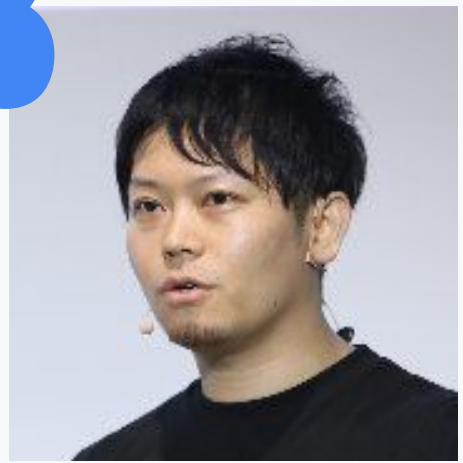
Cloud OnBoard へようこそ

GCP を使ったアプリケーション開発

Google Cloud

ようこそ



篠原 一徳
グーグル・クラウド・ジャパン合同会社
アプリケーション モダナイゼーション スペシャリスト

アジェンダ

時間	トピック
13:00 ～ 13:05	ご挨拶
13:10 ～ 14:00	モジュール 1: アプリケーション開発の概要
14:10 ～ 15:00	モジュール 2: ストレージとモニタリング
15:10 ～ 15:25	Qwiklabs の概要
15:35 ～ 16:15	モジュール 3: サーバーレス コンピューティング／クロージング

Google Cloud

本日の講演データについて

本日の講演データは、
下記 URL もしくは QR コードより
PDF をダウンロードいただけます
https://goo.gle/0618onboardtext

Google Cloud

Google Cloud OnBoard

Google Cloud

モジュール 1

アプリケーション開発の概要

Google Cloud

GCP を使ったアプリケーション開発

アプリケーション開発のベストプラクティス

Google Cloud

クラウドで実行されるアプリケーション

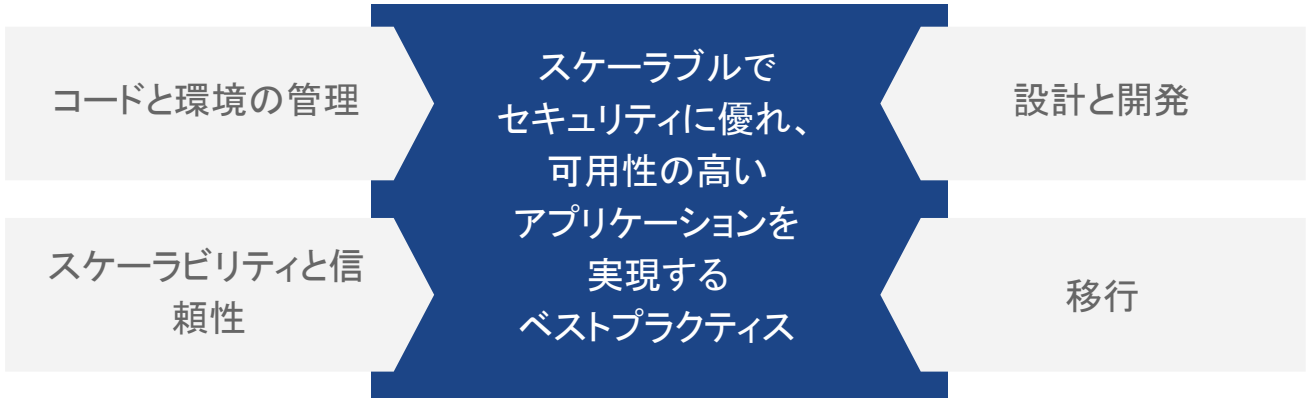
グローバルリーチ

スケーラビリティと高可用性

セキュリティ

Google Cloud

スケーラブルでセキュリティに優れ、可用性の高いアプリケーションを構築するためのベストプラクティスの実装



Google Cloud

アプリケーションのコードと環境の管理

コードリポジトリ

依存性管理

構成設定

Google Cloud

マイクロサービスの実装

モノリシック アプリケーション

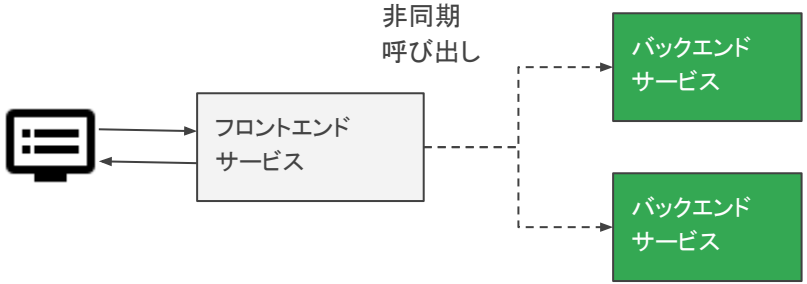
- コード ベースのサイズが大きくなる
- パッケージ内の依存関係が煩雑になる

マイクロサービス

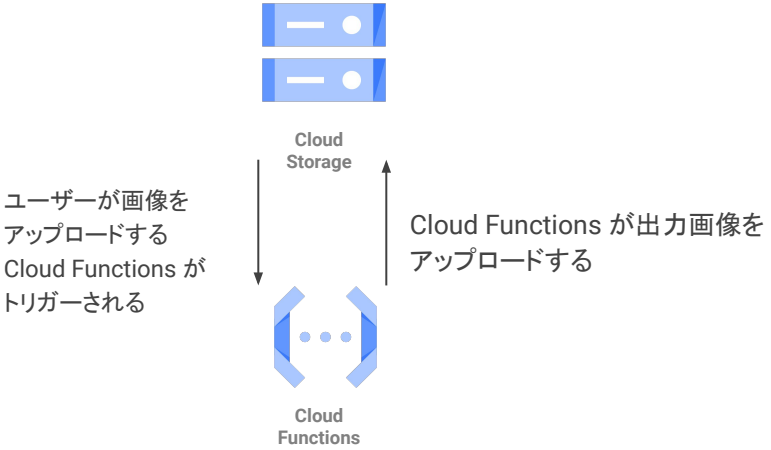
- サービス境界とビジネス境界が一致する
- モジュール式のコード ベース
- サービスごとに独立した更新、デプロイ、スケールが可能

非同期処理

UI の応答性を維持し、バックエンド
オペレーションを非同期で実行



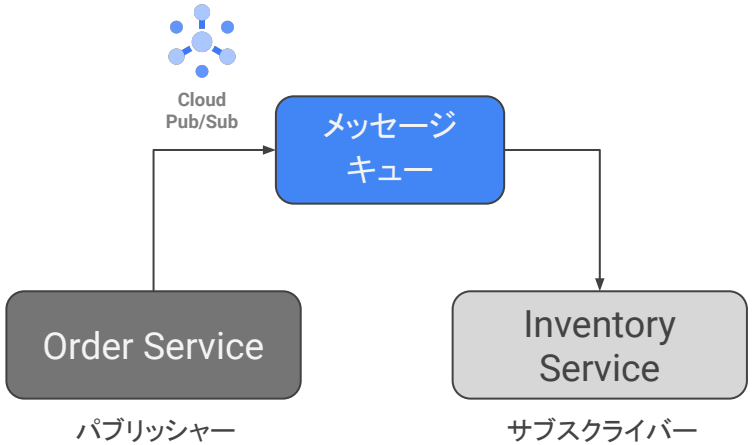
イベント ドリブンな処理を利用



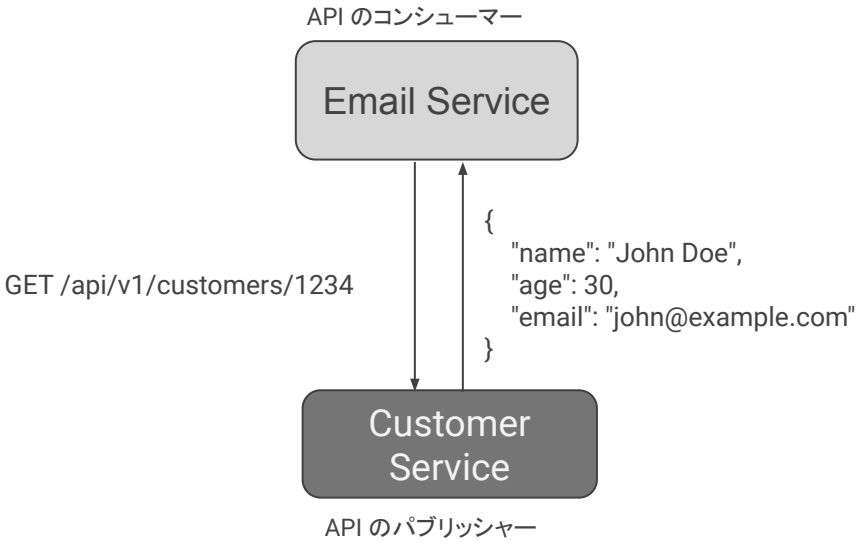
Google Cloud

疎結合な設計

パブリッシャーとサブスクライバーは疎結合



HTTP API の利用側はパブリッシャーの
ペイロードの依存性を持たせない実装に



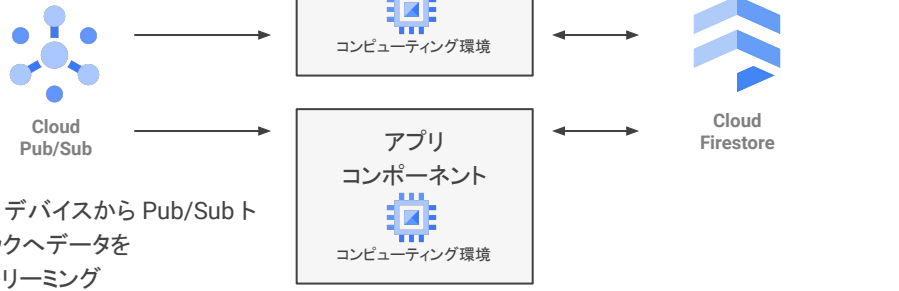
Google Cloud

スケールが必要なタスク処理には状態をもたせない実装を

ワーカー
パターン

状態を共有せず、コンピューティングタスクを担うワーカーが
スケールアップ / スケールダウンしやすい状態に

IoT デバイスからPub/Sub トピックへデータをストリーミング
Cloud Functions がトリガーされる



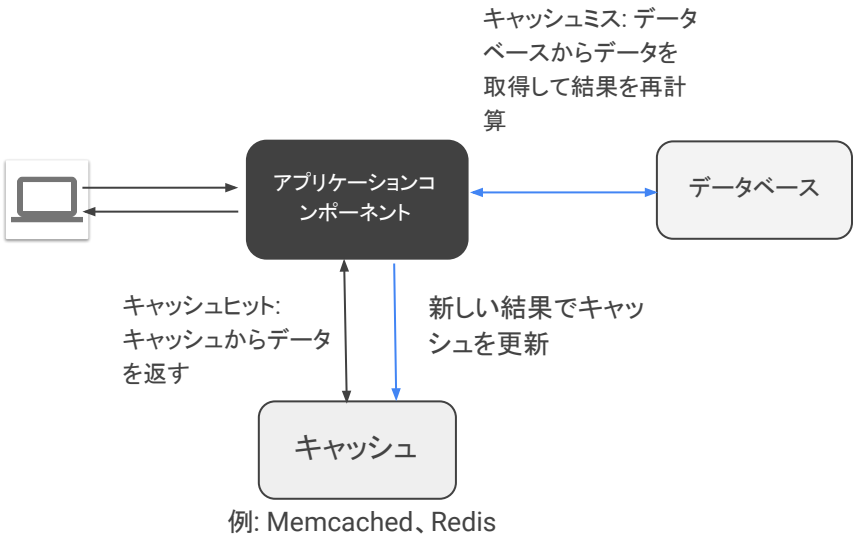
アプリコンポーネントがデータを処理、変換、保管

IoT デバイスから Pub/Sub トピックへデータを
ストリーミング
アプリコンポーネントが
トピックからメッセージをサブスクライブ

Google Cloud

コンテンツのキャッシング

アプリケーション データのキャッシング



例: Memcached、Redis

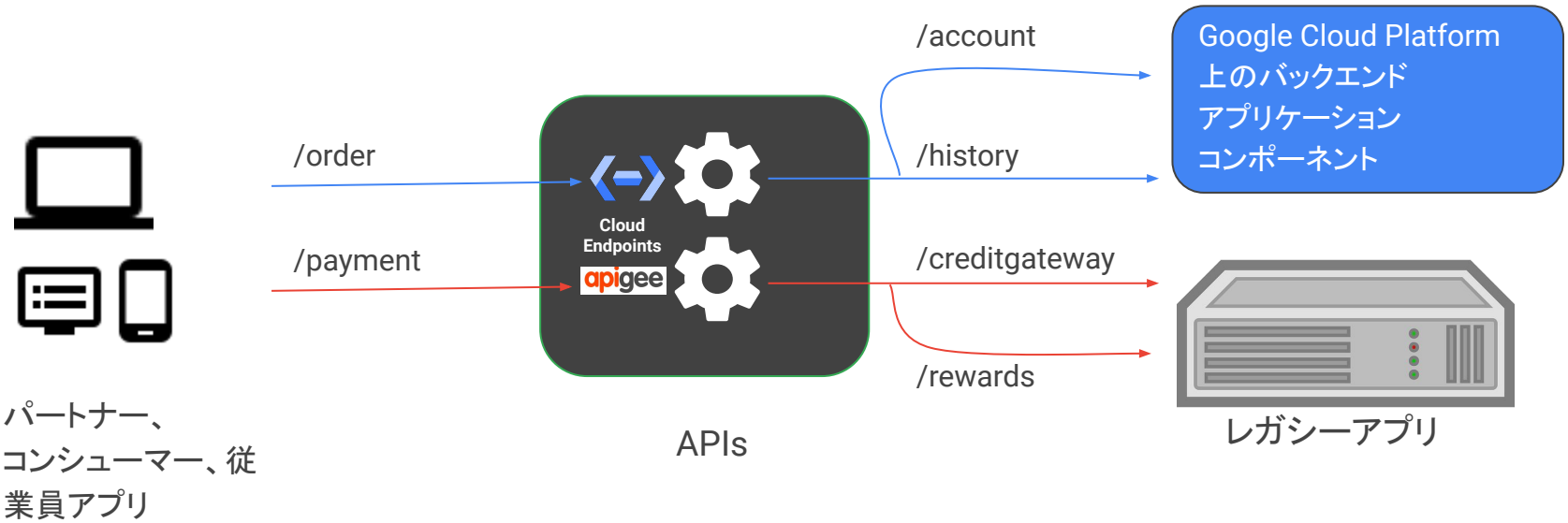
静的コンテンツのキャッシング

Cloud CDN

- Compute Engine VM インスタンス グループからの負荷分散された フロントエンド コンテンツをキャッシュ
- Cloud Storage から送られる静的 コンテンツをキャッシュ

