	25/26 (※1)	26/26	24/26 (※2)
ネットワーク	7/7	7/7	7/7
ストレージ	15/15	15/15	14/15 (※2)
ロードバランサ	19/19	19/19	19/19
可観測性	8/8	8/8	8/8
運用性評価	◎	× (※2)	○
安定性	◎	○	× (※2)

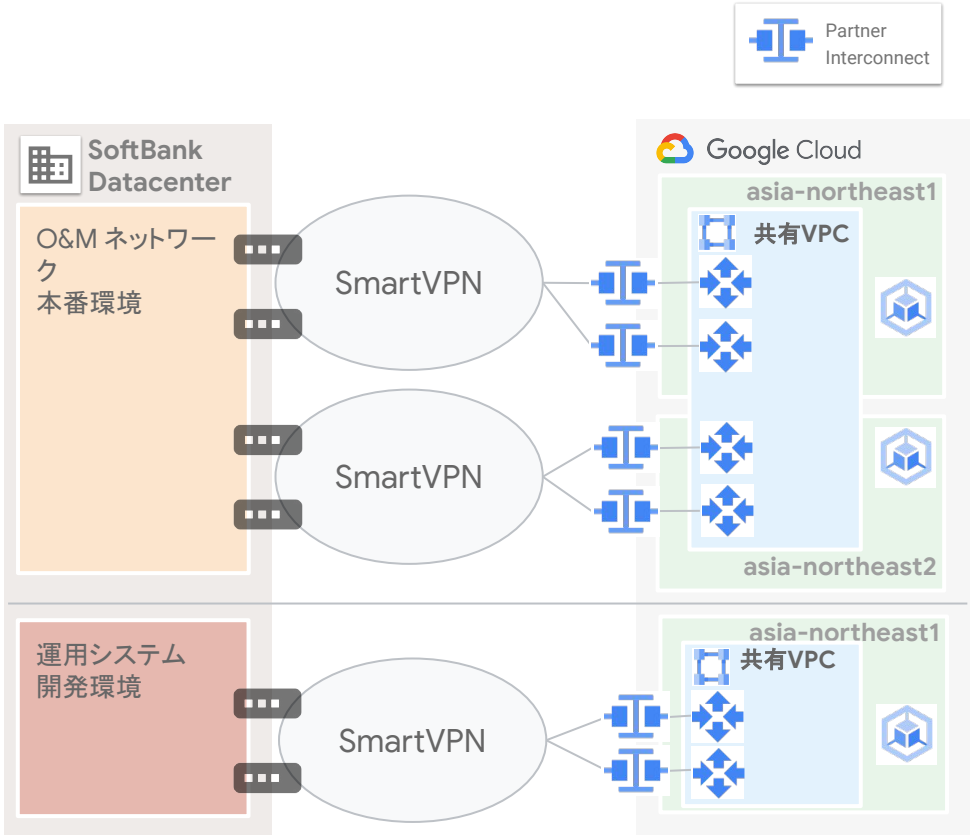
Google Cloud とソフトバンク運用保守 (O&M) ネットワーク接続

Partner Interconnect

- サービス プロバイダを介して、オンプレミス ネットワークと Virtual Private Cloud (VPC) ネットワークの接続を提供する
  - プロバイダを介さず直接物理的に接続するDedicated Interconnect もある
- 本構成では、自社サービスの SmartVPN を通じて「ダイレクトアクセス for GCI-Partner」というサービスで Partner Interconnect を使用している

共有 VPC

- 複数のプロジェクトをまとめるため、共有 VPC を作成して接続している

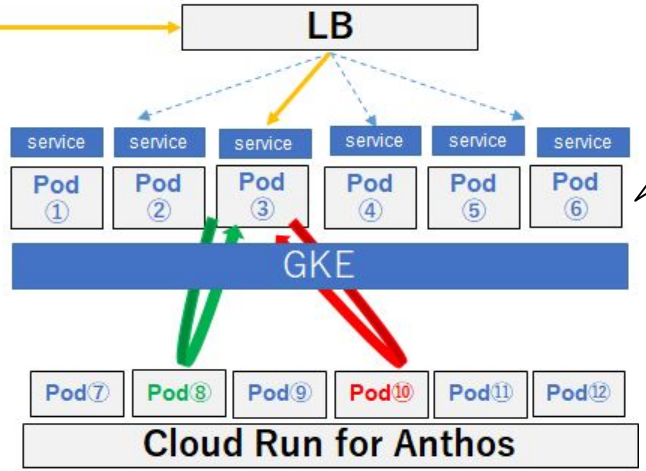


株式会社NTTドコモ様の事例

セッションのダイジェスト

- NTT ドコモにおけるサービス開発の高速化を目指し、バックエンドの基盤システムが個別に提供している API を集約する中間基盤「RAFTEL」を実現すべく Apigee を採用
- 「マイマガジン」では、ニュースメディアとしての基本的価値速報性の向上とパーソナライズ強化を実現すべく Google Cloud を全面的に採用
- マイマガジンのフロントシステムのバックエンドコンテナに Cloud Run for Anthos を採用

フロントのアーキテクチャ | マイクロサービス化のポリシー



フロントエンド

端末アプリとやり取りする外部公開 API
→ ビジネスファンクション単位で分割

バックエンド

コンテナ間通信のみとなる内部向け API
→ コンテンツ返却情報単位で分割

機能間の疎結合 、開発箇所の局所化を実現

Google Cloud 導入を進める中で良かったこと

強力なマネージドサービスを利用することで、複雑な要件を満たしつつ効率的に開発を進めることができた

- **BigQuery**
  - CPU リソースの制約を受けることなく柔軟なデータ分析が可能
- **Cloud Run for Anthos**
  - トラフィックを制御を行いカナリヤリリースが可能
- **Cloud Spanner**
  - 無停止リリースと性能拡張が可能で、リリース頻度を上げることに貢献
- **Cloud Build**
  - サーバレスで CI/CD パイプラインを構築し、リリース高速化に貢献

今年度夏運用開始予定