



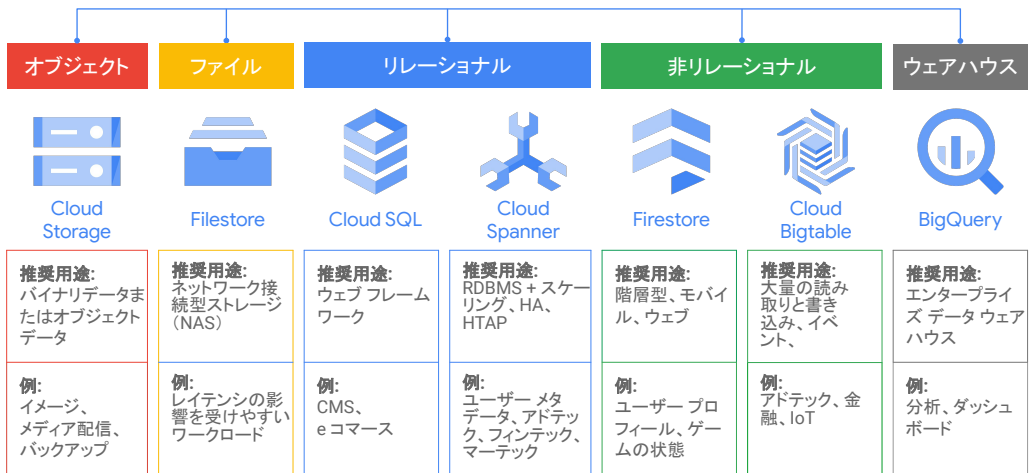
## ストレージとデータベース のサービス

このモジュールでは、Google Cloud のストレージとデータベースのサービスについて説明します。どのアプリケーションも、ビジネスデータ、ストリーミングされるメディア、デバイスのセンサーデータなどのデータを保存する必要があります。

アプリケーション側から見ると、テクノロジーがデータの保存と取得を行います。重要なのは、そのサービスがデータベースなのか、それともオブジェクト ストアなのかよりも、データの特徴に基づいて効率的に保存および取得し、アプリケーションの要件を満たせるかどうかです。

Google はいくつかのデータ ストレージ サービスを選択肢として用意しています。このモジュールでは、Cloud Storage、Filestore、Cloud SQL、Cloud Spanner、Cloud Firestore、Cloud Bigtable について詳しく説明します。まずは、これらのサービスの概要を説明します。

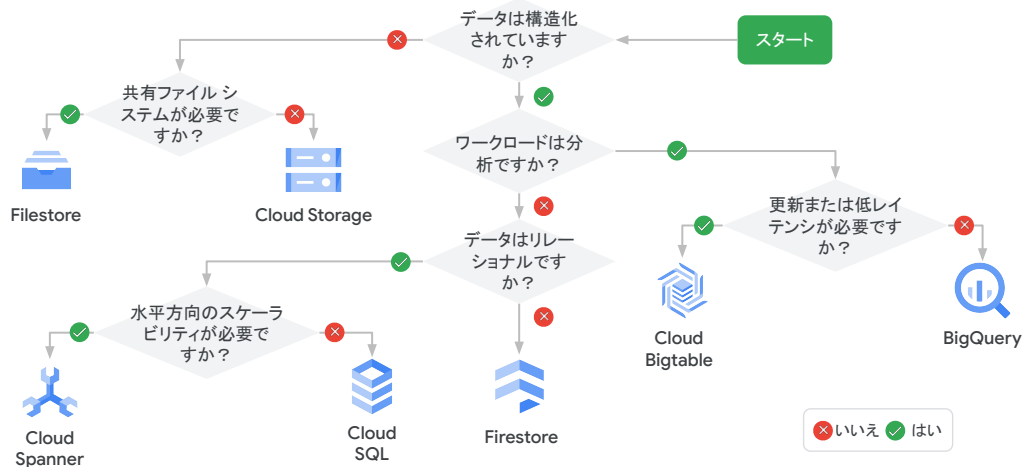
## ストレージとデータベースのサービス



この表には、ストレージとデータベースのサービスが示されています。ストレージ サービスタイプ(オブジェクト、ファイル、リレーショナル、非リレーショナル、データ ウェアハウス)、それぞれのサービスの推奨用途、使用目的も記載されています。

BigQuery も右側にあります。ここでこのサービスに触れたのは、データ ストレージとデータ処理の境目に位置するサービスだからです。BigQuery にデータを保存することはできませんが、BigQuery の本来の用途はビッグデータ分析とインタラクティブなクエリの実行です。したがって、BigQuery についてはこのコースの後半で説明します。

## ストレージとデータベースに関する意思決定チャート



表ではわかりにくいという方のためにディシジョン ツリーも用意しました。質問に答えていくだけで、アプリケーションに最適なソリューションがわかります。

詳しく見ていきましょう。

- まずは、データが構造化されているかを確認します。構造化されていない場合は、共有ファイル システムが必要かどうかを考えます。必要であれば、Filestore を選びます。
- 必要でなければ、Cloud Storage を選びます。
- データが構造化されている場合は、ワークロードの中心は分析かどうかを確認します。そうであれば、レイテンシと更新のニーズに応じて、Cloud Bigtable または BigQuery を選ぶ必要があるでしょう。
- そうでない場合は、データがリレーショナルかどうかを確認します。リレーショナルでない場合は、Firestore を選びます。
- リレーショナルである場合は、水平方向のスケーラビリティが必要かどうかを基準に Cloud SQL または Cloud Spanner を選ぶ必要があるでしょう。

アプリケーションで行う処理に応じて、これらのサービスから 1 つを選ぶ場合もあれば、複数を選ぶ場合もあります。これらのサービスから適切なものを選ぶ方法の詳細については、この動画のリンク セクションをご覧ください。

<https://cloud.google.com/storage-options/>,  
<https://cloud.google.com/products/databases/>

## 範囲

### インフラストラクチャトラック

- サービスの差別化要因
- 各サービスの利用を検討すべきタイミング
- サービスの設定と接続

### データ エンジニアリングトラック

- データベース システムの使い方
- アプリケーションの設計、組織、構造、スキーマ、使用
- サービスでの構造化データの保存 / 取得方法の詳細

個々のデータ ストレージ サービスについて詳しく確認する前に、このモジュールの範囲を明確にしておきましょう。

このモジュールの目的は、インフラストラクチャの観点から、利用できるサービスと、利用を検討すべきタイミングを学習することです。データベース システムの使い方を詳しく知らなくても、サービスの設定と接続を行えるようにすることが目的です。

各サービスの設計、組織、構造、スキーマの詳細や、データを適切に最適化、提供、保存する方法を詳しく学ぶ必要がある場合は、Google Cloud のデータ エンジニアリング コースの受講をおすすめします。

## アジェンダ

### Cloud Storage と Filestore

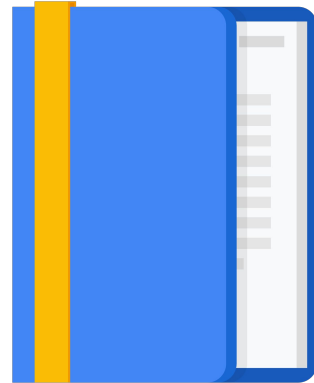
Cloud SQL

Cloud Spanner

Firestore

Cloud Bigtable

Memorystore



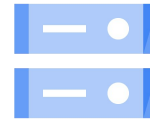
アジェンダをご覧ください。このモジュールでは、先ほど触れたすべてのサービスについて説明します。慣れていただくために、2つのラボで実際にこれらのサービスを使っていただきます。