Google Cloud

API ゲートウェイの実装による、コンシューマーアプリケーションへのバックエンド機能の公開



Google Cloud

フェデレーション ID 管理

Google でサインイン

Facebook でサインイン

Twitter でサインイン

GitHub でサインイン

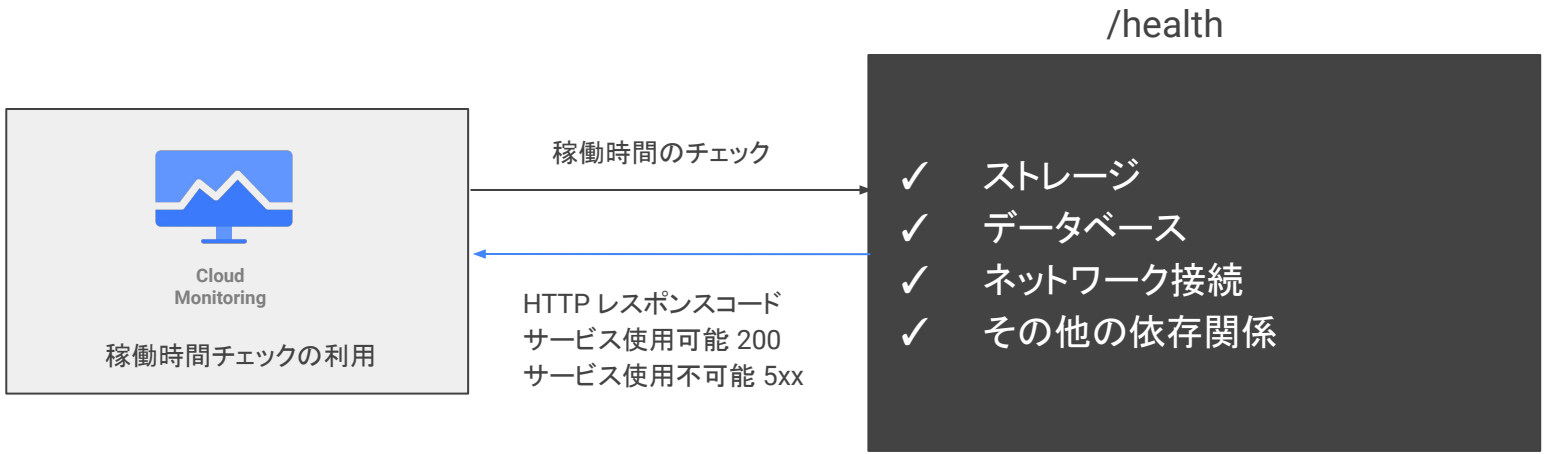
メールアドレスでサインイン

Firebase Authentication

外部 ID プロバイダを使して
ユーザーを認証

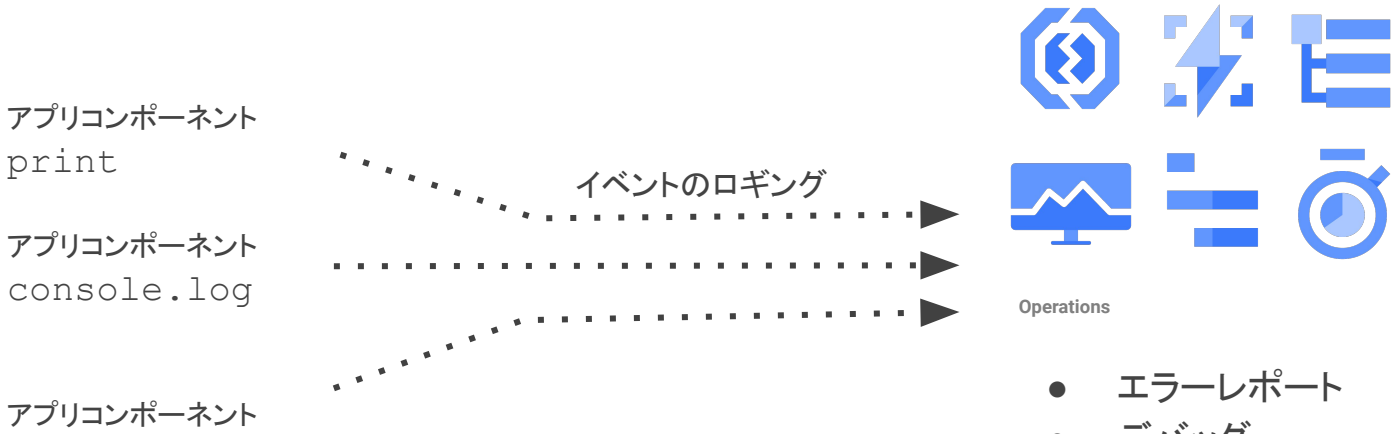
Google Cloud

ヘルスチェック エンドポイントの実装



Google Cloud

ロギングとアプリケーションパフォーマンスの監視



- エラーレポート
- デバッグ
- ロギング
- モニタリング
- トレース
- プロファイリング

Google Cloud

一時的 / 持続するエラーに対する行儀のよい対処

一時的エラー:
指数バックオフを使用したリトライ

サービス可用性エラー:
サーキット ブレーカーの実装

Google Cloud

データ主権およびコンプライアンス要件の考慮

EU - 米国間とスイス - 米国間のプライバシーシールド
フレームワーク

Google Cloud

可用性テストの実施と障害復旧の計画

機能テストとパフォーマンス テストに
加えて可用性テストを実施し、
障害復旧計画を策定する

- 障害シナリオの特定
- 障害復旧計画 (要員、プロセス、ツール) の作成
- 机上テストの実施

- カナリア テストとブルー / グリーンデプロイの実施
- 障害復旧計画の検証

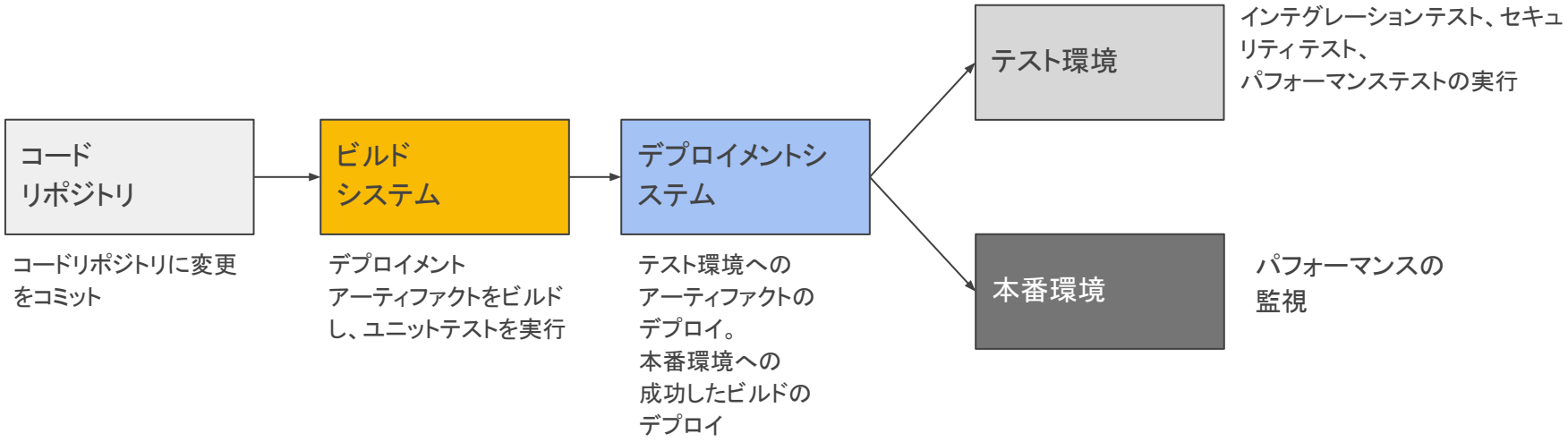
障害シナリオの例:

- 接続障害
- オンプレミス データセンターや他のクラウド プロバイダの障害
- GCP のゾーン / リージョン障害
- デプロイメントのロールバック
- ネットワークやアプリケーションの問題に起因するデータ破損

Google Cloud

CI / CD パイプラインの実装

継続的インテグレーションおよび継続的デリバリー



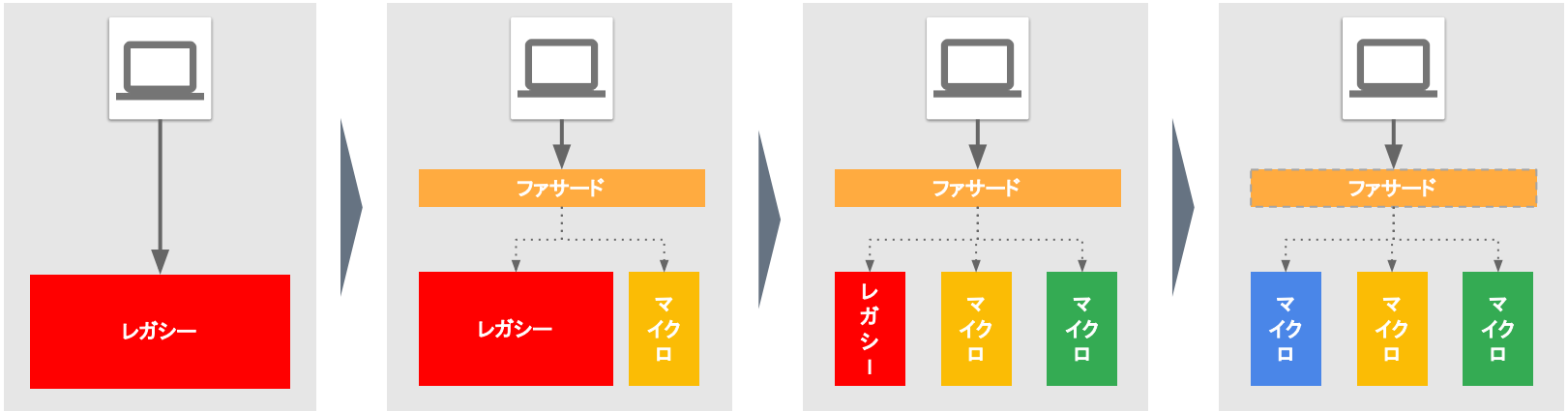
Google Cloud

疎結合 と マイクロサービス アーキテクチャ

モノリス ファーストがおすすめ
最初からモノリスで構築し、後に分割した方が成功例は多い

サービスを分割するパターンはいくつかある

- Anti Corruption Layer Pattern
- Strangler Pattern



Google Cloud

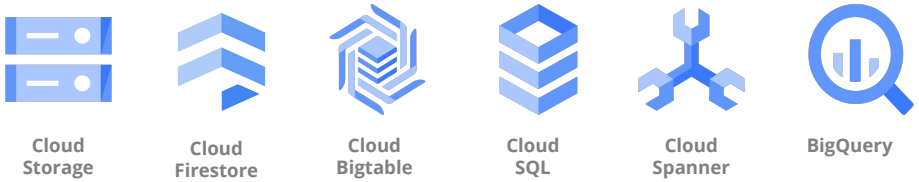
出典: Martin Fowler "MonolithFirst"
https://martinfowler.com/bliki/MonolithFirst.html

GCP を使ったアプリケーション開発

データストレージ オプションの概要

Google Cloud

GCP が提供する包括的なストレージサービスオプション



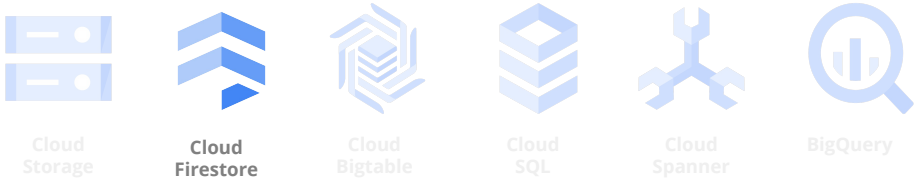
- 優れたコスト効率
- 以下に合わせて選べる多様な選択肢
  - アプリケーション
  - ワークロード

Cloud Storage

概要	最適な用途
• フルマネージド、高可用性 • コスト効率の高いスケーラブルなオブジェクト/blob ストア • HTTP リクエスト経由でオブジェクトにアクセス • オブジェクト名が唯一のキー	• 画像およびビデオ • オブジェクトおよび blob • 非構造化データ • 静的なウェブサイトホスティング

Google Cloud

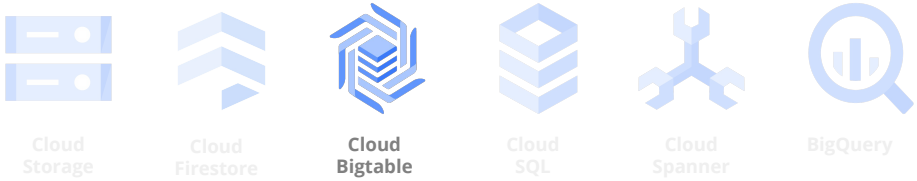
Cloud Firestore



概要	最適な用途
• フルマネージド NoSQL ドキュメント データベース • スケーラビリティ • リアルタイム同期	• 半構造化アプリケーション データ • 耐久性のある Key-Value データ • 階層型データ • 複数インデックスの管理 • トランザクション • Cloud Functions とのイベント連携

Google Cloud

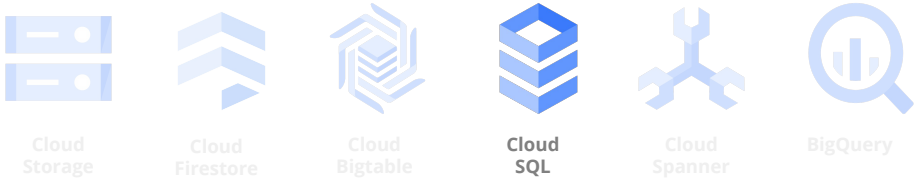
Cloud Bigtable



概要	最適な用途
• パフォーマンスが高く列数の多い NoSQL データベースサービス • 低密度で格納されるテーブル • 数十億行、数千列までスケール可能 • TB ～ PB 規模のデータを格納	• オペレーショナル アプリケーション • 分析アプリケーション • 大量のシングル キーデータの保管 • MapReduce オペレーション

Google Cloud

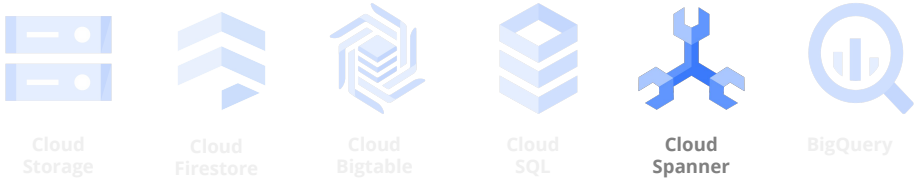
Cloud SQL



概要	最適な用途

• マネージド サービス
  ○ レプリケーション
  ○ フェイルオーバー
  ○ バックアップ
• MySQL、PostgreSQL、SQL Server
• リレーショナル データベース サービス
• プロキシによってホワイトリストなしで Cloud SQL 第 2 世代インスタンスにセキュアなアクセス
	• ウェブ フレーム ワーク
	• 構造化データ
	• OLTP ワークロード
	• MySQL / PGS を使用するアプリケーション

Cloud Spanner

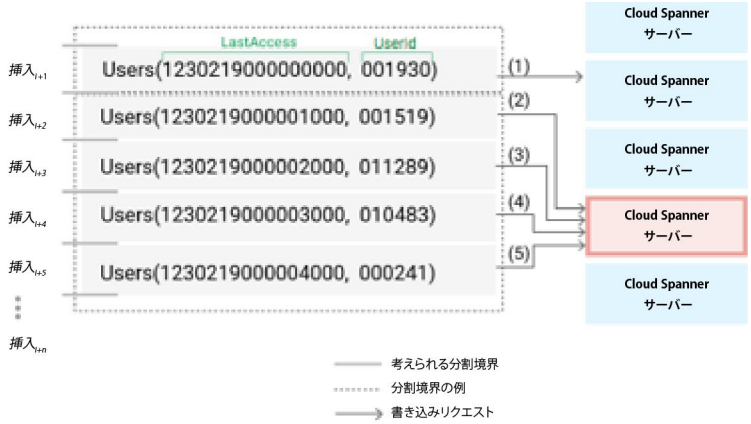


概要	最適な用途
• ミッション クリティカルなリレーショナル データベースサービス • トランザクションの一貫性 • グローバル スケール • 高可用性 • マルチリージョンのレプリケーション • 99.999 % SLA	• ミッション クリティカル アプリケーション • 大量トランザクション • スケールと整合性の要件

Google Cloud

Spanner に関する考慮事項

単調に増加するキーを回避



書き込みが十分に分散されていることを確認してから、複数のワーカーを使用してデータをロード

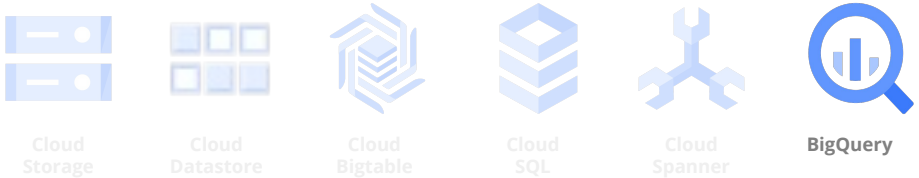
インターリーブされたテーブルを使用して階層を作成

```
-- Schema hierarchy:
-- + Singers
--   + Albums (interleaved table, child table of Singers)

CREATE TABLE Singers (
  SingerId   INT64 NOT NULL,
  FirstName  STRING(1024),
  LastName   STRING(1024),
  SingerInfo BYTES(MAX),
) PRIMARY KEY (SingerId);

CREATE TABLE Albums (
  SingerId     INT64 NOT NULL,
  AlbumId      INT64 NOT NULL,
  AlbumTitle   STRING(MAX),
) PRIMARY KEY (SingerId, AlbumId),
  INTERLEAVE IN PARENT Singers ON DELETE CASCADE;
```

単調に増加 / 減少するキーを持つ列に、インターリーブされていないインデックスを作成しない

Google Cloud

BigQuery



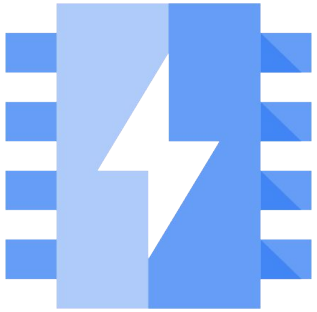
概要	最適な用途
• 低コストの分析用エンタープライズデータウェアハウス • フルマネージド • ペタバイト規模 • 高速応答時間 • サーバーレス	• オンライン分析処理 (OLAP) ワークロード • ビッグデータの探索と処理 • ビジネス インテリジェンス (BI) ツールを使用したレポート

モバイル向けのストレージオプション

	Cloud Storage for Firebase	Firestore Native mode	Firebase Hosting
概要	• Google Cloud Storage へのモバイル / ウェブアクセス • サーバーレスのサードパーティ認証および認可	• リアルタイム データ同期 • NoSQL JSON データ ベース	• ウェブ / モバイル コンテンツのホスティング • 本番グレード
Ideal for	• 画像、写真、ビデオ • オブジェクトおよび blob • 非構造化データ	• モバイル / ウェブ アプリケーション • リアルタイム	• アトミックなリリース管理 • JS アプリのサポート • Firebase 統合

Google Cloud

アプリケーションデータのキャッシング



Redis
Memcached
をサポート

フルマネージドなインメモリ データストア
高可用性の実現やフェイルオーバー、パッチ適用、
モニタリングなどが自動で行われ、開発者は開発に専念

必要に応じてスケーリング
最初はスモール スタートし、可用性への影響を最小限に抑えながら、インスタンスを容易に拡張可能

高可用性
高可用性が必要なユース ケースでは、Memorystore for Redis インスタンスは、2 つのゾーンにレプリケートされ、
99.9 % の可用性 SLA を提供

Google Cloud

ストレージのまとめ

製品	簡単な特徴	最適な用途	非推奨
 Cloud Storage	バイナリ / オブジェクト ストア	大規模またはアクセス頻度の低い非構造化データ	構造化データ、高速アプリの構築
 Firestore	スケーラブルな ドキュメント DB	モバイルアプリ トランザクション	リレーショナル または分析データ
 Bigtable	データ量の多い 低レイテンシデータベース	読み取り/書き込みの多い「フラット」データ、分析データ	高度な構造化データ、トランザクションデータ
 Cloud SQL	使い慣れた VM ベース RDBMS	ウェブフレームワーク、既存アプリケーション	スケーリング、分析、大量書き込み
 Spanner	リレーショナル DB サービス	低レイテンシの トランザクションシステム	分析データ
 BigQuery	自動スケーリング分析 データ ウェアハウス	静的データセットの インタラクティブ分析	高速アプリの構築
 Memorystore	マネージドの Redis 及び Memcache	インメモリ キャッシュ	高度な構造化データ

ストレージオプションに関する技術的な考慮事項

製品	読み取り/ 書き込み レイテンシ	代表的なサイズ	ストレージ タイプ
Cloud Storage	中 (数百ミリ秒)	任意	オブジェクト
Firestore	中 (数十ミリ秒)	< 200 TB	ドキュメント
Bigtable	低 (数ミリ秒)	2 TB ～ 10 PB	Key-Value
Cloud SQL	低 (数ミリ秒)	< 10 TB	リレーショナル
Spanner	低 (数ミリ秒)	任意	リレーショナル
BigQuery	高 (数秒)	任意	カラム型