Google Cloud のフルマネージド Redis サービスである Memorystore の概要も説明します。

まずは Cloud Storage と Filestore について詳しく見ていきましょう。

## Cloud Storage - オブジェクト ストレージ サービス

- ウェブサイトのコンテンツ
- アーカイブと障害復旧のためのデータ保存
- 直接ダウンロードによるユーザーへの大規模なデータ オブジェクトの配布

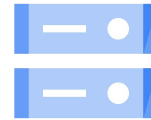


Cloud  
Storage

Cloud Storage は Google Cloud のオブジェクト ストレージ サービスで、世界中のどこからでも、いつでもデータを保存、取得できます。データの量に制限はありません。ウェブサイト コンテンツの提供、アーカイブと障害復旧のためのデータの保存、直接ダウンロードによるユーザーへの大規模なデータ オブジェクトの配布など、さまざまなシナリオで Cloud Storage を使用できます。

## Cloud Storage の主な機能

- エクサバイト規模まで拡張できる  
スケーラビリティ
- ミリ秒単位の時間で最初のバイト  
を転送
- あらゆるストレージ クラスで  
高可用性を実現
- 単一の API ですべてのストレージ  
クラスに対応



Cloud  
Storage

Cloud Storage の主な機能として、次のものがあります。

- エクサバイト規模のデータまで拡張できるスケーラビリティ
- 最初のバイトの転送時間がミリ秒単位
- あらゆるストレージ クラスで高可用性を実現
- 単一の API ですべてのストレージ クラスに対応

Cloud Storage をファイル システム内のファイルと考える人もいますが、実際にはファイル システムではありません。Cloud Storage は、オブジェクトを収めるバケットの集まりです。ディレクトリを作成できますが、ディレクトリは実際には、バケット内の別のオブジェクトを参照するオブジェクトです。これらのファイルすべてにインデックスを作成することは、ファイル システムのように簡単にはできません。オブジェクトにアクセスする固有の URL があるだけです。

## ストレージ クラスの概要

	Standard	Nearline	Coldline	Archive
ユースケース	「ホット」データや、短時間だけ保存されるデータ（処理で大量のデータを使用する場合など）	アクセス頻度の低いデータ（データのバックアップ、ロングテールのマルチメディアコンテンツ、データアーカイブなど）	四半期に1回程度しか読み取りや変更を行わない、アクセス頻度の低いデータ	データアーカイブ、オンラインバックアップ、障害復旧
最小保存期間	なし	30日	90 日	365 日
取得費用	なし	\$0.01/GB	\$0.02/GB	\$0.05/GB
可用性 SLA	99.95%（デュアル / マルチ） 99.90%（リージョン）	99.90%（デュアル / マルチ） 99.00%（リージョン）		なし
耐久性	99.999999999%			

Cloud Storage には、Standard、Nearline、Coldline、Archive の 4 つのクラスがあり、それぞれに 3 つのロケーション タイプがあります。

- マルチリージョンは、米国などの、2 つ以上の地理的な場所を含む広い地理的なエリアとなります。
- デュアルリージョンは、フィンランドやオランダなどの、リージョンの特定のペアです。
- リージョンは、ロンドンなどの特定の地理的な場所です。



## ストレージ クラスの概要

	Standard	Nearline	Coldline	Archive
ユースケース	「ホット」データや、短時間だけ保存されるデータ（処理で大量のデータを使用する場合など）	アクセス頻度の低いデータ（データのバックアップ、ロングテールのマルチメディアコンテンツ、データアーカイブなど）	四半期に1回程度しか読み取りや変更を行わない、アクセス頻度の低いデータ	データアーカイブ、オンラインバックアップ、障害復旧
最小保存期間	なし	30日	90 日	365 日
取得費用	なし	\$0.01/GB	\$0.02/GB	\$0.05/GB
可用性 SLA	99.95%(デュアル / マルチ) 99.90%(リージョン)	99.90%(デュアル / マルチ) 99.00%(リージョン)		なし
耐久性	99.999999999%			

マルチリージョンまたはデュアルリージョンに保存されるオブジェクトは、地理的に冗長になります。次に、各ストレージ クラスについて説明します。

**Standard Storage** は、頻繁にアクセスされるデータ（「ホット」データ）や、短時間だけ保存されるデータに最適です。最も高価なストレージ クラスですが、最小保存期間はなく、取得費用もかかりません。

リージョンで使用する場合、Standard Storage は、データを使用する Google Kubernetes Engine クラスタまたは Compute Engine インスタンスと同じ場所にデータを保存するのに適しています。リソースを共通の場所に配置すると、大量のデータを使用する処理のパフォーマンスが最大限に高まり、ネットワーク料金を削減できます。

デュアルリージョンで使用する場合、リージョンで使用する場合と同様に、関連付けられたいずれかのリージョンに配置されている Google Cloud プロダクトにアクセスする際にパフォーマンスが最適化されるのに加えて、データが地理的に離れた場所に保存されるために可用性も向上します。

マルチリージョンで使用する場合、Standard Storage はウェブサイトのコンテンツの配信、動画のストリーミング、インタラクティブなワークロードの実行、モバイル アプリケーションやゲーム アプリケーションをサポートするデータの提供に適しています。

## ストレージ クラスの概要

	Standard	Nearline	Coldline	Archive
ユースケース	「ホット」データや、短時間だけ保存されるデータ（処理で大量のデータを使用する場合など）	アクセス頻度の低いデータ（データのバックアップ、ロングテールのマルチメディアコンテンツ、データアーカイブなど）	四半期に1回程度しか読み取りや変更を行わない、アクセス頻度の低いデータ	データアーカイブ、オンラインバックアップ、障害復旧
最小保存期間	なし	30日	90 日	365 日
取得費用	なし	\$0.01/GB	\$0.02/GB	\$0.05/GB
可用性 SLA	99.95%(デュアル / マルチ) 99.90%(リージョン)	99.90%(デュアル / マルチ) 99.00%(リージョン)		なし
耐久性	99.999999999%			

**Nearline Storage** は、アクセス頻度の低いデータの保存に適した低コストで耐久性の高いストレージ サービスであり、データのバックアップ、ロングテールのマルチメディア コンテンツ、データ アーカイブなどに利用できます。可用性が若干低く、最小保存期間が 30 日で、データアクセスで費用が発生しても、保存時のストレージ コストを抑えたい場合には、Standard Storage よりも Nearline Storage のほうが最適な選択肢となります。

## ストレージ クラスの概要

	Standard	Nearline	Coldline	Archive
ユースケース	「ホット」データや、短時間だけ保存されるデータ（処理で大量のデータを使用する場合など）	アクセス頻度の低いデータ（データのバックアップ、ロングテールのマルチメディアコンテンツ、データアーカイブなど）	四半期に1回程度しか読み取りや変更を行わない、アクセス頻度の低いデータ	データアーカイブ、オンラインバックアップ、障害復旧
最小保存期間	なし	30日	90 日	365 日
取得費用	なし	\$0.01/GB	\$0.02/GB	\$0.05/GB
可用性 SLA	99.95%(デュアル / マルチ) 99.90%(リージョン)	99.90%(デュアル / マルチ) 99.00%(リージョン)		なし
耐久性	99.999999999%			

**Coldline Storage** は、アクセス頻度の低いデータの保存に適した、極めて低コストで、耐久性の高いストレージ サービスです。可用性が低めで、最小保存期間が 90 日で、データアクセスの費用が高くて、保存時のストレージ コストを抑えたい場合には、Standard Storage や Nearline Storage よりも Coldline Storage のほうが最適な選択肢となります。