モジュール 2
ストレージと
モニタリング

Google Cloud

Cloud OnBoard

GCP を使ったアプリケーション開発

バケットとオブジェクトに関するオペレーションの実行

Google Cloud

Google Cloud Storage の概念

リソースは Google Cloud Platform におけるエンティティで、
以下を含む

- プロジェクト
- バケット - Cloud Storage の基本コンテナ
- オブジェクト - Google Cloud Storage に保管する個別のデータ

ストレージクラス

ハイパフォーマンス
オブジェクト ストレージ

バックアップ & アーカイブ

Standard		Nearline	Coldline	Archive
コンテンツを グローバルに提供	頻繁にアクセス 高スループット	月に 1 回程度の アクセス	1 年に 1 回程度の アクセス	長期保存
マルチリージョン	リージョナル			
ビデオ ストリーミング	ビデオ トランス コーディング	アクセスが 少ない ドキュメント	ほとんど 使用されない データ	規制対応
画像	ゲノミクス	バックアップ	画像 アーカイブ	テープの 置き換え
ウェブサイト	データ分析		災害対策	
ドキュメント				

Google Cloud

強整合性が確保されるオペレーション

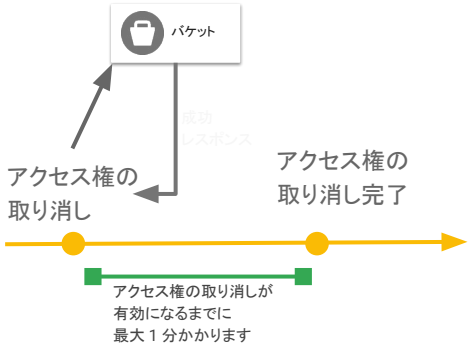
read-after-write
read-after-metadata-update
read-after-delete
バケットの一覧表示
オブジェクトの一覧表示
リソースへのアクセス権の付与

強整合性: Cloud Storage でオペレーションを実行して成功レスポンスを受信すると、オブジェクトが直ちにダウンロードおよびメタデータ オペレーションの対象となる

Google Cloud

結果整合性が確保されるオペレーション

オブジェクトに対するアクセス権の取り消し
一般公開された読み取り可能なキャッシュ済み
オブジェクトへのアクセス



結果整合性：オペレーションを実行するとき、
オペレーションが有効になるまでにしばらく
時間がかかる場合がある

Google Cloud

使用できるリクエストエンドポイント

	URIs	HTTP	HTTPS

通常の API リクエスト	XML:
	`storage.googleapis.com/<bucket>/<object>`
	`<bucket>.storage.googleapis.com/<object>`
	JSON:
	`www.googleapis.com/download/storage/v1/b/<bucket>/o/<object-encoded-as-URL-path-segment>?alt=media`
		✔	✔

CNAME リダイレクト	CNAME レコードのホスト名部分で以下のURI を使用します。 `c.storage.googleapis.com.` たとえば、`travel-maps.example.com CNAME c.storage.googleapis.com`と公開した場合、 `http://travel-maps.example.com/paris.jpg`を使用してオブジェクトにアクセスできます。	✔	
認証済ブラウザでのダウンロード	Cookie ベースの認証を使用してオブジェクトをダウンロードするには、以下のURI を使用します。 `https://storage.cloud.google.com/<bucket>/<object>`		✔
コンテンツベースの負荷分散	使用しているCloud Storage バケットのバックエンドバケットを作成し、`web-map` URL マップを変更します。以下のどちらかを使用してリソースをフェッチします。 `https://[IP_ADDRESS]/static/[OBJECT_NAME]`または `https://[IP_ADDRESS]/static/[OBJECT_NAME]`	✔	✔

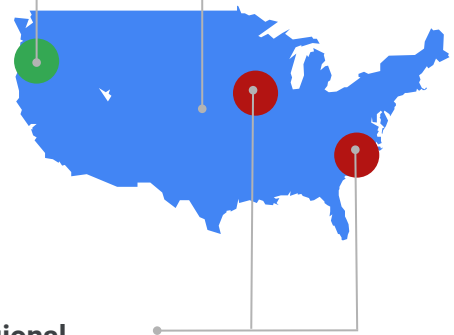
Google Cloud

データのロケーションと可用性を考慮したストレージオプションの選択

ロケーションタイプ

Regional

特定のリージョンにデータを保管。ゾーン間でデータレプリケーション



Multi-regional

選択した地域（US, EU, アジア）のリージョン内に分散してデータを保管

Dual-regional

ペアとなるリージョンにデータレプリケーション

Google Cloud

バケットのセキュリティ制御オプション

Identity and access Management (IAM) を使用して付与できる権限

- バケットへのアクセス
- バケットのオブジェクトへの一括アクセス

アクセス制御リスト (ACL) を使用して付与できる権限

- 個別バケット / オブジェクトに対するユーザーの読み取り / 書き込みアクセス
- 個別オブジェクトへのきめ細かい制御が必要な場合のアクセス

署名付き URL (クエリ文字列認証)

- 生成 URL を介して、時間制限付きの読み取り / 書き込みオブジェクトアクセス権を提供
- Gsutil またはプログラムを使用して作成可能

署名付きポリシードキュメントで実行できる処理

- バケットにアップロードできる内容を指定
- サイズ、コンテンツタイプなどのアップロード特性を制御

Firebase セキュリティルール

- Cloud Storage 用の Firebase SDK を使用して、モバイル / ウェブアプリへのアクセスを属性ベースできめ細かく制御

Google Cloud

GCP を使ったアプリケーション開発

デバッグとモニタリング

Google Cloud

GCP オペレーションスイート一覧

Error Reporting

エラーの通知
エラー ダッシュボード

Cloud Debugger

本番のデバッグ スナップ ショット
条件付きスナップ ショット
IDE 統合

Cloud Logging

プラットフォーム、システム、
アプリのログ
ログの検索 / 表示 / フィルタ
ログベースの指標

Cloud Monitoring

プラットフォーム、システム、
アプリの指標
稼働時間/ヘルスチェック
ダッシュボード
アラート

Cloud Trace

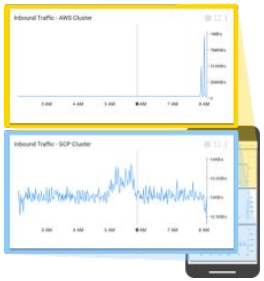
レイテンシレポート
URL 単位のレイテンシサンプリング

Cloud Profile

影響を抑えた本番
アプリケーションのプロファイリング

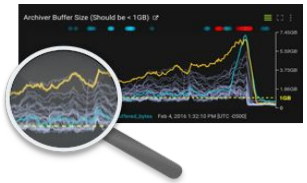
Google Cloud

オペレーションスイートによるアプリケーションの信頼性向上



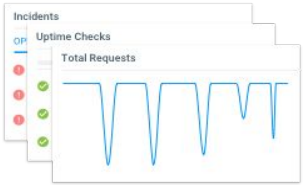
GCP、AWS、マルチクラウド環境の監視

最小限の設定で必要なインサイトを獲得
ホステッドサービスとクラウドアーキテクチャを監視



トレンドの識別と問題の防止

柔軟なグラフとダッシュボードでトレンドを視覚化
スコアリング、異常検出、予測を使用してリスクを識別



モニタリングオーバーヘッドの軽減

異なるシステム間での指標、アラート、ログの関連付けにかかる時間を短縮
スケーリング
ツールに関する心配がない

シグナル/ノイズ比の改善

最新の分散システム向けに設計された高度なアラート機能により、誤検出とアラート疲れを削減

迅速な問題解決

稼働時間およびヘルスチェックにより、エンドポイントのアクセス不可を迅速に通知
アラートからダッシュボード、ログ、トレースへとドリルダウンすることで、根本原因を素早く特定

Google Cloud

アプリケーションのデバッグ

Google Cloud

開発および本番環境でのアプリケーションのデバッグ

Error Reporting

Errors in the last hour

Occurrences	Error
679	**IndexOutOfBoundsException: Index: 4, Size: 4** com.callbyreference.demos.markov.Markov.isPromising (Markov.java)

Stack trace sample

Parsed Raw

```
java.lang.IndexOutOfBoundsException: Index: 4, Size: 4
    ...
    at com.callbyreference.demos.markov.Markov.isPromising (Markov.java:76)
    at com.callbyreference.demos.markov.Markov.generate (Markov.java:55)
    at com.callbyreference.demos.markov.Template$MarkovDiv.generateBegin (Template.java:113)
    at com.callbyreference.demos.markov.Template.generate (Template.java:20)
    at com.callbyreference.demos.markov.Template.generate (Template.java:22)
    at com.callbyreference.demos.markov.Template.generate (Template.java:22)
    at com.callbyreference.demos.markov.MarkovTemplate.generate (MarkovTemplate.java:57)
    at com.callbyreference.demos.markov.MarkovServlet.doGet (MarkovServlet.java:115)
    ...
```

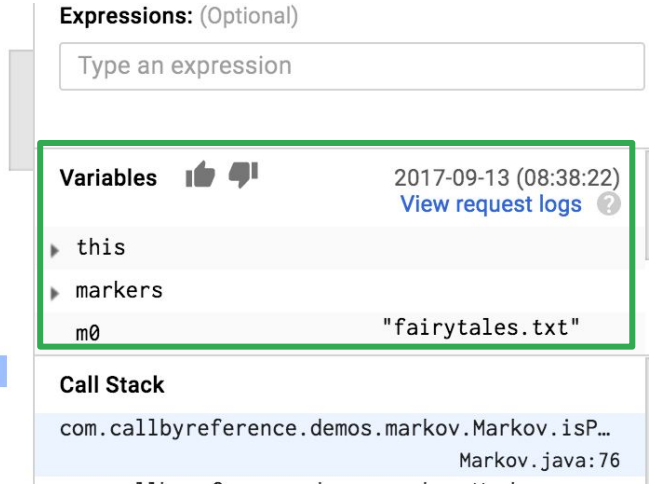
Google Cloud

Debugger によるデバッグスナップショットの自動作成

Cloud Debugger

新しいデバッグ スナップショットを
生成するためにアプリケーションを
再起動する必要はありません。

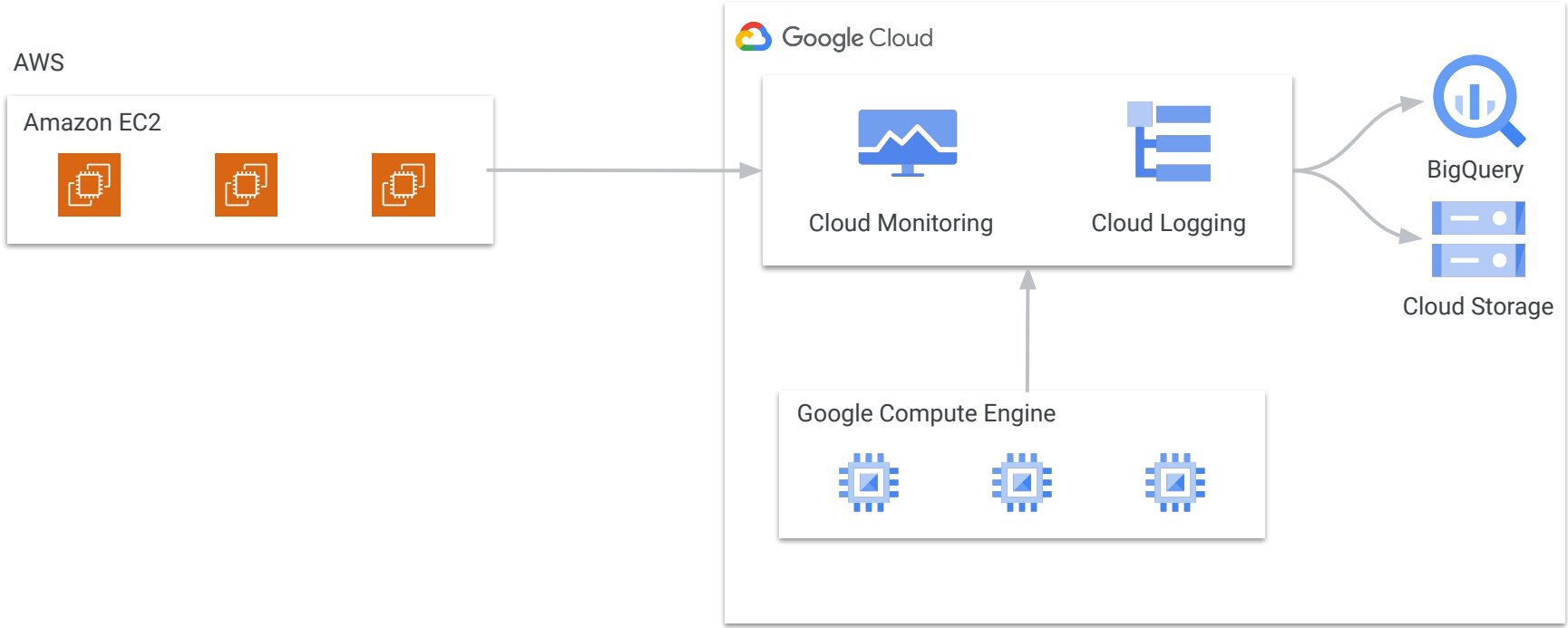
```
65  private boolean isPromising(List<Object> markers) {
66    if (markers.size() > 32)
67      return false;
68    Object m0 = markers.get(0);
69    Object m_1 = markers.get(markers.size() - 1);
70    if (m0 == m_1) {
71      return false;
72    }
73    Object m1 = markers.get(1);
74    Object m2 = markers.get(2);
75    Object m3 = markers.get(3);
76    Object m4 = markers.get(4);
77    if (m0 != m1 || m0 != m2 || m0 != m3 || m0 != m4
78        || m0 != markers.get(5)) {
79      return false;
```



Google Cloud

Logging

Cloud Logging エージェントのインストールによるログの取得



Google Cloud

Cloud Logging が事前設定されている
その他のコンピューティング環境

Google Cloud Run

Google Cloud Functions

Google App Engine
フレキシブル環境およびスタンダード環境

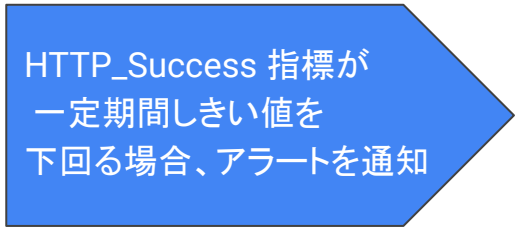
Google Kubernetes Engine

Google Cloud

ログベースの指標およびアラートのセットアップ

Cloud Logging

09:21:51.246GET2001.13 KB6 exampleapp/
- - [14/Sep/2017:09:21:51 -0700] "GET / HTTP/1.1" 200 1156 - "exampleapp"
"exampleapp-git.appspot.com" ms=6 cpu_ms=11
cpm_usd=1.2919299999999998e-7 loading_request=0 instance=some_instance_id
app_engine_release=1.9.54
Expand all | Collapse all
{
httpRequest: {
status: 200
}



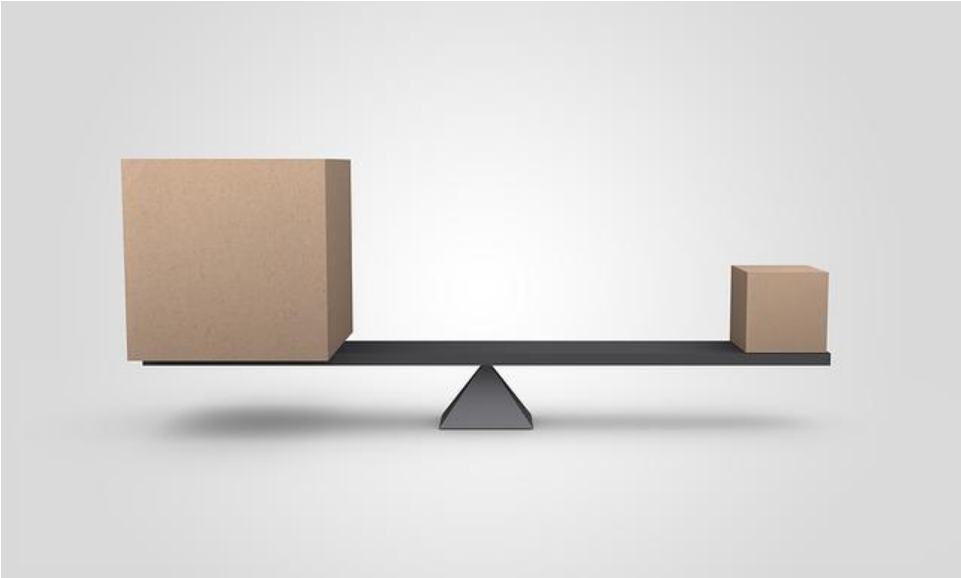
モニタリングとパフォーマンスチューニング

Google Cloud

監視を通じた長期トレンドの分析

Google Cloud

監視を通じた経時的な結果比較と、試験構成間での結果比較



Google Cloud

監視を通じた障害の発生時または発生しそうな場合の注意喚起

監視を通じたアドホックな回顧的分析の実施

Google Cloud

監視する API とリソースの特定

例

- パブリック エンド ポイントとプライベート エンド ポイント
- Compute Engine VM インスタンス、Cloud Storage バケット、Amazon EC2 インスタンス、Amazon RDS データベースをはじめとするマルチクラウド リソース

サービス レベル指標とサービス レベル目標の特定

サービス レベル指標 (SLI): レイテンシ
サービス レベル目標 (SLO): 30 日間のリクエストの 99.9 % が 100 ミリ秒未満のレイテンシ

Google Cloud

4 大シグナルを含むダッシュボードの作成

レイテンシ

トラフィック
(リクエスト数)

エラー

サチュレーション

パフォーマンス問題の特定と
トラブルシューティング

Google Cloud

開発環境と本番環境におけるパフォーマンスの監視

Google Cloud

テスト スイートへのパフォーマンス テストの追加

開発環境

Google Cloud

受信リクエストに関連する
パフォーマンスウォッチ ポイントの確認

Web ページ分析

Web ページの分析ツールを
利用し、パフォーマンスに
関する潜在的な問題や
改善可能な点を確認する

コールドブートの
パフォーマンス

トレーシングを利用し、
コールド ブート時における
各処理の処理時間を確認する

アプリケーション自体による
負荷

クライアント サイドや
サーバ サイドの分析ツールを
利用し、アプリケーションの負荷
状況を確認する

開発環境

Google Cloud

パフォーマンス問題発生時の
アプリケーション コードとログの確認ポイント

アプリケーション エラー

HTTP エラーと例外エラーを確認し、ログメッセージの根本原因を特定する

静的リソース

静的コンテンツは
Cloud CDN 等の CDN
サービスを利用し配信する

キャッシング

データベースから頻繁に
取得するデータや計算コストの高いデータはキャッシングする

複数リクエスト送信

複数のリクエストを
順次送信するような処理は、単一のバッチリクエストに
置き換えるかリクエストを
並列送信する

エラー処理

エラー発生時の再試行には
指数バックオフを利用する
一定の回数失敗した場合は
試行を止めるようサーキットブレーカーを実装する

開発環境

Google Cloud

本番環境での、受信リクエストに関連するパフォーマンスウォッチポイントの確認

外部ユーザーのロード

最も頻繁に受信する
リクエストと最も低速な
リクエストを分析する

定期的ロード

長い期間のトラフィックを
分析し、使用率が高い期間を特定する

悪意のあるロード

トラフィックが正当な
クライアントから送信されている
ことを確認する

本番環境

Google Cloud

デプロイメント設定の確認

スケーリング

適切なロードバランシングと自動スケーリングが設定されていることを確認する

リージョン

トラフィックの送信元に近いリージョンにデプロイし、レイテンシを低減する

クーロンジョブ

クーロンジョブが正確にスケジューリングされていることを確認する

本番環境

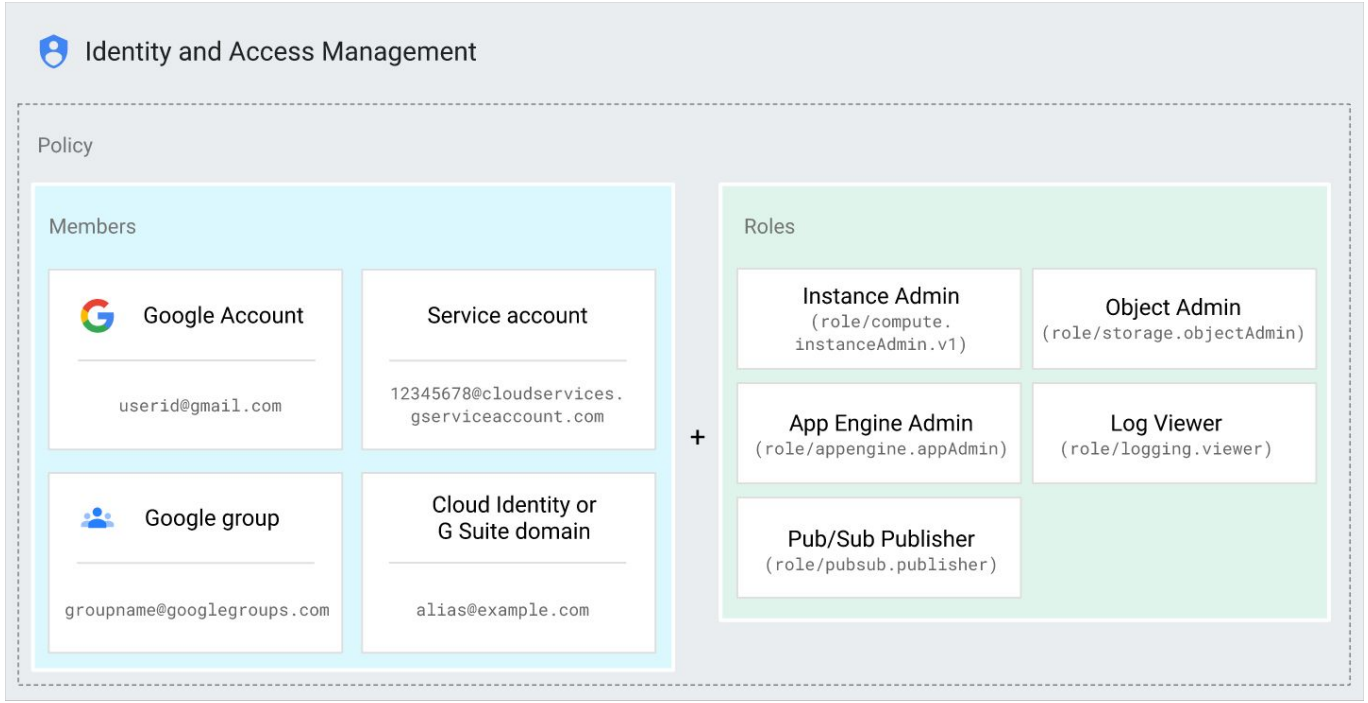
Google Cloud

GCP を使ったアプリケーション開発

認証と認可の取り扱い

Google Cloud

Cloud IAM (Identity and Access Management)



Google Cloud

IAM メンバーを使用した、アクセスできるユーザーの指定

IAM メンバーの種類:

- Google アカウント
- サービスアカウント
- Google グループ
- G Suite ドメイン
- Cloud Identity ドメイン

Google Cloud

メンバーがアクセスできるリソースの指定

- 特定の GCP リソースへのアクセス権をユーザーに付与
- 対象となるリソース例:
  - GCP プロジェクト
  - Compute Engine インスタンス
  - Cloud Storage バケット
  - Pub/Sub トピック

リソースで許可される操作の指定

権限は以下の構文で表されます。

`<service>.<resource>.<verb>`

例:

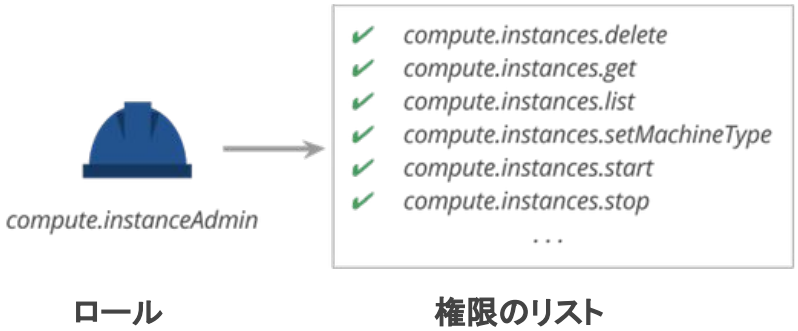
`pubsub.subscriptions.consume`

`storage.objects.list`

`compute.disktypes.list`

Google Cloud

ロールを使用した権限の付与



IAM ロールは 3 種類あります。

基本ロール
事前定義ロール
カスタムロール

Google Cloud

プロジェクト レベルでの基本ロールの適用

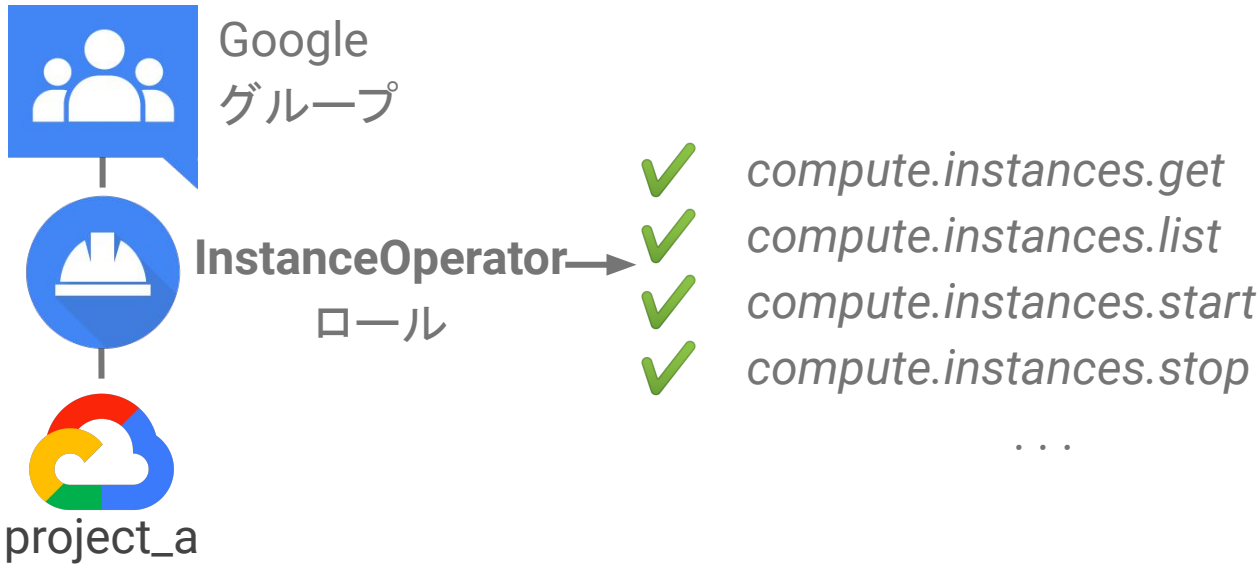
ロール名	ロール タイトル	権限
roles/viewer	閲覧者	状態を維持する読み取り専用アクションを行うために必要な権限。
roles/editor	編集者	閲覧者権限、および状態を変更するアクションを行うために必要な権限。
roles/owner	オーナー	編集者権限、および以下を行うために必要な権限。 ● プロジェクトおよびプロジェクトのすべてのリソースに対するアクセス制御を管理する。 ● プロジェクトの課金情報を設定する。

GCP リソースへのきめ細かいアクセス権を定めた事前定義ロールの適用

事前定義ロールでは、特定のGCP に対するきめ細かいアクセス権を定めることができます。
同じユーザーに複数のロールを付与することもできます。

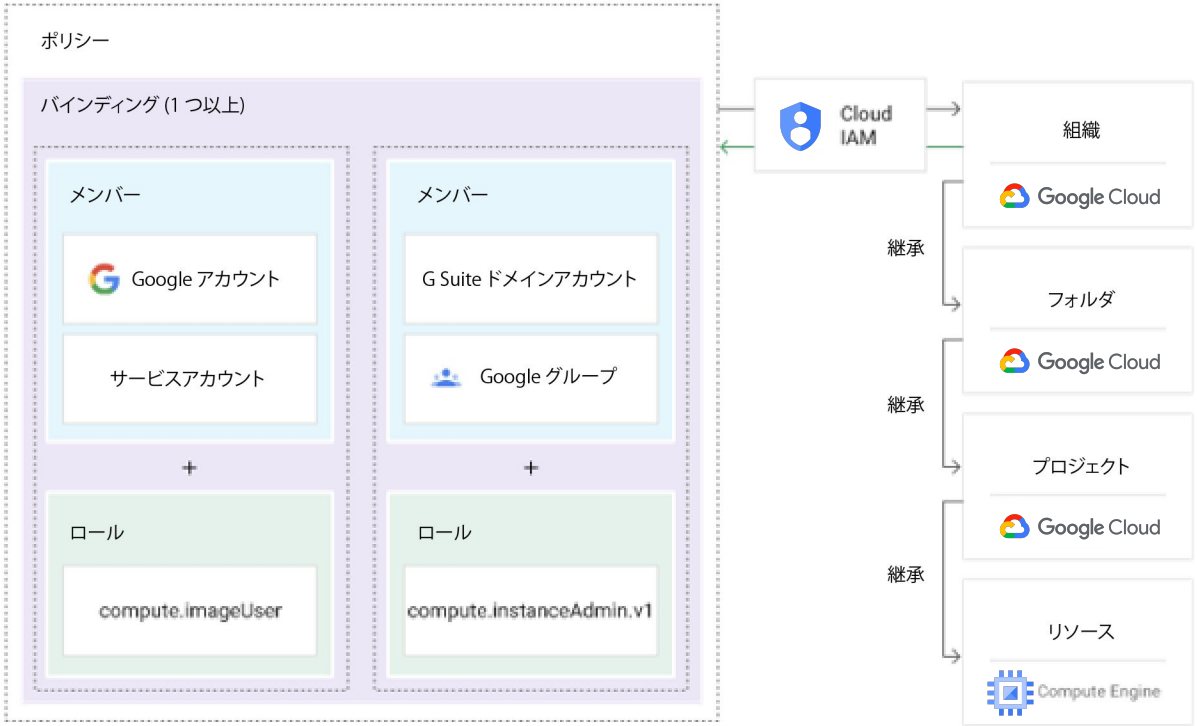
ロール名	ロールタイトル	説明	リソースタイプ
roles/bigtable.admin	Cloud Bigtable 管理者	テーブル内に保存されているデータなど、 プロジェクト内のすべてのインスタンスを管理する。新しいインスタンスを作成できる。 プロジェクト管理者を対象とする。	組織 プロジェクト インスタンス
roles/bigtable.user	Cloud Bigtable ユーザー	テーブル内に保存されたデータへの読み取り 書き込みアクセス権を提供する。アプリケーション デベロッパーやサービスアカウントを対象とする。	組織 プロジェクト インスタンス
roles/bigtable.reader	Cloud Bigtable リーダー	テーブル内に保存されたデータへの読み取り専用 アクセス権を提供する。データサイエンティスト、ダッシュボード生成ツール、その他のデータ分析 シナリオを対象とする。	組織 プロジェクト インスタンス

正確な権限一式を定義できる IAM カスタムロール



Google Cloud

ポリシーを使用した、アクセス権の *種類* と そのアクセス権を持つユーザーの定義



Google Cloud

Cloud IAM ポリシーの例

```
{
      "bindings": [
      {
            "role": "roles/owner",
            "members": [
                  "user:alice@example.com",
                  "group:admins@example.com",
                  "Domain:google.com",
                  "serviceAccount:my-other-app@appspot.gserviceaccount.com"
            ]
      },
      {
            "role": "roles/viewer",
            "members": ["user:bob@example.com"]
      }]
}
```

ポリシーオーナー

ポリシー閲覧者

Cloud IAM API メソッド:

- setIAMPolicy()
- getIAMPolicy()
- testIamPermissions()

Google Cloud

サービスアカウントを使用した、Google API 呼び出し時のアプリケーション認証

サービスアカウントの特徴:

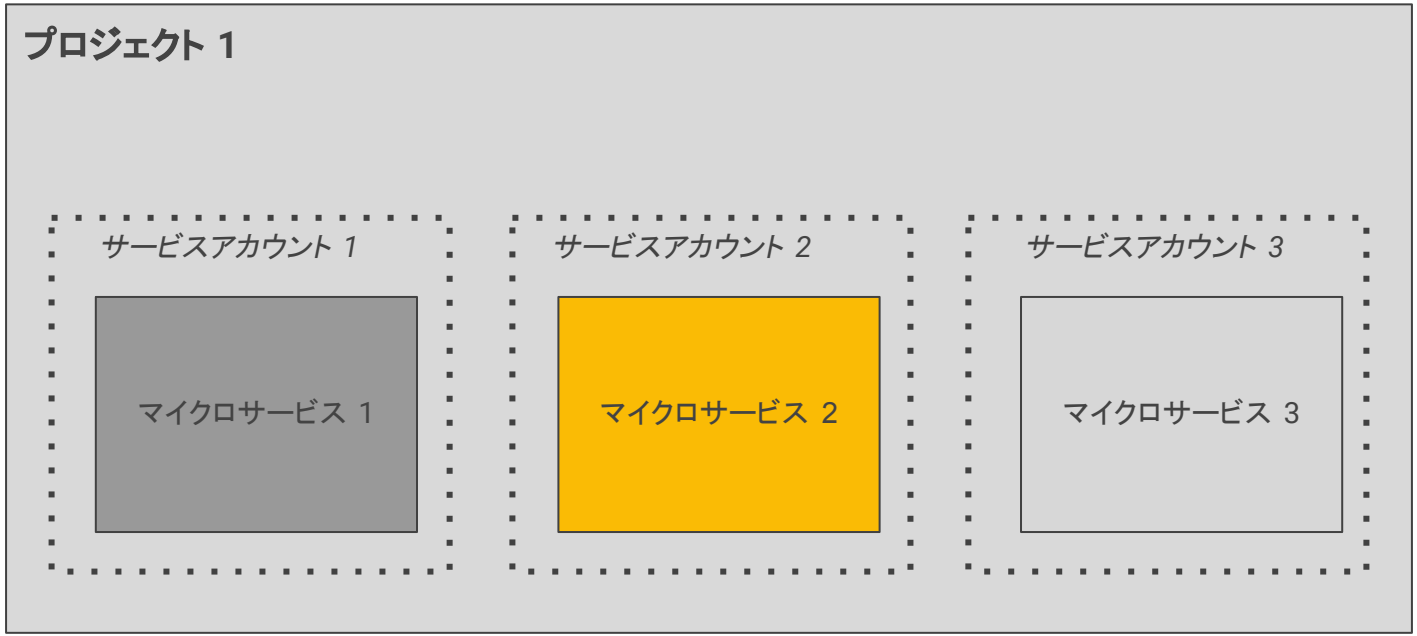
- アプリケーションまたは VM に属する。
- アプリケーションが Google API や Google サービスを呼び出すために使用する。そのため、ユーザーは直接関与しない。
- 一意のメールアドレスによって識別される。
- 鍵ペアに関連付けられている。
- キーのローテーションを容易にするために、最大で 10 のキーと関連付けることができる (キーのローテーションは、Google によって毎日実施される)。
- すべての GCP API でサポートされる。
- 特定の IAM ロールをサービスアカウントに割り当てることができるため、認証と認可が可能である。

GCP の外で使用する外部キーの作成

外部キーの特徴:

- 作成して、GCP の外で使用できる。
- 秘密鍵のセキュリティ、およびキーのローテーションをはじめとするその他の管理操作に対する責任は、お客様が担う必要がある。
- 以下を介して管理できる。
  - IAM API
  - gcloud コマンドラインツール
  - GCP Console の [Service Accounts] ページ