## ストレージ クラスの概要

	Standard	Nearline	Coldline	Archive
ユースケース	「ホット」データや、短時間だけ保存されるデータ（処理で大量のデータを使用する場合など）	アクセス頻度の低いデータ（データのバックアップ、ロングテールのマルチメディアコンテンツ、データアーカイブなど）	四半期に 1 回程度しか読み取りや変更を行わない、アクセス頻度の低いデータ	データアーカイブ、オンラインバックアップ、障害復旧
最小保存期間	なし	30 日	90 日	365 日
取得費用	なし	\$0.01/GB	\$0.02/GB	\$0.05/GB
可用性 SLA	99.95%（デュアル / マルチ） 99.90%（リージョン）	99.90%（デュアル / マルチ） 99.00%（リージョン）		なし
耐久性	99.999999999%			

**Archive Storage** は、データ アーカイブ、オンライン バックアップ、障害復旧のための最も低コストで、耐久性に優れたストレージ サービスです。他のクラウド プロバイダによって提供されている「最もコールドな」ストレージ サービスと異なり、データに迅速にアクセスできます。必要なデータの取得に数時間あるいは数日かかることはありません。他の Cloud Storage ストレージ クラスとは異なり、Archive Storage には可用性 SLA はありませんが、一般的な可用性は Nearline Storage や Coldline Storage と同等です。また、Archive Storage はデータのアクセスと操作に関する費用が高く、最小保存期間も 365 日間に及びます。Archive Storage は、1 年間に 1 回未満しかアクセスしないデータに最適です。

## ストレージ クラスの概要

	Standard	Nearline	Coldline	Archive
ユースケース	「ホット」データや、短時間だけ保存されるデータ（処理で大量のデータを使用する場合など）	アクセス頻度の低いデータ（データのバックアップ、ロングテールのマルチメディアコンテンツ、データアーカイブなど）	四半期に1回程度しか読み取りや変更を行わない、アクセス頻度の低いデータ	データアーカイブ、オンラインバックアップ、障害復旧
最小保存期間	なし	30日	90 日	365 日
取得費用	なし	\$0.01/GB	\$0.02/GB	\$0.05/GB
可用性 SLA	99.95%(デュアル / マルチ) 99.90%(リージョン)	99.90%(デュアル / マルチ) 99.00%(リージョン)		なし
耐久性	99.999999999%			

耐久性と可用性に注目してみましょう。すべてのストレージ クラスはイレブンナインの耐久性を備えていますが、これは何を意味するのでしょうか。いつでもファイルにアクセスできるということでしょうか。違います。これは、データが失われないということです。データにアクセスできないこともあります。それは銀行に行って、ここに自分のお金があると言うようなものです。これがイレブンナインの耐久性です。しかし、銀行の営業時間でなければ利用できません。これが可用性であり、ストレージ クラスとロケーション タイプによって違いがあります。

## Cloud Storage の概要

### バケット

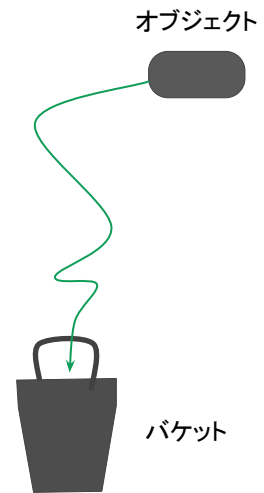
- 命名の要件
- ネストできない

### オブジェクト

- 作成時にバケットのストレージ クラスを継承
- 最小サイズなし、無制限ストレージ

### アクセス

- gsutil コマンド
- (RESTful)JSON API または XML API



Cloud Storage は、ここに示す 2 つの異なるアイテムに分かれています。

# Cloud Storage の概要

## バケット

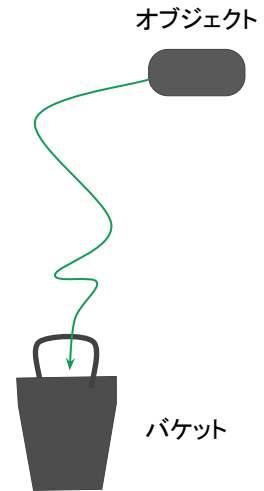
- 命名の要件
- ネストできない

## オブジェクト

- 作成時にバケットのストレージ クラスを継承
- 最小サイズなし、無制限ストレージ

## アクセス

- gsutil コマンド
- (RESTful)JSON API または XML API



まずバケットがあります。バケットにはグローバルに一意の名前を付ける必要があります。また、バケットをネストすることはできません。

# Cloud Storage の概要

## バケット

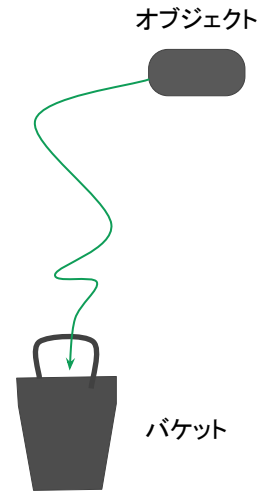
- 命名の要件
- ネストできない

## オブジェクト

- 作成時にバケットのストレージ クラスを継承
- 最小サイズなし、無制限ストレージ

## アクセス

- gsutil コマンド
- (RESTful)JSON API または XML API



バケットに収めるデータをオブジェクトと呼びます。オブジェクトはバケットのストレージ クラスを継承します。オブジェクトはテキスト ファイル、ドキュメント ファイル、動画ファイルなどであり、これらにサイズの制限はなく、割り当て内であればいくらでも大きくすることができます。



## Cloud Storage の概要

### バケット

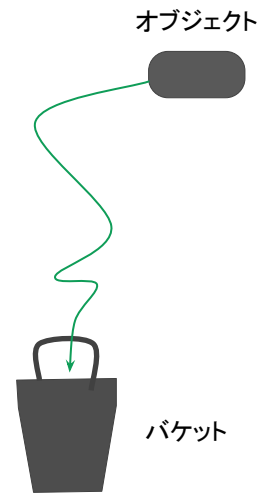
- 命名の要件
- ネストできない

### オブジェクト

- 作成時にバケットのストレージ クラスを継承
- 最小サイズなし、無制限ストレージ

### アクセス

- gsutil コマンド
- (RESTful)JSON API または XML API



データにアクセスするには、gsutil コマンドか、JSON API または XML API を使います。

## デフォルトのストレージ クラスの変更

- 新しいオブジェクトにはデフォルトのクラスが適用される
- リージョン バケットはマルチリージョンまたはデュアルリージョンに変更できない
- マルチリージョン バケットはリージョンに変更できない
- オブジェクトは別のバケットに移動できる
- オブジェクトのライフサイクル管理でオブジェクトのクラスを管理できる

オブジェクトをバケットにアップロードすると、特にストレージ クラスを指定しない限り、そのオブジェクトにはバケットのストレージ クラスが割り当てられます。バケットのデフォルトのストレージ クラスは変更できますが、ロケーション タイプをリージョンからマルチリージョンまたはデュアルリージョン(またはその逆)に変更することはできません。

バケットにあるオブジェクトのストレージ クラスも変更できます。オブジェクトを別のバケットに移動したり、URL を変更したりする必要はありません。オブジェクトごとにストレージ クラスを設定すると、バケット内に保存しているオブジェクトの中にアクセス頻度が低いオブジェクトがある場合などに便利です。そのような場合、それらのオブジェクトのストレージ クラスを Nearline Storage から Coldline Storage または Archive Storage に変更することで、コストを最小限に抑えることができます。

バケット内のオブジェクトのクラスを簡単に管理できるように、Cloud Storage にはオブジェクトのライフサイクル管理機能があります。詳しくは後ほど説明します。

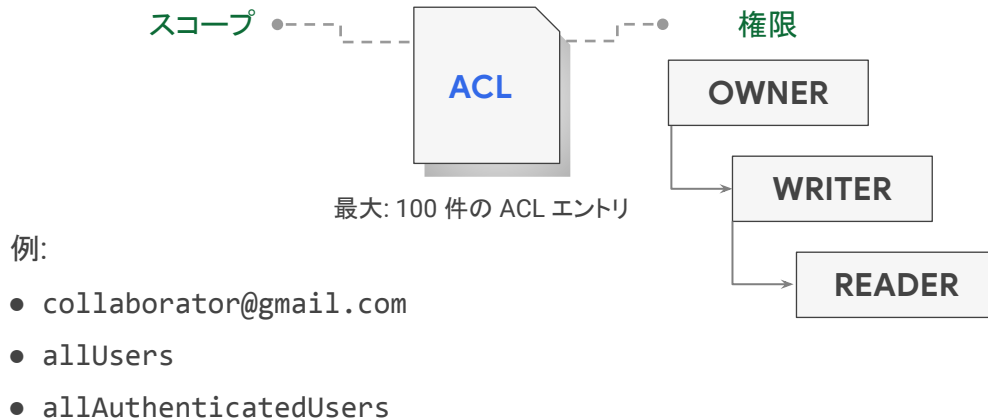
## アクセス制御



プロジェクトの一部であるオブジェクトとバケットのアクセス制御について確認しましょう。

- プロジェクトで IAM を使って、バケットの表示、バケットにあるオブジェクトの一覧表示、バケットにあるオブジェクトの名前の表示、新しいバケットの作成をどのユーザーまたはサービス アカウントに許可するかを制御できます。ほとんどの用途に IAM で十分に対応できます。ロールはプロジェクトからバケット、オブジェクトに継承されます。
- さらに細かく制御するにはアクセス制御リスト (ACL) を使います。
- それ以上に詳細な制御が必要であれば、署名付き URL の暗号鍵を使うと、バケットまたはオブジェクトへのアクセスを時間制限付きで許可できます。
- また、署名付きポリシードキュメントは、署名付き URL を使ってアップロードできるファイルの種類を限定することで、さらに細かい制御を提供します。ACL と署名付き URL について詳しく見ていきましょう。

## アクセス制御リスト (ACL)



ACL とは、バケットとオブジェクトへのアクセス権を持つユーザーと各ユーザーのアクセスのレベルを定義するために使用できるメカニズムです。1 つのバケットまたはオブジェクトについて作成できる ACL エントリは最大 100 件です。

各 ACL は 1 つ以上のエントリからなり、各エントリには次の 2 つの情報が含まれます。

- スコープ: 指定した操作を誰が行うことができるかを定義する(たとえば、特定のユーザーやユーザー グループなど)。
- 権限: どの操作を行うことができるかを定義する(たとえば、読み取りや書き込み)。

このスライドに示した allUsers ID は、Google アカウントを持つかどうかに関係なく、インターネット上のすべてのユーザーを表します。一方で、allAuthenticatedUsers ID は、Google アカウントで認証済みのすべてのユーザーを表します。

ACL の詳細については、この動画のリンク セクションをご覧ください。

[<https://cloud.google.com/storage/docs/access-control/lists>]

## 署名付き URL

- チケットを介したバケットとオブジェクトへの「バレットキー」アクセス:
  - チケットとは暗号署名付きの URL
  - 時間制限付き
  - チケットに指定される操作: HTTP GET、PUT、DELETE (POST はない)
  - URL を知るユーザーは誰でも、許可された操作を呼び出し可能
- アカウントの秘密鍵と gsutil を使用する例:

```
gsutil signurl -d 10m path/to/privatekey.p12  
gs://bucket/object
```

アプリケーションによっては、アカウントベースの認証でリソースへのアクセスを制御するよりも、すべてのユーザーが使える時間制限付きアクセス トークンを付与するほうが簡単で効率的なことがあります (たとえば、Google アカウントの取得をユーザーに強制したくない場合)。

署名付き URL を使うと、これを Cloud Storage で行うことができます。特定の Cloud Storage リソースへの読み取りまたは書き込みのアクセスを許可する、アクセス期限付きの URL を作成します。この URL には、サービス アカウントに関連付けられた秘密鍵で署名が付けられます。Cloud Storage がリクエストを受け取ると、そのアクセス付与 URL が信頼できるセキュリティ プリンシパル (ここではサービス アカウント) の代理として発行されていることが確認されて、その URL を知るユーザーにそのアカウントへの信頼が委任されます。

署名付き URL は、いったん提供すると制御ができなくなるので、相応の時間が経過したら期限切れにする必要があります。

## Cloud Storage の機能

- 顧客指定の暗号鍵 (CSEK)
  - Google が管理する鍵ではなく独自の鍵を使う
- オブジェクトのライフサイクル管理
  - オブジェクトを自動で削除またはアーカイブする
- オブジェクトのバージョンング
  - オブジェクトの複数のバージョンを管理する
- ディレクトリ同期
  - VM ディレクトリをバケットと同期させる
- Pub/Sub を使用したオブジェクト変更通知

Cloud Storage では他にも使える機能があります。詳しい説明は後で行うので、ここでは概要のみを説明します。

## Cloud Storage の機能

- 顧客指定の暗号鍵 (CSEK)
  - Google が管理する鍵ではなく独自の鍵を使う
- オブジェクトのライフサイクル管理
  - オブジェクトを自動で削除またはアーカイブする
- オブジェクトのバージョンング
  - オブジェクトの複数のバージョンを管理する
- ディレクトリ同期
  - VM ディレクトリをバケットと同期させる
- Pub/Sub を使用したオブジェクト変更通知

顧客指定の暗号鍵については、永続ディスクを仮想マシンにアタッチするときの使い方についてこのコースですでに簡単に触れました。顧客指定の暗号鍵を使うと、Google が管理する鍵ではなく独自の暗号鍵を指定できます。この機能を Cloud Storage でも使用できます。

## Cloud Storage の機能

- 顧客指定の暗号鍵 (CSEK)
  - Google が管理する鍵ではなく独自の鍵を使う
- オブジェクトのライフサイクル管理
  - オブジェクトを自動で削除またはアーカイブする
- オブジェクトのバージョンニング
  - オブジェクトの複数のバージョンを管理する
- ディレクトリ同期
  - VM ディレクトリをバケットと同期させる
- Pub/Sub を使用したオブジェクト変更通知

Cloud Storage には、オブジェクトを自動で削除またはアーカイブできる、オブジェクトのライフサイクル管理機能もあります。



## Cloud Storage の機能

- 顧客指定の暗号鍵 (CSEK)
  - Google が管理する鍵ではなく独自の鍵を使う
- オブジェクトのライフサイクル管理
  - オブジェクトを自動で削除またはアーカイブする
- オブジェクトのバージョンング
  - オブジェクトの複数のバージョンを管理する
- ディレクトリ同期
  - VM ディレクトリをバケットと同期させる
- Pub/Sub を使用したオブジェクト変更通知