1つのプロジェクトに多数のサービスアカウントを定義可能



Google Cloud

サービスアカウントをお使いのアプリケーションで使用する手順

1. コンソールからサービスアカウントを作成する。
2. 認証情報ファイルを生成してダウンロードする。
3. 環境変数を設定して認証情報をお使いのアプリケーションに提供する。
4. デフォルトの認証情報を使用してコードを認証する。

Linux または OS X:

```
export GOOGLE_APPLICATION_CREDENTIALS=<path_to_service_account_file>
```

Windows:

```
set GOOGLE_APPLICATION_CREDENTIALS=<path_to_service_account_file>
```

```
def implicit():
    from google.cloud import storage

    storage_client = storage.Client()

    # Make an authenticated API request
    buckets = list(storage_client.list_buckets())
    print(buckets)
```

クライアントの構築時に認証情報を指定しない場合、クライアントライブラリは環境の認証情報を検索する。

Google Cloud

アプリケーションのデフォルト認証情報 (ADC) を使用したアプリケーション間認証

ADC は、以下の順序で認証情報を確認する。

1. `GOOGLE_APPLICATION_CREDENTIALS` 環境変数を確認する。
2. デフォルトのサービスアカウントを確認する。
3. 1 および 2 のいずれも見つからなかった場合はエラーがスローされる。

```
def implicit():

    from google.cloud import storage

    storage_client = storage.Client()

    buckets = list(storage_client.list_buckets())

    print(buckets)
```

認証済みの API リクエストを実行する。

クライアントの構築時に認証情報を指定しない場合、クライアントライブラリは環境の認証情報を検索する。

Google Cloud

Google Cloud

Qwiklabs マニュアル

1. Qwiklabs とは
2. ユーザ作成
3. ラボの始め方
4. ラボの進め方
5. ラボの終了

Google Cloud

Google Cloud

Qwiklabs とは

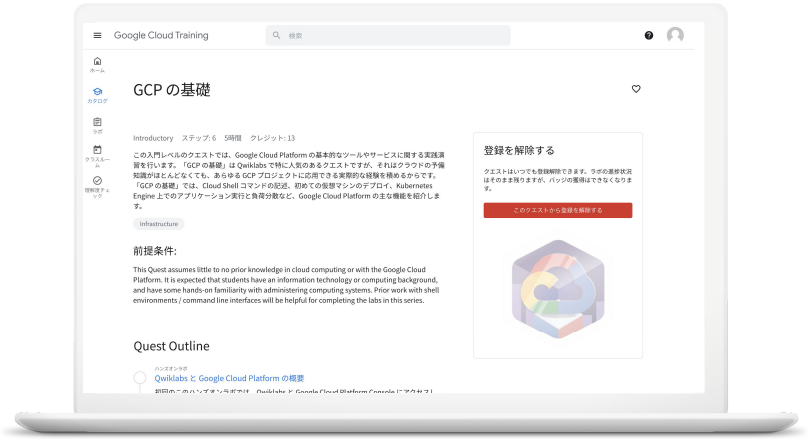
Google Cloud Platform を実際に使って演習（ハンズオン）する環境を
提供するプラットフォームです。
ご自分のレベルに合わせた演習問題を選択し、 GCP をご自分のペースで
学習できます。

ハンズオンラボ は一つ一つ独立した演習になりますが、
目的別に**クエスト**をご用意しております。
クエストには学習目標に合わせて、いくつかのハンズオンラボがセットになっています。
1 つのラボは、早いものだと 10 分程度から 30 分程度で完了できます。

ラボを楽しんでいただくためにユーザーを作成いただきますが、
Qwiklabs ユーザーは GCP アカウントとは異なります。

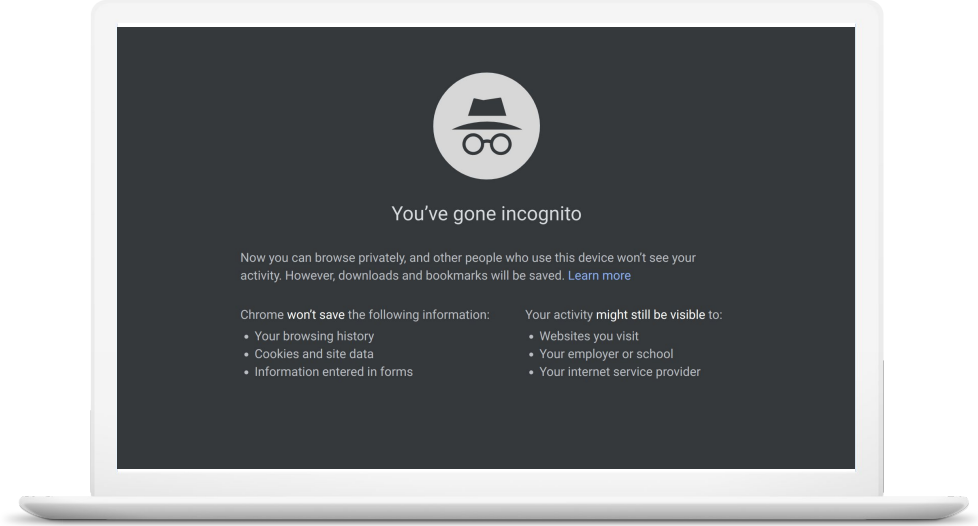
演習で利用する GCP アカウントは演習実行中、貸出しされます。
（演習終了後は利用不可となります。）

今回、無料でこちらのハンズオンラボを楽しんでいただけますので、
お楽しみください。



Google Cloud

Qwiklabsページを Incognito / Private モードで開く



Google Cloud

Google Cloud

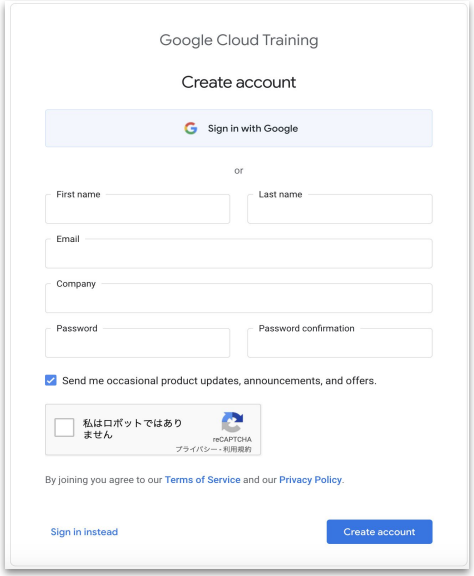
https://goo.gle/howtojoin

Qwiklabs ユーザの作成方法

1. https://goo.gle/howtojoin にアクセスしてください。
2. 右上の「Qwiklabs」をクリックして、Qwiklabs のウェブサイトを開いてください。
3. 初めてご利用になられる方は、ページ右上の「参加」ボタンから、アカウントを作成します。
（既にアカウントをお持ちの方は「サインイン」してください ）