また、バケット内でオブジェクトの複数のバージョンを管理できる、オブジェクトのバージョンング機能もあります。複数のファイルがある場合と同様に課金されるので、そのことは念頭に置いておいてください。

## Cloud Storage の機能

- 顧客指定の暗号鍵 (CSEK)
  - Google が管理する鍵ではなく独自の鍵を使う
- オブジェクトのライフサイクル管理
  - オブジェクトを自動で削除またはアーカイブする
- オブジェクトのバージョンング
  - オブジェクトの複数のバージョンを管理する
- ディレクトリ同期
  - VM ディレクトリをバケットと同期させる
- Pub/Sub を使用したオブジェクト変更通知

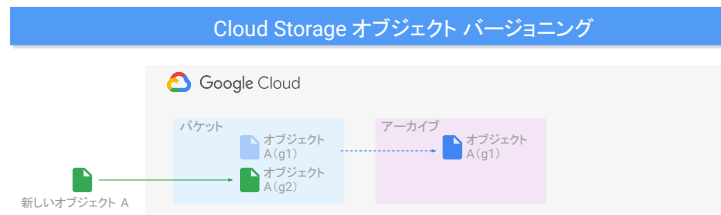
Cloud Storage には、VM ディレクトリをバケットと同期させるディレクトリ同期機能もあります。

## Cloud Storage の機能

- 顧客指定の暗号鍵 (CSEK)
  - Google が管理する鍵ではなく独自の鍵を使う
- オブジェクトのライフサイクル管理
  - オブジェクトを自動で削除またはアーカイブする
- オブジェクトのバージョンング
  - オブジェクトの複数のバージョンを管理する
- ディレクトリ同期
  - VM ディレクトリをバケットと同期させる
- Pub/Sub を使用したオブジェクト変更通知

Cloud Storage のオブジェクト変更通知は、Pub/Sub を使って構成できます。詳しくは後ほど説明します。

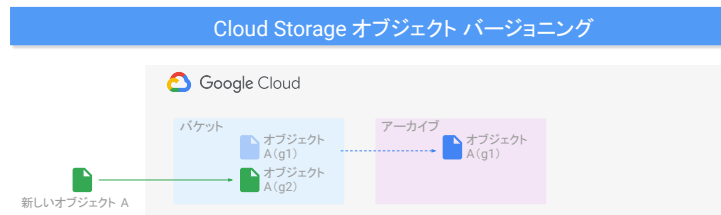
# オブジェクトのバージョンング - 削除または上書きされたオブジェクトの取得をサポート



- オブジェクトは不変。
- オブジェクトのバージョンング:
  - オブジェクト変更履歴を保持
  - オブジェクトのアーカイブ バージョンの一覧表示、オブジェクトの古い状態への復元、アーカイブ バージョンの削除が可能

Cloud Storage ではオブジェクトは不変であり、アップロードされたオブジェクトは、その保存期間が終わるまで変わることはありません。削除または上書きされたオブジェクトを取得できるようにするため、Cloud Storage にはオブジェクトのバージョンング機能があります。

# オブジェクトのバージョンング - 削除または上書きされたオブジェクトの取得をサポート



- オブジェクトは不変。
- オブジェクトのバージョンング:
  - オブジェクト変更履歴を保持
  - オブジェクトのアーカイブ バージョンの一覧表示、オブジェクトの古い状態への復元、アーカイブ バージョンの削除が可能

オブジェクトのバージョンングはバケットで有効にすることができます。有効にすると、オブジェクトのライブ バージョンが上書きまたは削除されるたびに、Cloud Storage によってオブジェクトのアーカイブ バージョンが作成されます。アーカイブ バージョンでもオブジェクト名は変更されませんが、世代番号(このスライドでは g1)はそれぞれ異なります。

オブジェクトのバージョンングを有効にすると、必要に応じて、オブジェクトのアーカイブ バージョンを一覧表示したり、オブジェクトのライブ バージョンを古い状態に復元したり、アーカイブ バージョンを完全に削除したりできます。バケットのバージョンングのオン / オフは、いつでも切り替えることができます。バージョンングをオフにすると、既存のオブジェクト バージョンはそのまま保持され、オブジェクトのアーカイブ バージョンを新たに作成して蓄積する処理が停止されます。

オブジェクトのバージョンングの詳細については、この動画のリンク セクションをご覧ください。<https://cloud.google.com/storage/docs/object-versioning>

## オブジェクトのライフサイクル管理ポリシー - 特定のルールに一致するオブジェクトに 実行するアクションを指定

- 例:
  - 1 年以上経過したオブジェクトのストレージ クラスをダウングレードする
  - 特定の日より前に作成されたオブジェクトを削除する
  - オブジェクトの最新のバージョン 3 つのみを維持する
- オブジェクトの検査は非同期バッチで実行される
- 変更が適用されるまで 24 時間かかる場合がある

Cloud Storage では、オブジェクトの有効期間の設定、古いバージョンのアーカイブ、コスト管理を容易にするためのストレージ クラスのダウングレードといった、一般的な作業をサポートするため、オブジェクトのライフサイクル管理を提供しています。

バケットにライフサイクル管理の構成を割り当てることができます。 この構成は、バケット内のすべてのオブジェクトに適用される一連のルールです。オブジェクトがいずれかのルールの条件に一致すると、指定した操作が自動的に実行されます。

## オブジェクトのライフサイクル管理ポリシー - 特定のルールに一致するオブジェクトに 実行するアクションを指定

- 例:

- 1 年以上経過したオブジェクトのストレージ クラスをダウングレードする
- 特定の日より前に作成されたオブジェクトを削除する
- オブジェクトの最新のバージョン 3 つのみを維持する
- オブジェクトの検査は非同期バッチで実行される
- 変更が適用されるまで 24 時間かかる場合がある

次のような処理が例として挙げられます。

- 1. 1 年以上経過したオブジェクトのストレージ クラスを Coldline Storage にダウングレードする
- 2. 特定の日より前に作成されたオブジェクトを削除する ( 2017 年 1 月 1 日など)
- 3. バージョニングが有効になっているバケット内の各オブジェクトで、最新のバージョン 3 つのみを維持する

## オブジェクトのライフサイクル管理ポリシー - 特定のルールに一致するオブジェクトに 実行するアクションを指定

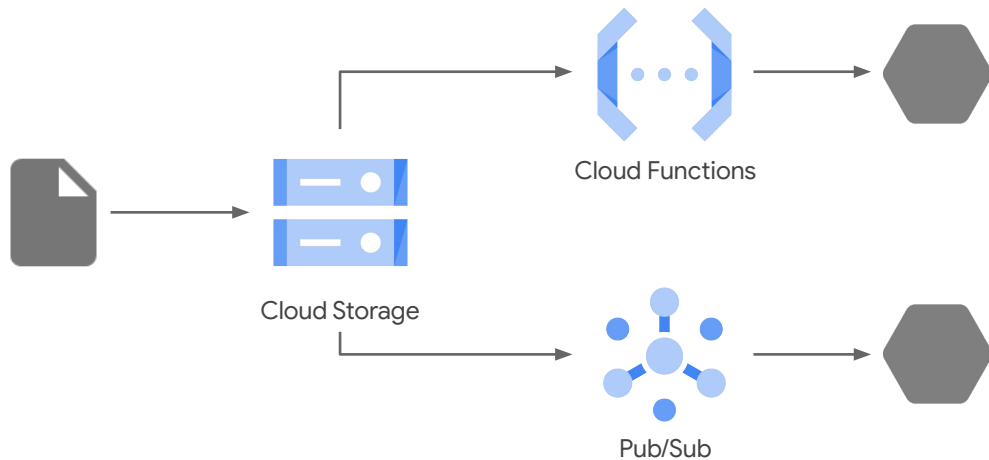
- 例:
  - 1 年以上経過したオブジェクトのストレージ クラスをダウングレードする
  - 特定の日より前に作成されたオブジェクトを削除する
  - オブジェクトの最新のバージョン 3 つのみを維持する
- オブジェクトの検査は非同期バッチで実行される
- 変更が適用されるまで 24 時間かかる場合がある

オブジェクトの検査は非同期バッチで実行されるため、ルールがすぐに適用されない場合があります。また、ライフサイクル構成に対する更新が反映されるまでに、最大で 24 時間かかります。したがって、ライフサイクル管理の構成を変更してから 24 時間が経過するまでは、古い構成で操作が実行される可能性があります。注意事項として覚えておいてください。

オブジェクトのライフサイクル管理の詳細については、この動画のリンク セクションをご覧ください。<https://cloud.google.com/storage/docs/lifecycle>



## Cloud Storage の Pub/Sub 通知



Cloud Storage では Pub/Sub を使ってオブジェクト変更通知を構成できます。Pub/Sub 通知は、バケット内のオブジェクトに変更が加えられるとその情報を Pub/Sub に送信します。情報は、選択した Pub/Sub トピックにメッセージの形式で追加されます。これにより、バケット内で作成されたオブジェクトと削除されたオブジェクトの追跡などを行えます。各通知には、トリガーとなったイベントと変更されたオブジェクトの両方が含まれます。通知は、必要な権限が自分に付与されているプロジェクトの任意の Pub/Sub トピックに送信できます。Pub/Sub トピックで受信されたメッセージは、そのトピックの任意の数のサブスクライバーに送信できます。

Cloud Storage バケットを Pub/Sub トピックに接続する方法の詳細については、この動画のリンク セクションにある前提条件のドキュメントを参照してください。

[\[https://cloud.google.com/storage/docs/reporting-changes#prereqs\]](https://cloud.google.com/storage/docs/reporting-changes#prereqs)

Pub/Sub 通知は高速で柔軟性に優れ、簡単に設定でき、コスト効率が高いため、Cloud Storage バケット内のオブジェクトの変更を追跡する方法として推奨されています。

Pub/Sub は Google の分散型リアルタイム メッセージング サービスで、アプリケーションの開発の学習トラックに含まれるトピックです。

## データ インポート サービス

- **Transfer Appliance:** ラックマウントした機器にデータをキャプチャして Google Cloud に移動
- **Storage Transfer Service:** オンライン データ (別のバケット、S3 バケット、またはウェブソース) をインポート
- **オフライン メディア インポート:** サードパーティ プロバイダが物理メディアを使ってデータをアップロード



Cloud Console を使用すると、個別のファイルをバケットにアップロードできます。アップロードするデータが数テラバイト、または数ペタバイトある場合は、この規模のアップロードに対応できるサービスとして、Transfer Appliance、Storage Transfer Service、オフラインメディア インポートの 3 つがあります。

## データ インポート サービス

- **Transfer Appliance:** ラックマウントした機器にデータをキャプチャして Google Cloud に移動
- **Storage Transfer Service:** オンライン データ (別のバケット、S3 バケット、またはウェブソース) をインポート
- **オフライン メディア インポート:** サードパーティ プロバイダが物理メディアを使ってデータをアップロード



Transfer Appliance は、業務を中断することなく、大量のデータ(数百テラバイトから 1 ペタバイトまで)を Google Cloud に安全に移行できるハードウェア アプライアンスです。このスライドの画像は Transfer Appliance のものです。

## データ インポート サービス

- **Transfer Appliance:** ラックマウントした機器にデータをキャプチャして Google Cloud に移動



- **Storage Transfer Service:** オンライン データ (別のバケット、S3 バケット、またはウェブソース) をインポート

- **オフライン メディア インポート:** サードパーティ プロバイダが物理メディアを使ってデータをアップロード



Storage Transfer Service は、オンライン データを高いパフォーマンスでインポートすることができます。データソースとして、別の Cloud Storage バケット、Amazon S3 バケット、HTTP/HTTPS のロケーションなどを使用できます。

## データ インポート サービス

- **Transfer Appliance:** ラックマウントした機器にデータをキャプチャして Google Cloud に移動
- **Storage Transfer Service:** オンライン データ (別のバケット、S3 バケット、またはウェブソース) をインポート
- **オフライン メディア インポート:** サードパーティ プロバイダが物理メディアを使ってデータをアップロード



最後に紹介するオフライン メディア インポートは、サードパーティ サービスで、データをアップロードするプロバイダに物理メディア (ストレージ アレイ、ハードディスク ドライブ、テープ、USB フラッシュ ドライブ) を送付します。

この 3 つのサービスについて詳しくは、この動画のリンク セクションをご覧ください。

<https://cloud.google.com/transfer-appliance/>

<https://cloud.google.com/storage-transfer/docs/>

<https://cloud.google.com/storage/docs/offline-media-import-export>

## Cloud Storage - グローバルな強整合性を提供

- 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



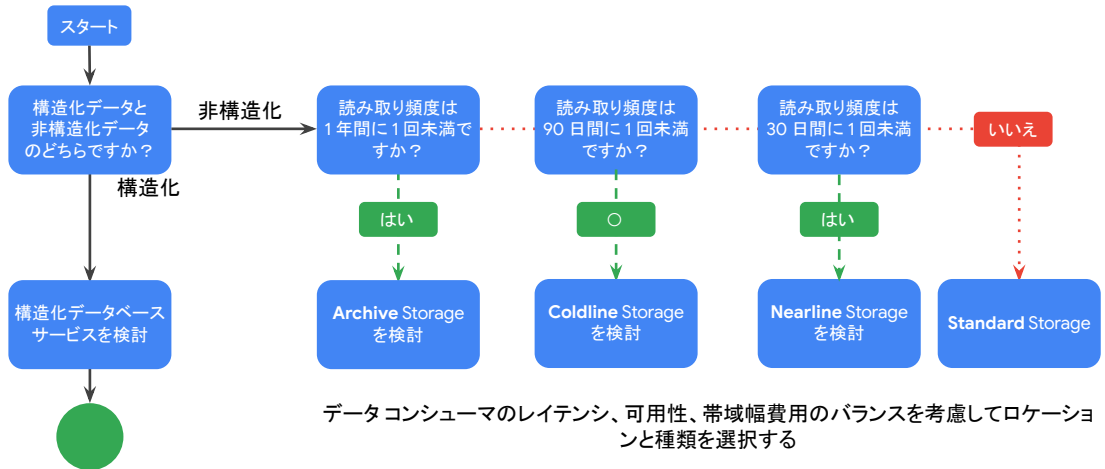
Cloud Storage にオブジェクトをアップロードし、成功のレスポンスを受け取ると、オブジェクトは直ちに、Google がサービスを提供するすべてのロケーションからのダウンロード オペレーションとメタデータ オペレーションの対象になります。これは、新しいオブジェクトを作成する場合でも、既存のオブジェクトを上書きする場合でも同じです。アップロードは強整合性なので、read-after-write または read-after-metadata-update オペレーションで 404 Not Found レスポンスや古いデータを受け取ることはありません。

グローバルな強整合性は、オブジェクトの削除オペレーションにも当てはまります。削除リクエストが成功した直後に、オブジェクトまたはそのメタデータをダウンロードしようとする、404 Not Found ステータス コードが返されます。削除オペレーションが成功した後はオブジェクトが存在しないため、404 エラーが返されます。