＊ ユーザーの作成方法は 2種類あります。

1. Google 認証を使用してユーザー作成する場合は **[Sign in with Google]** を選択します。
2. その他メールアドレスを使用する場合は必要情報を入力し、作成します。



https://goo.gle/howtojoin

Qwiklabs ユーザの作成方法

4. (1)-1 Google 認証を使用する場合
Gmailまたは Google アカウントが紐付けられたアカウントのメールアドレスを入力します。

Google Cloud

Google Cloud

https://goo.gle/howtojoin

Qwiklabs ユーザの作成方法

4. (1)-2 そのメールアドレスのパスワードを入力します。

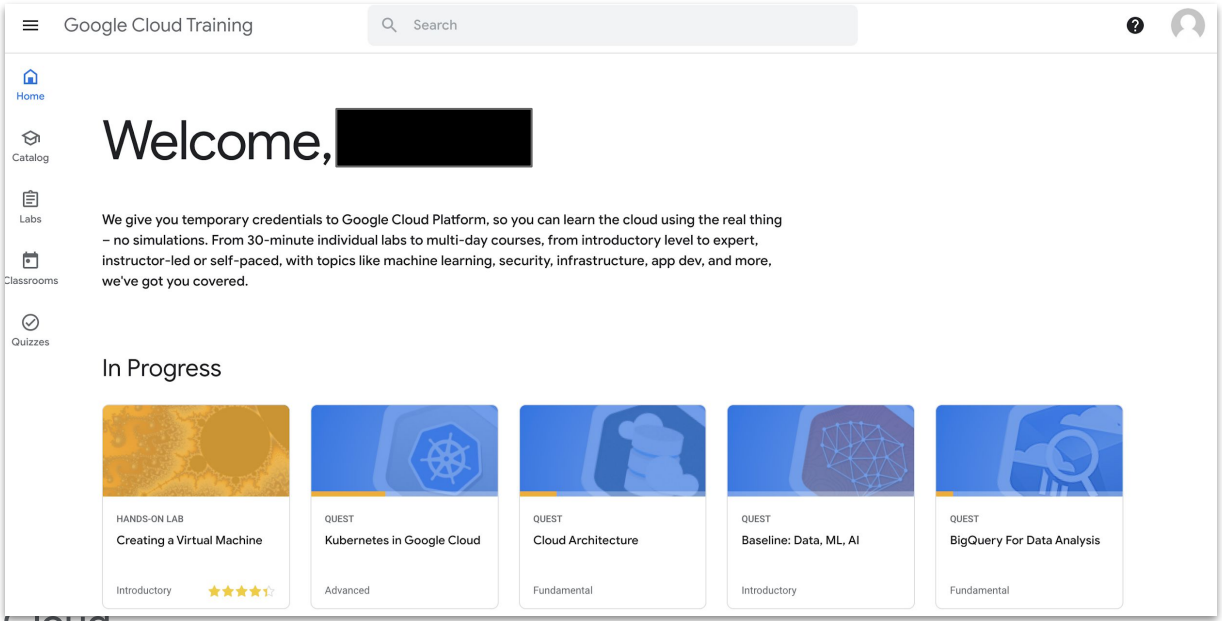
Google Cloud

Google Cloud

https://goo.gle/howtojoin

Qwiklabs ユーザの作成方法

4. (1)-3 ログイン完了です。

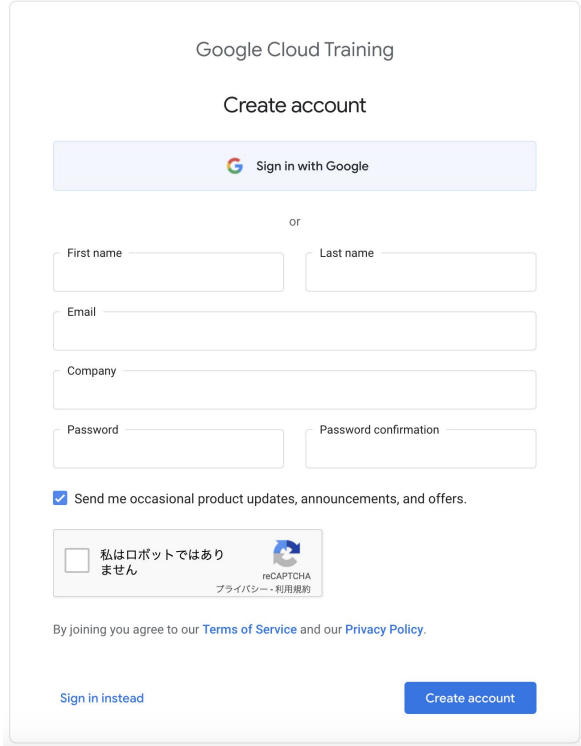


Google Cloud

https://goo.gle/howtojoin

Qwiklabs ユーザの作成方法

4. (2)-1 その他メールアドレスを使用する場合
   必要情報を入力します。

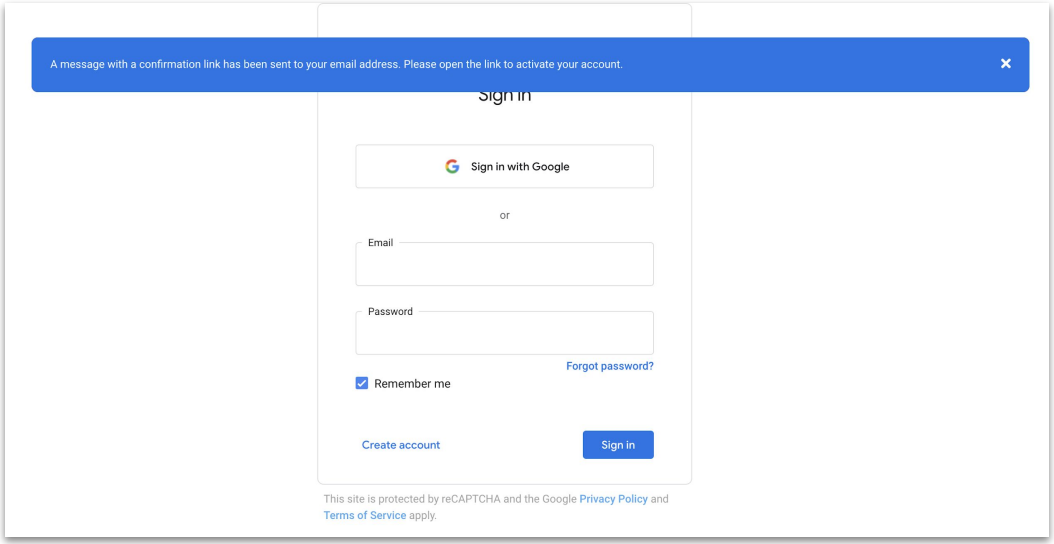


Google Cloud

https://goo.gle/howtojoin

Qwiklabs ユーザの作成方法

4. (2)-2 確認用リンクが登録メールアドレスに送信されます。



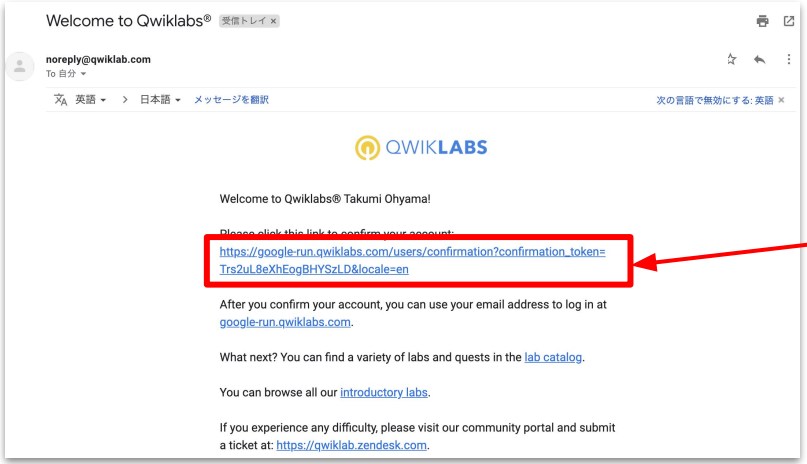
Google Cloud

Google Cloud

https://goo.gle/howtojoin

Qwiklabs ユーザの作成方法

4. (2)-3 メールに届いた確認用リンクをクリックして、アカウントの認証を行います。



メール内リンクをクリック

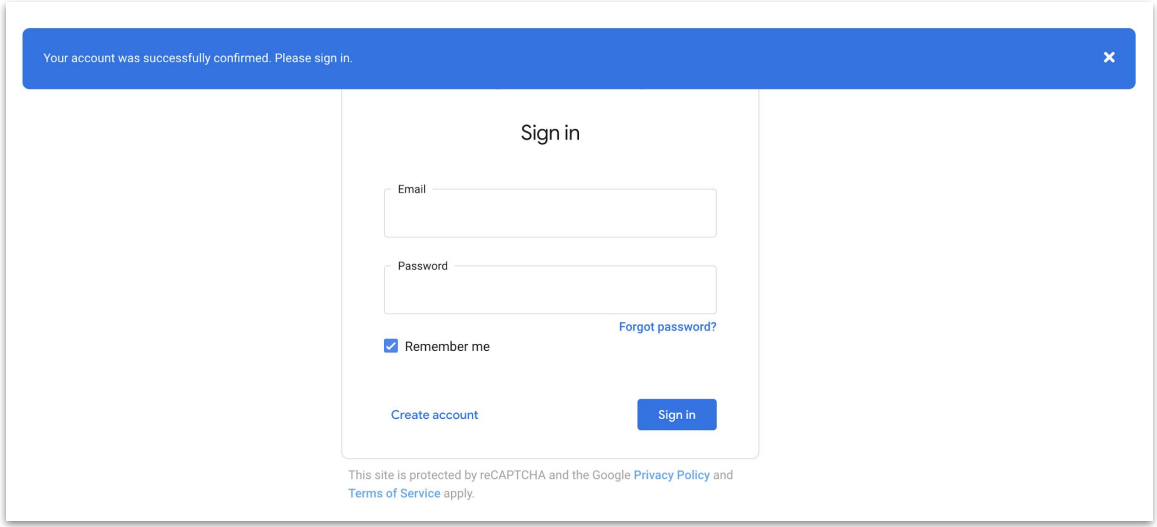
Google Cloud

Google Cloud

https://goo.gle/howtojoin

Qwiklabs ユーザの作成方法

4. (2)-4. 確認用リンクをクリックすると、認証が完了します。
登録したメールアドレスとパスワードでサインインします。



Google Cloud

ラボの始め方

本日のクエスト：Google Developer Essential
goo.gle/appdev0618

Google Cloud

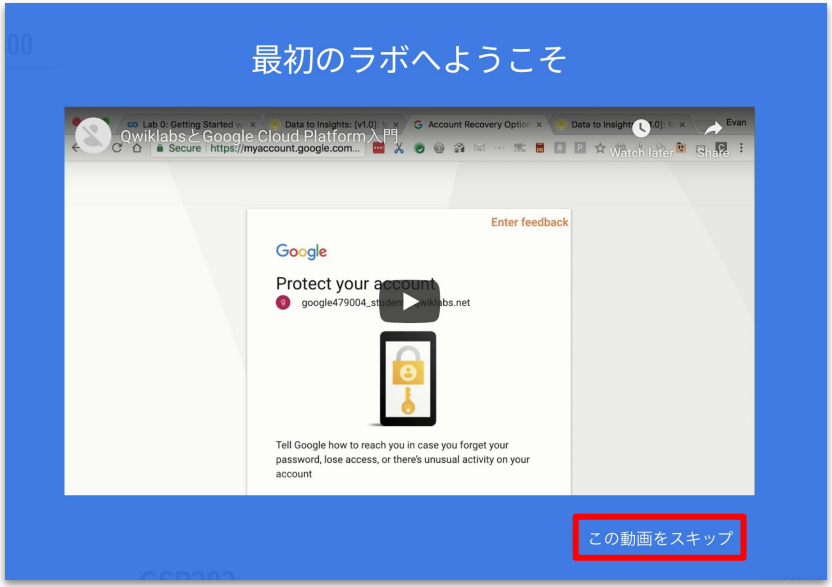
ラボの始め方

1. キャンペーンリンクにアクセスします。
   右側の「このクエストに登録する」をクリックします。

Google Cloud

ラボの始め方

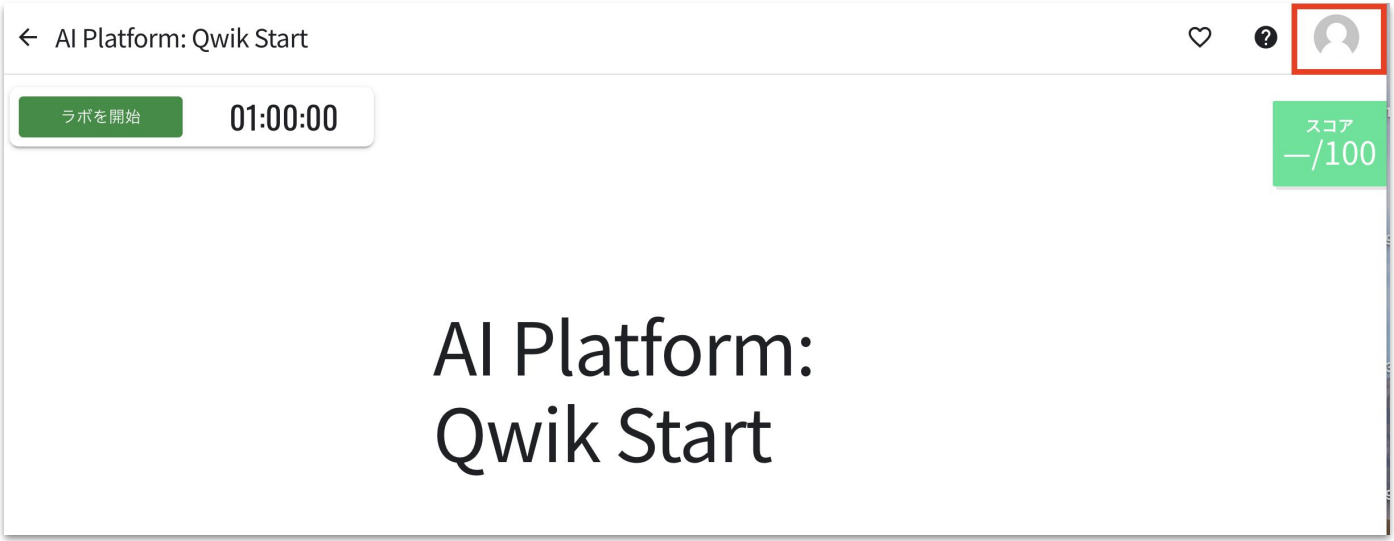
2.ラボを選択すると下記のようなチュートリアル動画が表示されることがあります。
こちらは [スキップ] してください。



Google Cloud

ラボの始め方

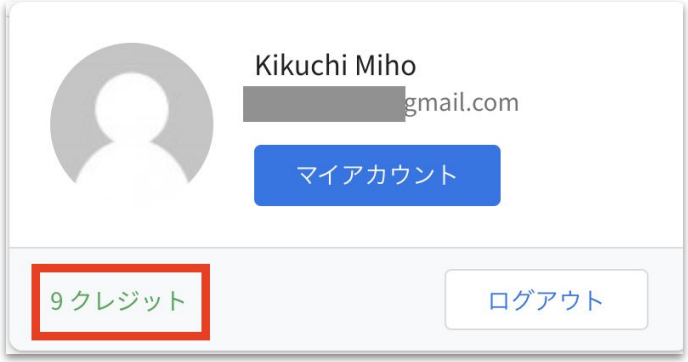
3. ラボの画面に切り替わったら、画面右上の人のマークをクリックしてください。



Google Cloud

ラボの始め方

4. 数クレジットが付与されているか確認してください。

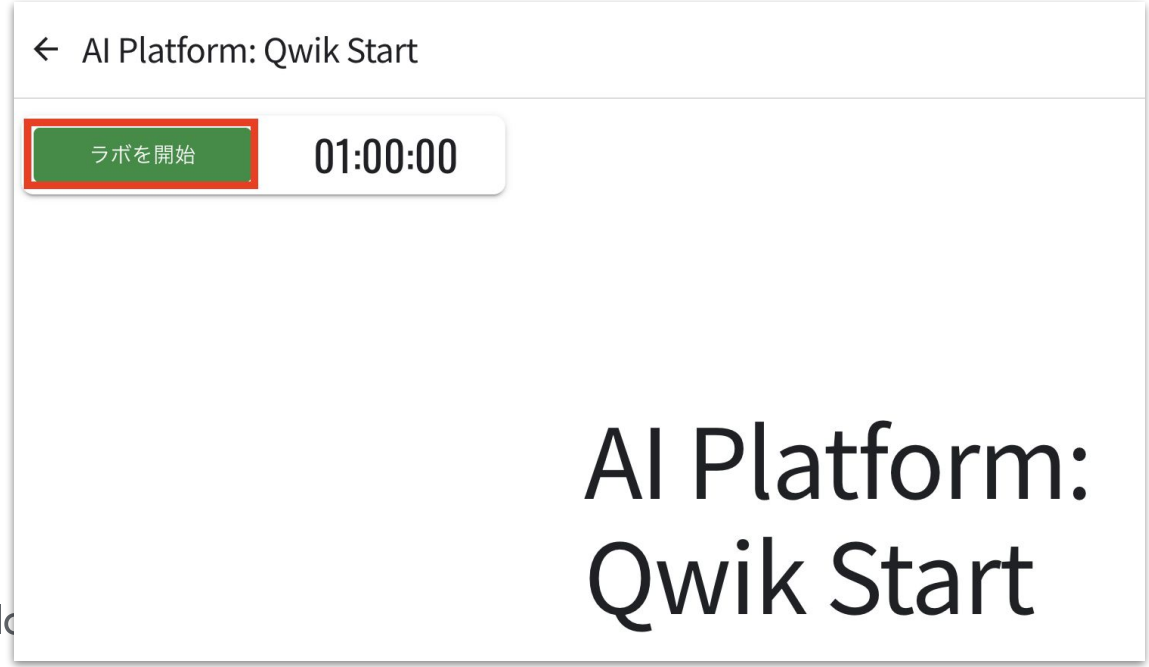


付与されない場合

- シークレットウィンドウでQwiklabs を開いていますか？
- クエストに登録はしていますか？
- 過去に登録したクエストを完了したことがありますか？
- サインインし直してみてもクレジット付与されませんか？

ラボの始め方

5. ラボの内容を読み確認したら、左上の [ラボを開始]をクリックします。



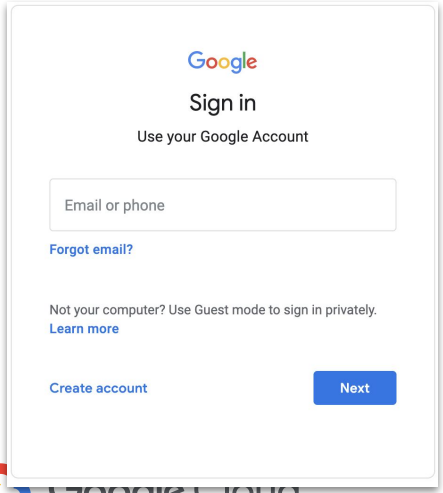
ラボの始め方

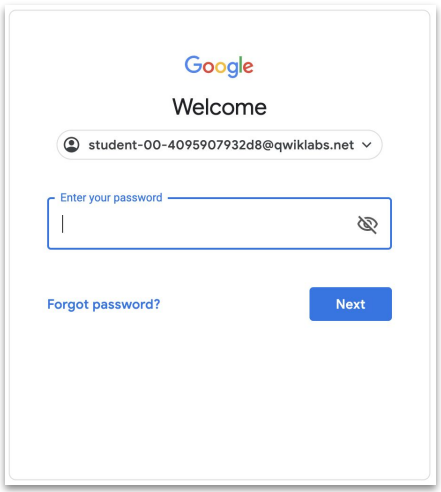
6. クレジットが必要なラボを実行する場合、「ラボを開始」をクリックすると、このような
   ポップアップが開きます。「一緒に開始：クレジット： 1」をクリックし、起動します。
7. ラボの手順に従って操作します。

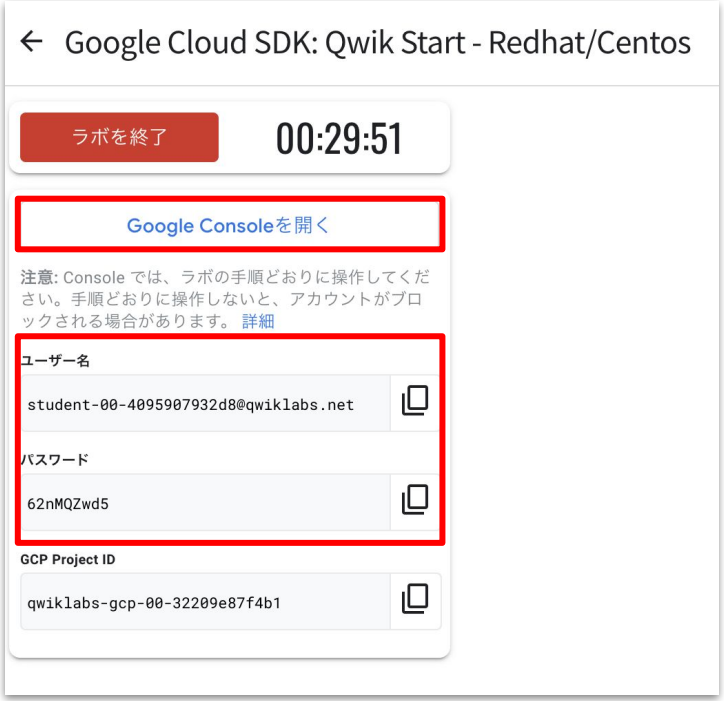
Google Cloud

ラボの始め方

8. 「Google Console を開く」をクリックする。
9. サインインの画面では、ご自身のアカウントではなく「Google Console を開く」の下に表示されている **ユーザ名、パスワード** でサインインしてください。

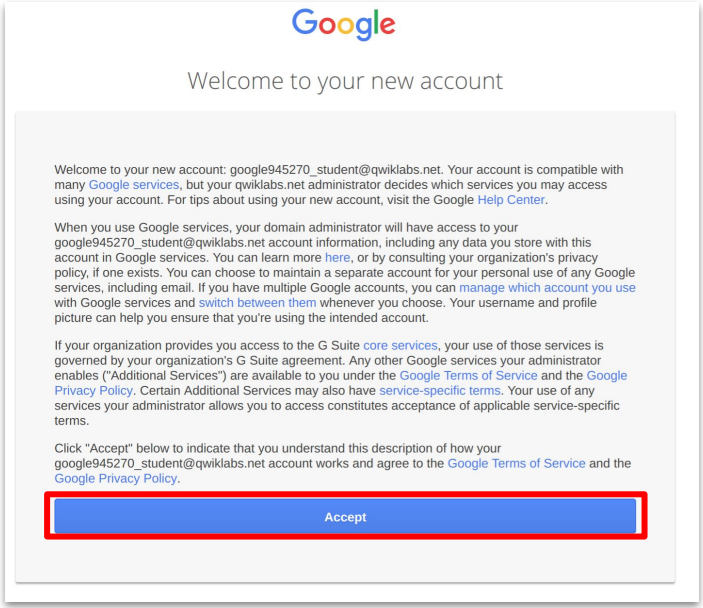






ラボの始め方

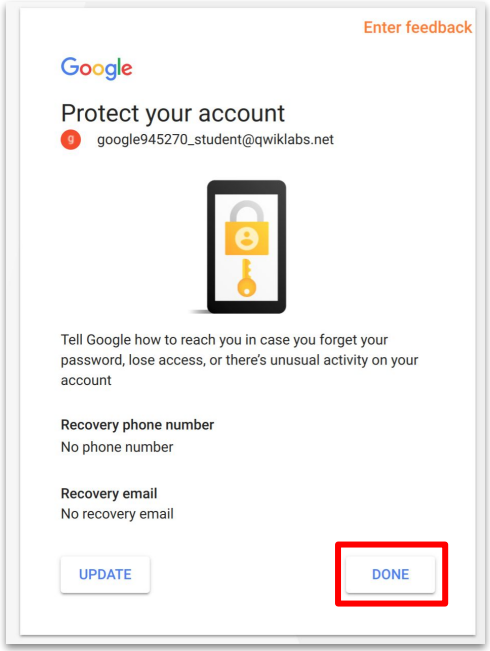
10. [利用規約] に同意します。(Accept ボタンを押してください)



Google Cloud

ラボの始め方

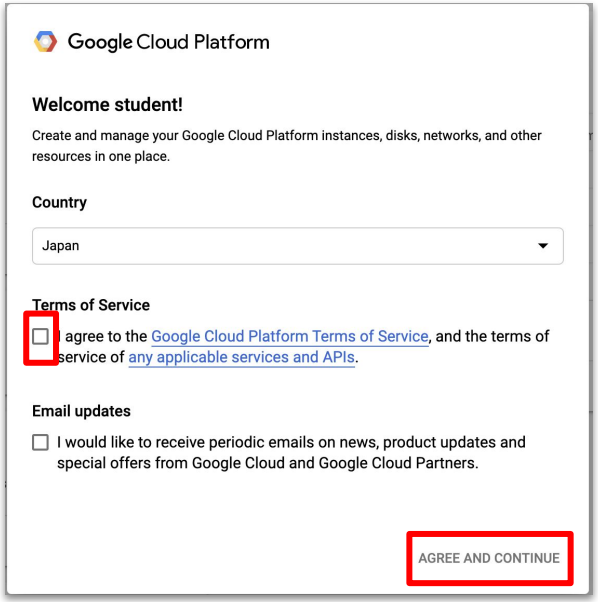
11. 復旧オプションは追加せずに、そのまま [Done] をクリックします。



Google Cloud

ラボの始め方

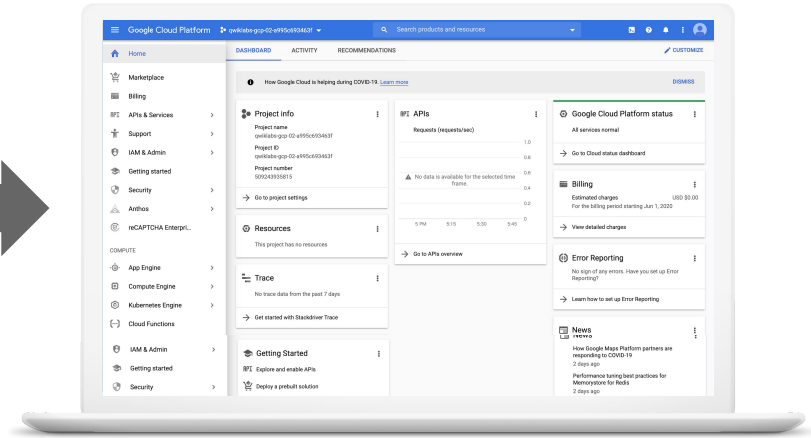
12. Terms of Service /Email Updates にチェックをお入れください。(Email Updatesは任意)
    [Accept] をクリックします。



ラボの進め方

Qwiklab のインストラクションとGCPコンソールを行き来しながら、ラボを先に進めてください

Qwiklabs

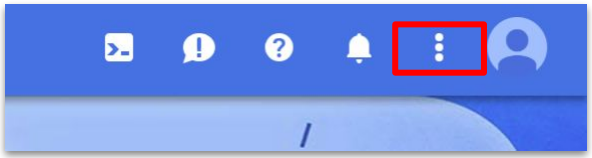


GCP Console

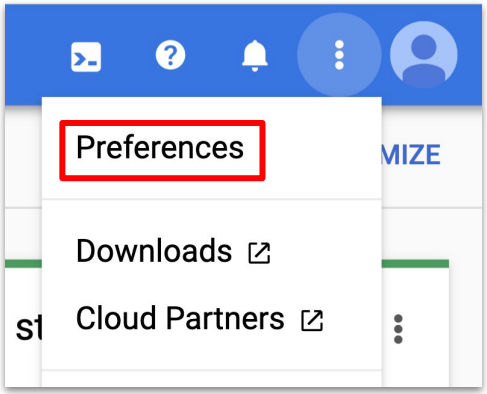
Google Cloud

ラボの進め方 (Console の言語変更)