バケットの一覧表示は強整合性のオペレーションです。たとえば、バケットを作成し、その直後に list buckets オペレーションを行うと、返されたバケットのリストに新しいバケットが表示されます。

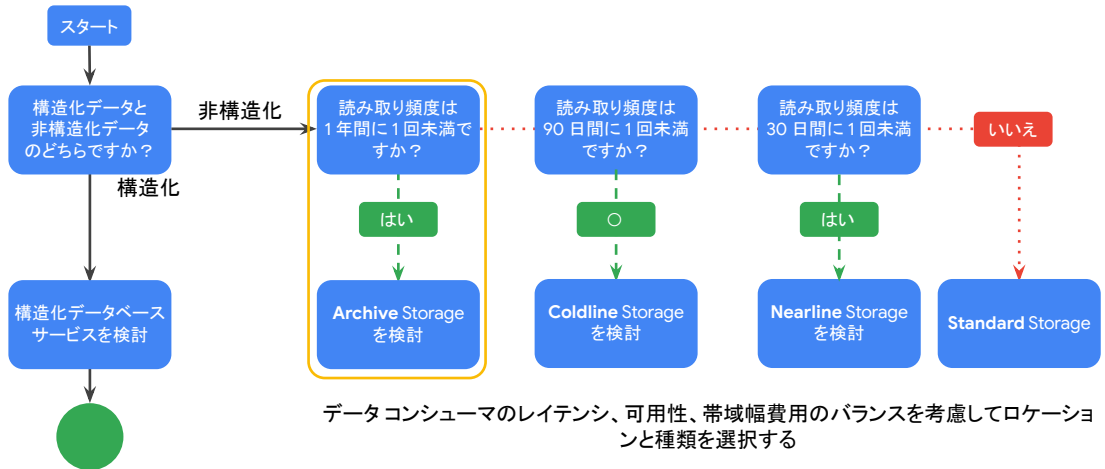
最後に、オブジェクトの一覧表示も強整合性のオペレーションです。たとえば、バケットにオブジェクトをアップロードし、その直後に list objects オペレーションを行うと、返されたオブジェクトのリストに新しいオブジェクトが表示されます。

## ストレージ クラスの選択



このディシジョン ツリーを使用して、Cloud Storage の適切なストレージ クラスを判断しましょう。

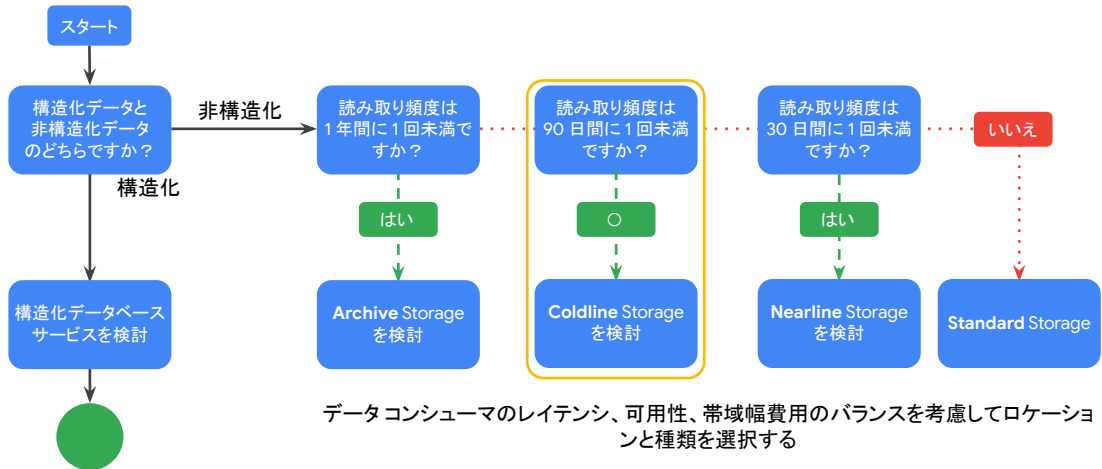
## ストレージ クラスの選択



データの読み取り頻度が 1 年間に 1 回未満の場合は、**Archive Storage** の使用を検討します。

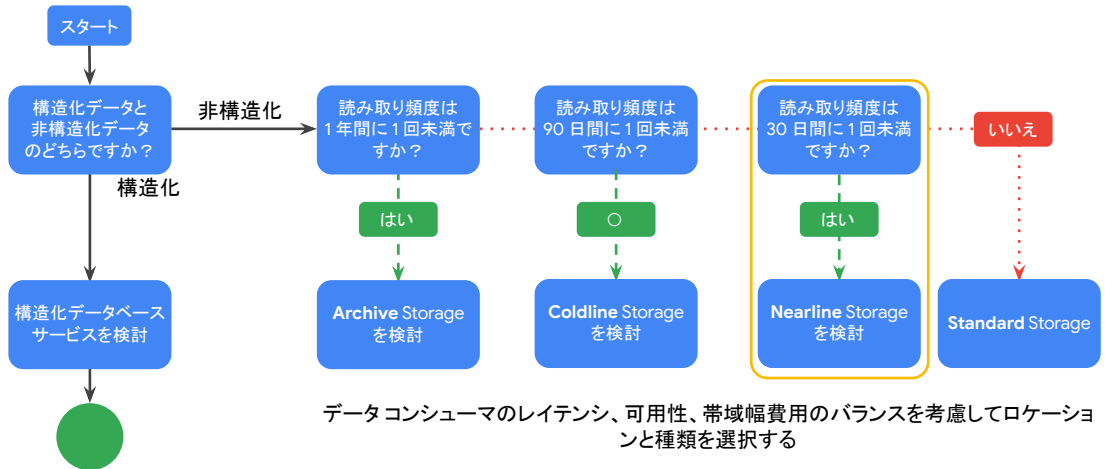


## ストレージ クラスの選択



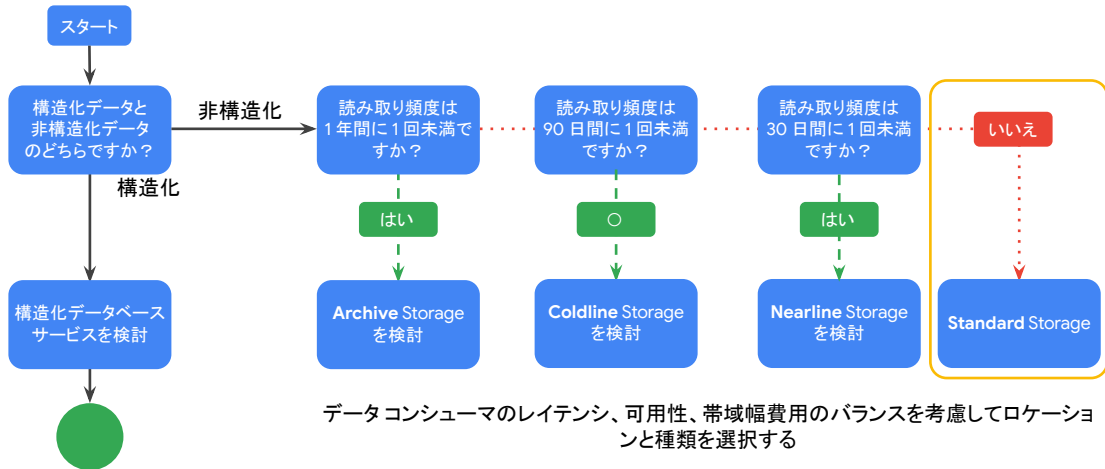
データの読み取り頻度が 90 日間に 1 回未満の場合は、**Coldline Storage** の使用を検討します。