1. Google Cloud Console で言語を変更したい場合、右上のドットをクリックします。



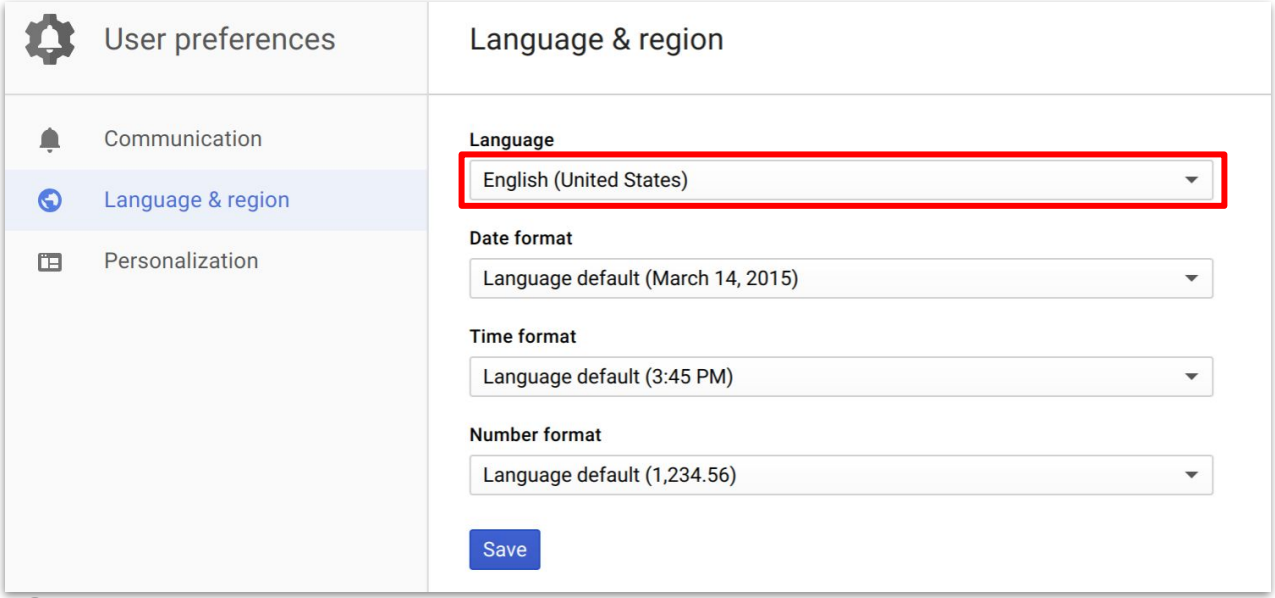
2. 一番上の[Preference]をクリックします。



Google Cloud

ラボの進め方 (Console の言語変更)

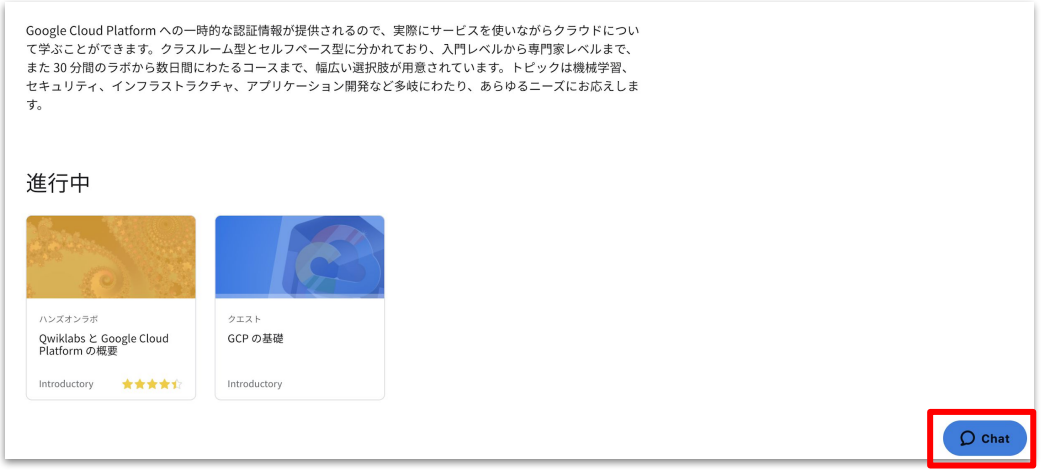
3. 左のメニューから [Languages & Region] を選択し、プルダウンで言語を選択します。



Google Cloud

ラボの進め方 (不明点がある際は)

操作など不明点がある場合は、Qwiklabs のチャットを利用して質問することが可能です。

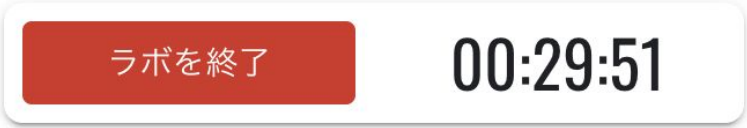


担当者がチャット対応致します。

Google Cloud

ラボの終了

全て終了したら、画面左上の [ラボを終了] をクリックします。



ラボが終了しました。お疲れ様でした。
（再度試したい際は、もう一度 [ラボを開始] ボタンをクリックしてください）

Google Cloud

Qwiklabs

クエスト Google Developer Essential （キャンペーンリンク）	goo.gle/appdev0618
アカウント作成	goo.gle/howtojoin
1 か月間無料適用の流れ (PDF)	goo.gle/free_guide

Google Cloud

モジュール 3
サーバーレス コンピューティング

Google Cloud

Cloud OnBoard

サーバーレスとは

運用モデル

インフラ管理が不要

マネージドセキュリティ

従量課金

プログラミングモデル

サービスベース

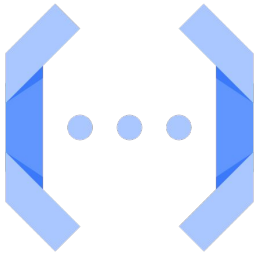
イベントドリブン

オープン

Google Cloud

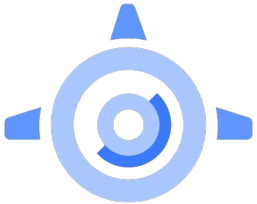
サーバーレス コンピューティングの選択肢

Cloud Functions



- **ソースコードベース**
- **イベントドリブン**
- 関数
- 最大 9 分の実行時間
- 1 concurrency / instance

Google App Engine



- **ソースコードベース**
- Web / API
- **ユーザー数・ナレッジの多さ**
- 1 プロジェクトにつき 1 リージョンの縛り

Cloud Run

- **コンテナベース**
- Web / API
- **ランタイム制約、ロックインなし**
- 最大 15 分 の実行時間
- 1 プロジェクトで複数リージョン利用可能

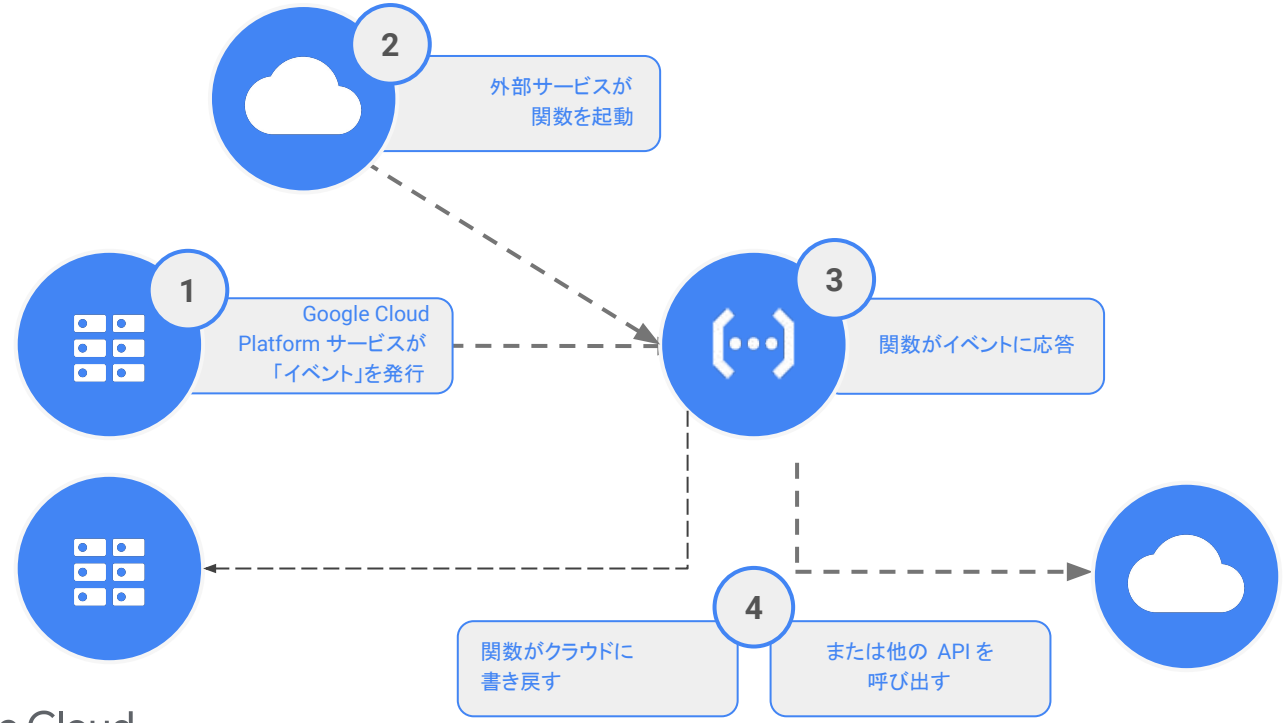
Google Cloud

GCP を使用したアプリケーション開発

イベントドリブン処理での Google Cloud Functions の使用

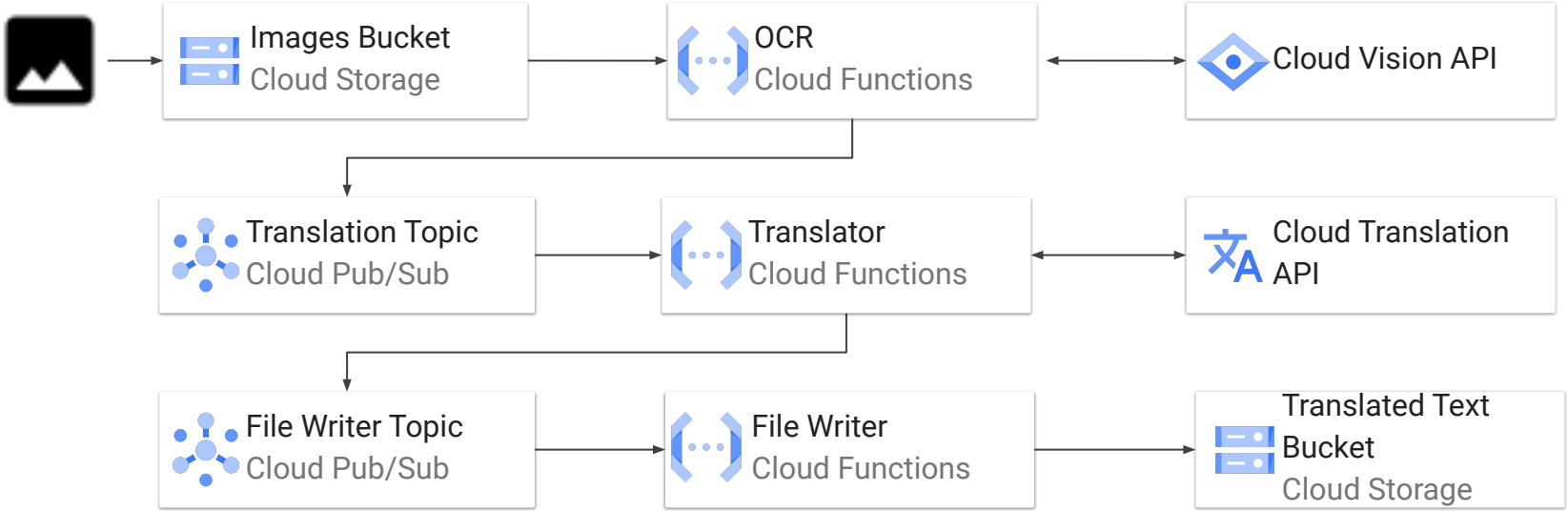
Google Cloud

概要



Google Cloud

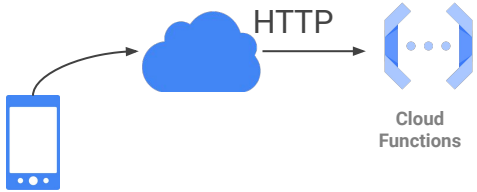
Cloud Functions が実現するイベントドリブン、サーバーレス、かつ極めてスケーラブルなマイクロサービス



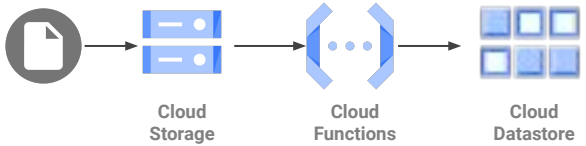
Google Cloud

ユース ケース

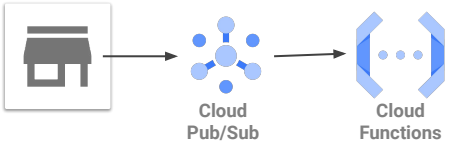
Webhooks



軽量 ETL

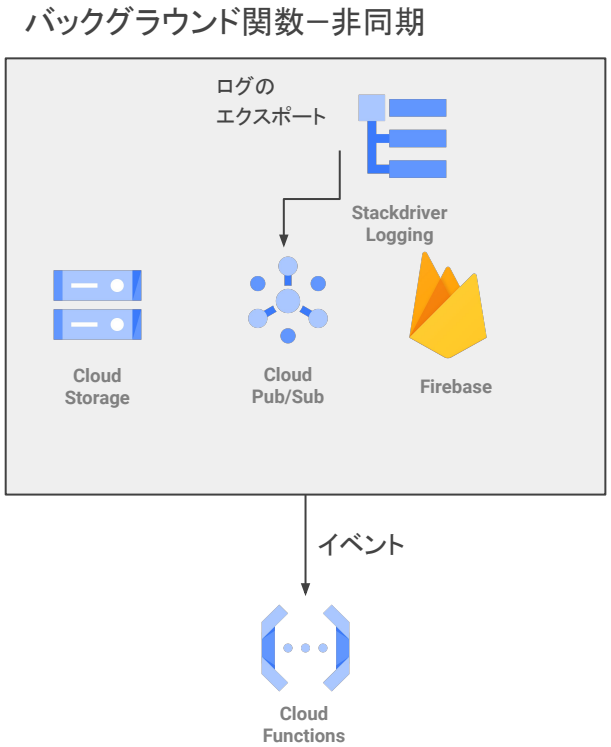


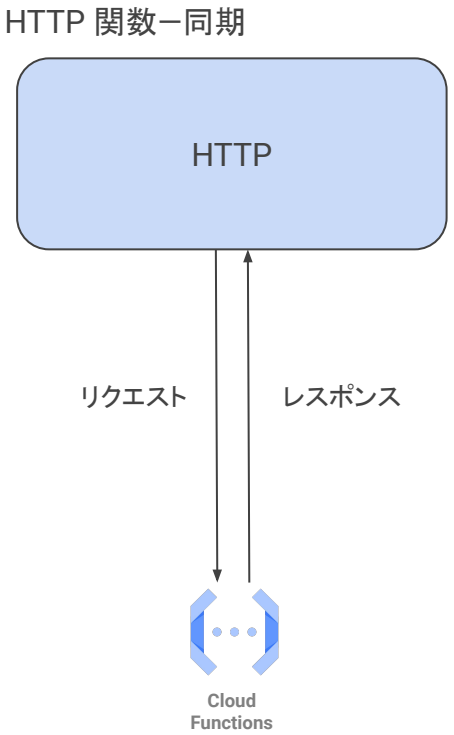
IoT



Google Cloud

Cloud Functions で可能な非同期トリガーと同期トリガー

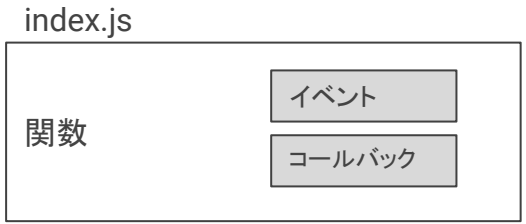




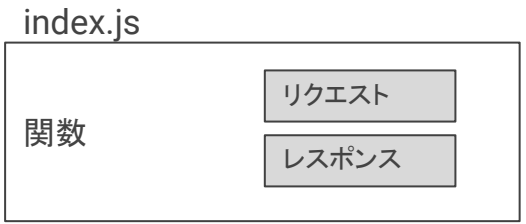
Google Cloud

Cloud Functions の記述

バックグラウンド関数



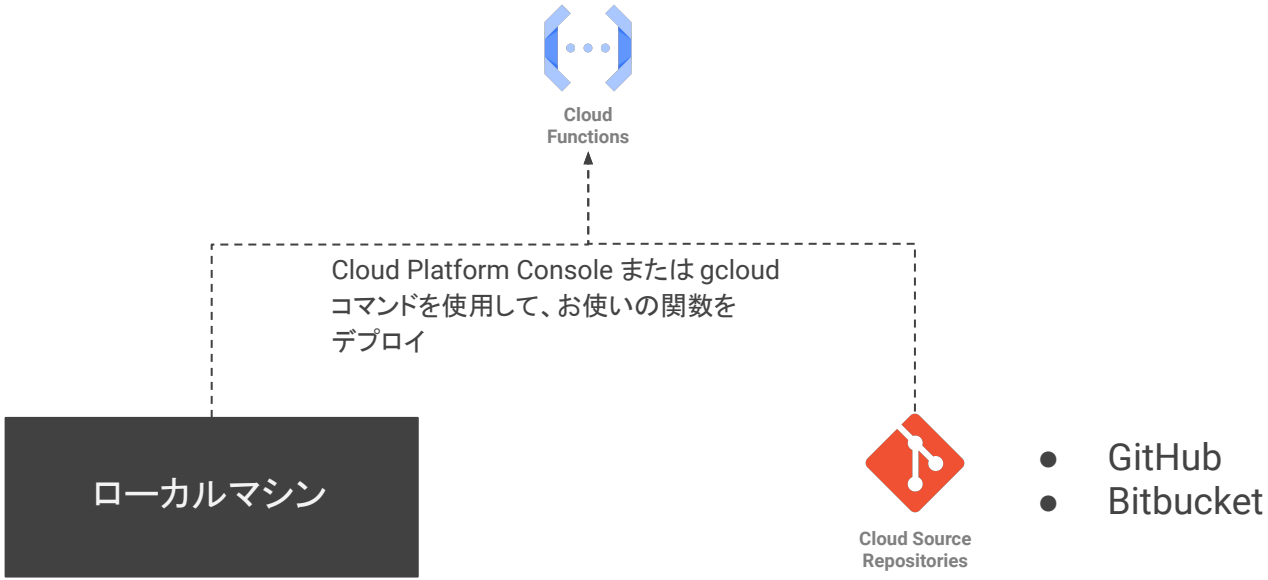
HTTP 関数



package.json ファイルの依存関係を指定

Google Cloud

Cloud Functions のデプロイ



Google Cloud

Cloud Functions がサポートするロギング、エラーレポート、モニタリング

INFO ログレベル: console.log(...)
ERROR ログレベル: console.error(...)
DEBUG ログレベル: 内部システムメッセージ

Stackdriver Logging

手動でスローまたはレポートされたエラー

Stackdriver Error Reporting

呼び出し数、実行時間、メモリ使用率

Cloud Functions

Google Cloud

GCP を使用したアプリケーション開発

App Engine

Google Cloud

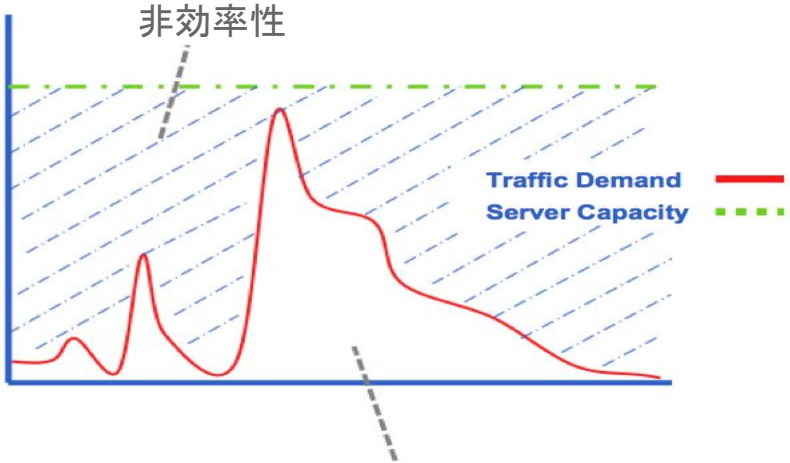
App Engine を使用した
スケーラブルなウェブアプリケーションのデプロイ

ネイティブなランタイム

Google Cloud

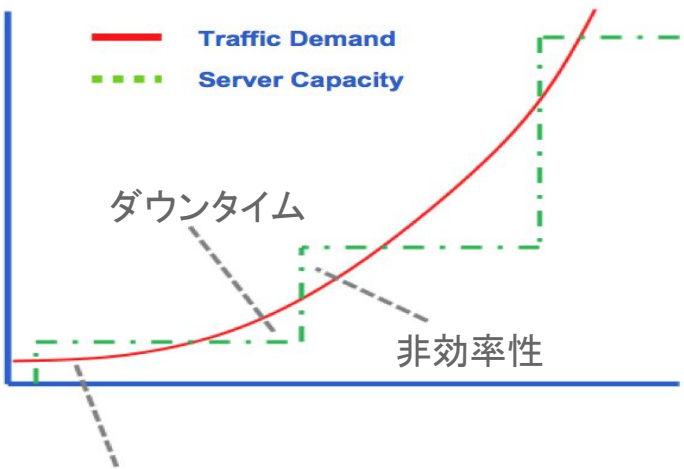
瞬時にスケールアウトするメリット

要求のスパイクが発生する場合



App Engine
使った分だけ支払いすれば良い

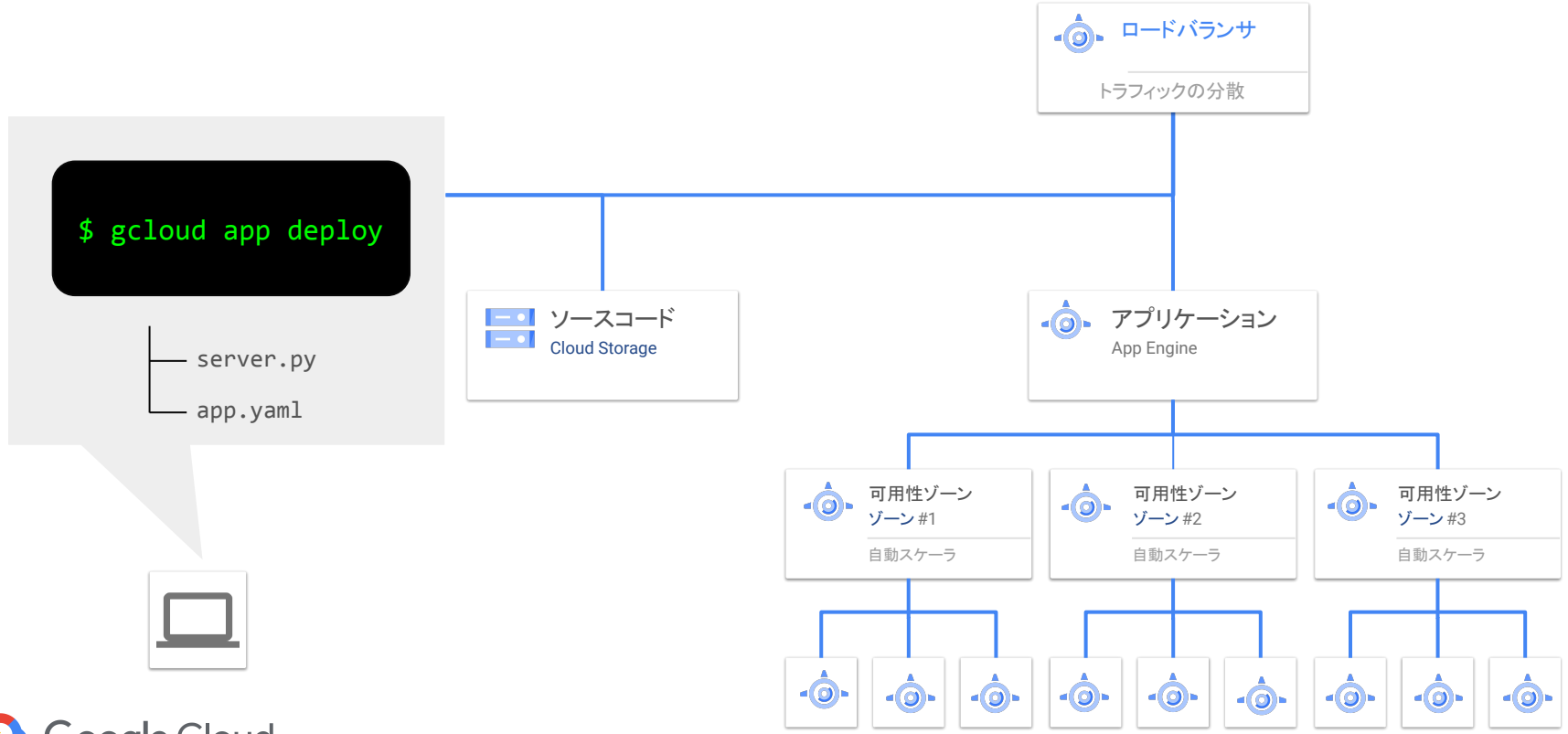
着実に要求が増える場合



App Engine
効率性と信頼性を伴いながらスケールする

Google Cloud

App Engine のデプロイ



Google Cloud

App Engine のデプロイ

コードをデプロイする際に、その裏では様々なセット アップも行っている


ロギング


自動スケーリング


負荷分散

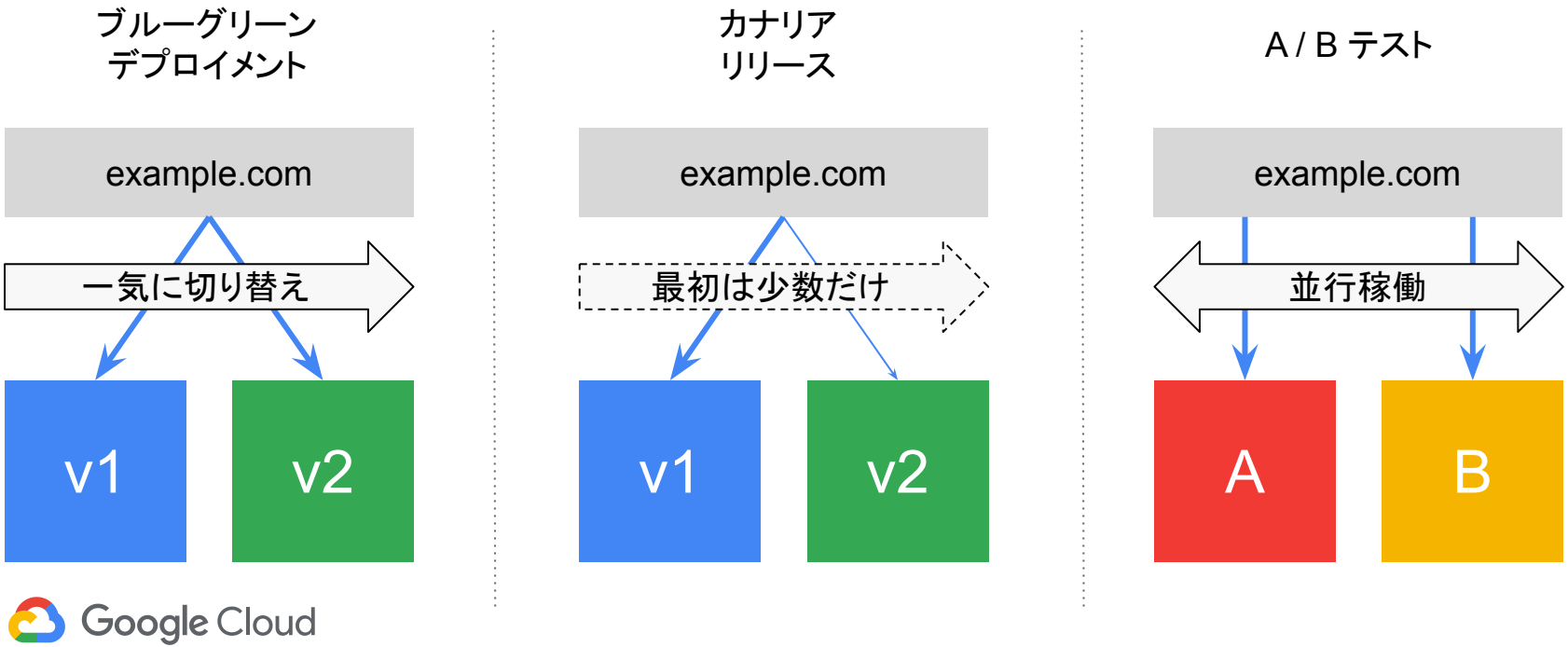

監視


ヘルスチェック


SSL とドメイン

Google Cloud

App Engine で実現できる柔軟なデプロイ



Google Cloud

GCP を使用したアプリケーション開発

Cloud Run

Google Cloud

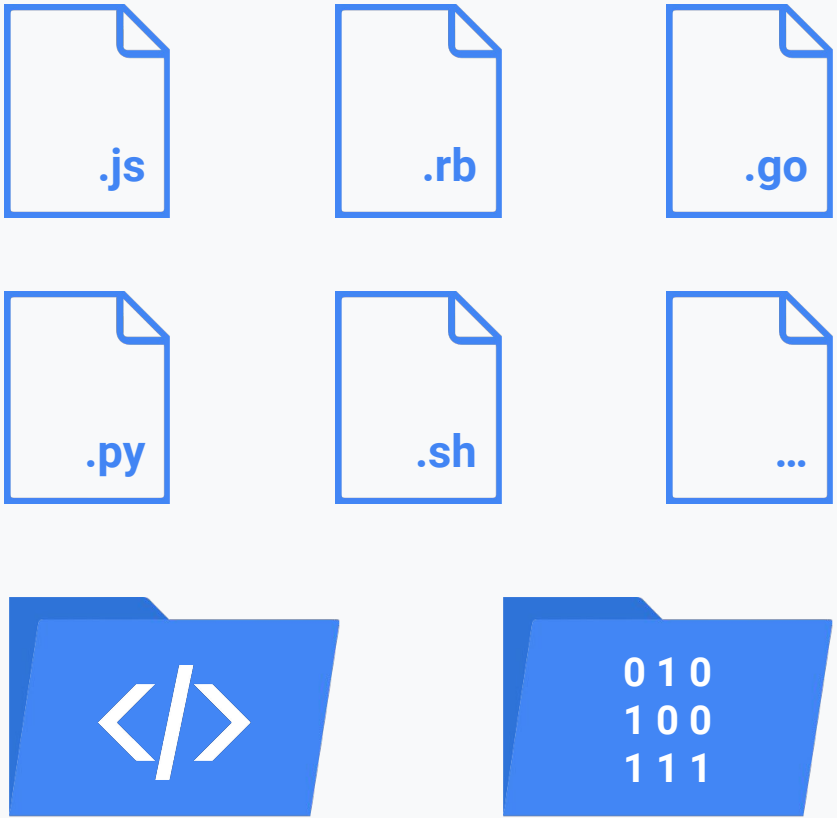
コンテナ: コードを分離しワークロードを管理する効率的な方法

ハイパーバイザベースの仮想化

コンテナベースの仮想化

コンテナ

- あらゆる言語
- あらゆるライブラリ
- あらゆるバイナリ
- ベース イメージのエコシステム



Google Cloud

コンテナを使用する理由

一貫性
開発環境、テスト環境、
本番環境にわたる一貫性

疎結合
アプリケーション層とオペレーティング
システム層間の疎結合

ワークロード移行
オンプレミスと
クラウド環境間の
シンプルなワークロード移行

俊敏性
俊敏な開発と運用

Google Cloud

Cloud Run

コンテナをサーバーレスに

Google Cloud

主な特徴

高速なデプロイ

ステートレスなコンテナ

高速に 0 to N スケール

数秒でデプロイし URL を付与

サーバーレス、ネイティブ

管理するサーバーはなし
コードに集中

言語やライブラリの制約なし

きっちり使った分だけお支払い

高いポータビリティ

どこでも同じ Developer Experience
フルマネージでも GKE のクラスタ上でも

Knative API の一貫性

ロックインの排除

Google Cloud

環境を選ばないサーバーレスコンテナ

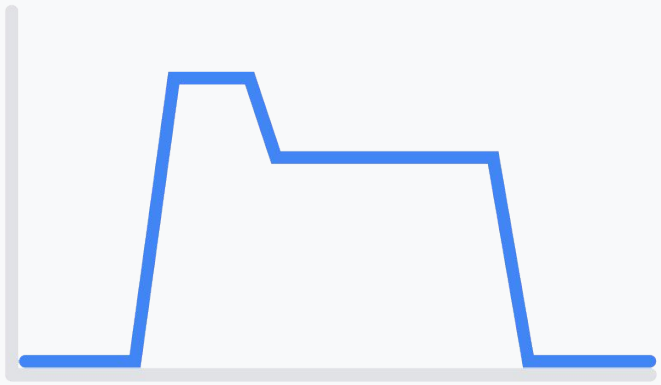
Cloud Run (フルマネージド)

- 完全なサーバーレス
- クラスタの管理は不要
- 使用量に応じた課金

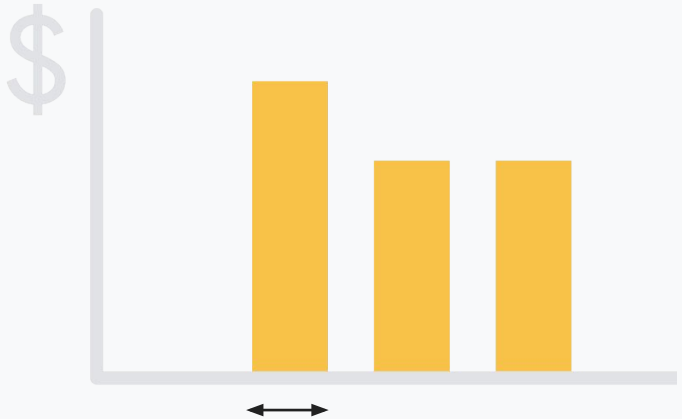
Cloud Run for Anthos

- サーバーレスのデベロッパー エクスペリエンス
- GKE クラスタまたはオンプレミスでの実行

Cloud Run : 従量課金



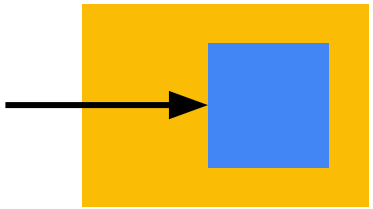
CPU/メモリ/リクエスト



100 ミリ秒

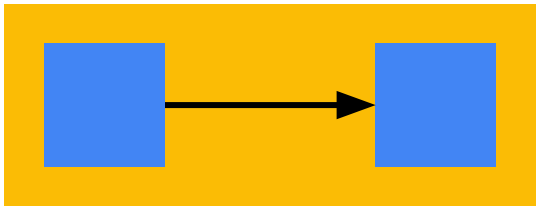
Google Cloud

ユース ケース



パブリック

- ウェブサイト
- API エンドポイント
- モバイル バックエンド
- Webhook



プライベート

- マイクロサービス
- 非同期タスク

Google Cloud

Google Cloud 認定資格

プロフェッショナル認定資格

- **Professional Cloud Architect**
  Google Cloud Platform でソリューションを **設計、構築、および管理する** 能力があることを証明します。
- **Professional Data Engineer**
  ●Google Cloud Platform で**データ処理システムを設計および構築し**、機械学習モデルを運用化する能力があることを証明します。
- **Professional Cloud Developer**
  ●Google Cloud Platform で**スケーラブルな高可用アプリケーションを構築し、デプロイする** 能力があることを証明します。
- **Professional Cloud Network Engineer**
  ●Google Cloud Platform で**ネットワークアーキテクチャを実装し、管理する** 能力があることを証明します。
- **Professional Cloud Security Engineer**
  ●GCP セキュリティテクノロジーを活用しながら **セキュアなインフラストラクチャを構成し、管理する** 能力があることを証明します。

アソシエイト認定資格

- **Associate Cloud Engineer**
  アソシエイトレベルの認定資格は、中核となる Google Cloud Platform テクノロジーを重視した、ジョブやタスクに基づく認定資格です。この認定資格は、クラウドや GCP に不慣れな方の出発点として最適であり、**プロフェッショナルレベルの認定資格への準備としてご利用いただけます** 。

Google Cloud

ありがとうございました

Google Cloud