## ストレージ クラスの選択



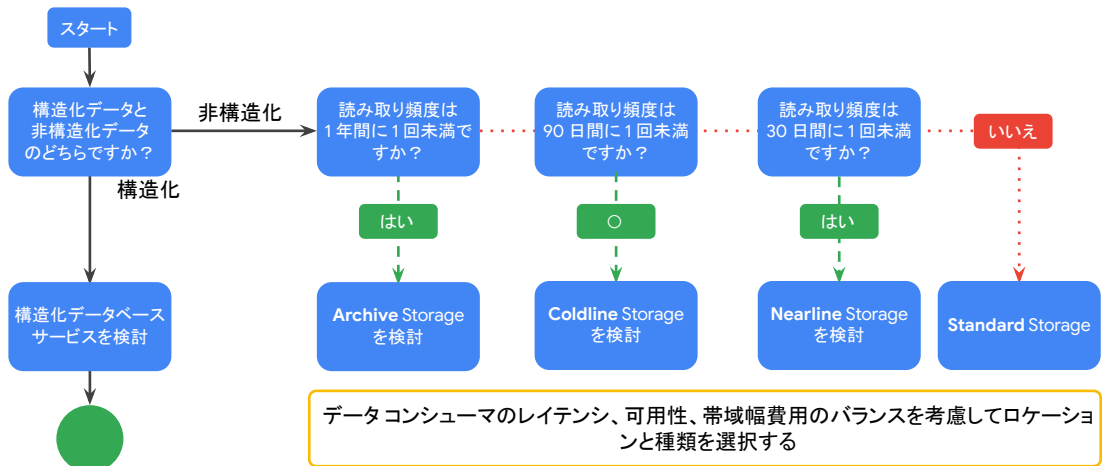
データの読み取り頻度が 30 日間に 1 回未満の場合は、**Nearline Storage** の使用を検討します。

## ストレージ クラスの選択



これよりも多い頻度でデータの読み書きを行う場合は、 **Standard Storage** の使用を検討します。

## ストレージ クラスの選択



ロケーション タイプについても考慮する必要があります。

- 同一リージョンにグループ化されている分析パイプラインなどのデータ コンシューマに対するレイテンシ、ネットワーク帯域幅を最適化する場合は、**リージョン**のロケーションを使用します。
- リージョンと同等のパフォーマンスに加えて地理的な冗長性による高可用性が必要な場合は、**デュアルリージョン**を使用します。
- Google ネットワークの外部にあり、広域に分散しているデータ コンシューマにコンテンツを配信する場合や、地理的冗長性による高可用性を求める場合は、**マルチリージョン**を使用します。

## Filestore - アプリケーション向けのマネージド ファイル ストレージ サービス



Filestore は、データ用のファイル システム インターフェースと共有ファイル システムを必要とするアプリケーション向けのマネージド ファイル ストレージ サービスです。

## Filestore - アプリケーション向けのマネージド ファイル ストレージ サービス

- Compute Engine インスタンスと GKE インスタンス向けのフル  
マネージド ネットワーク接続ストレージ (NAS)



Filestore を使用することにより、Compute Engine や Google Kubernetes Engine のインスタンスでマネージド ネットワーク接続ストレージ (NAS) をネイティブで簡単に活用できるようになります。

## Filestore - アプリケーション向けのマネージド ファイル ストレージ サービス

- Compute Engine インスタンスと GKE インスタンス向けのフル  
マネージド ネットワーク接続ストレージ(NAS)
- 予測可能なパフォーマンス



Filestore のパフォーマンスと容量は個別に細かく調整できるため、ファイルベースのワークロードに対して、想定どおりに高速のパフォーマンスを実現できます。

## Filestore - アプリケーション向けのマネージド ファイル ストレージ サービス

- Compute Engine インスタンスと GKE インスタンス向けのフル  
マネージド ネットワーク接続ストレージ(NAS)
- 予測可能なパフォーマンス
- NFSv3 のフルサポート



Filestore は既存のエンタープライズ アプリケーションにネイティブで対応し、NFSv3 互換のクライアントをサポートしています。



## Filestore - アプリケーション向けのマネージド ファイル ストレージ サービス

- Compute Engine インスタンスと GKE インスタンス向けのフル  
マネージド ネットワーク接続ストレージ (NAS)
- 予測可能なパフォーマンス
- NFSv3 のフルサポート
- 数百 TB までスケールして高パフォーマンスの  
ワークロードに対応可能



スケールアウト パフォーマンス、数百 TB の容量、ファイルロックなどの機能のメリットをアプリケーションで得られます。特別なプラグインやクライアント側ソフトウェアのインストールやメンテナンスは必要ありません。

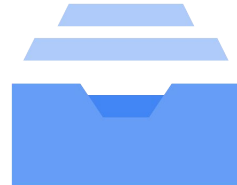
## Filestore の多様なユースケース



Filestore には多様なユースケースがあります。

## Filestore の多様なユースケース

- アプリケーションの移行



Filestore を使うことにより、エンタープライズ アプリケーションの移行を加速できます。多くのオンプレミス アプリケーションが、データに対するファイル システム インターフェースを必要としています。こうしたアプリケーションがクラウドへの移行を進める中、Filestore も共有ファイル システムを必要とするエンタープライズ アプリケーションを幅広くサポートしています。

## Filestore の多様なユースケース

- アプリケーションの移行
- メディアのレンダリング



メディアをレンダリングする場合、視覚効果アーティストは、Compute Engine インスタンスに簡単にマウントできる Filestore ファイル共有を活用して共同で作業を行えます。レンダリング ワークフローはコンピューティング マシンのフリート(「レンダー ファーム」)で実行されるのが一般的で、その際にすべてのマシンが同じ共有ファイル システムをマウントするため、レンダリングのニーズに応じて Filestore と Compute Engine をスケーリングできます。

## Filestore の多様なユースケース

- アプリケーションの移行
- メディアのレンダリング
- 電子設計自動化 (EDA)



電子設計自動化 (EDA) ではデータ管理が重要です。数千コアにまたがるワークロードを一括処理できる能力と大容量メモリが必要です。Filestore は、製造業のお客様の高度な EDA のニーズにも対応できる容量と規模を備えています。また、Filestore を使用すると、ファイルにどこからでもアクセスできるようになります。

## Filestore の多様なユースケース

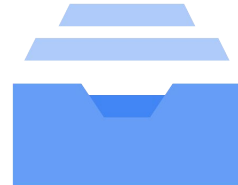
- アプリケーションの移行
- メディアのレンダリング
- 電子設計自動化 (EDA)
- データ分析



データ分析ワークロードには、複雑な財務モデルの計算や環境データの分析などが含まれます。これらはレイテンシの影響を受けやすいワークロードです。Filestore では低いレイテンシでファイルを操作することができ、容量またはパフォーマンスのニーズが変わったときも、必要に応じてインスタンスを簡単に拡大または縮小することができます。Filestore は永続的かつ共有可能なストレージ層であり、高パフォーマンスなスマート アナリティクス向けにデータへの即時アクセスを可能にします。クライアントのドライブへのデータのロードとオフロードに、貴重な時間を使う必要がありません。

## Filestore の多様なユースケース

- アプリケーションの移行
- メディアのレンダリング
- 電子設計自動化 (EDA)
- データ分析
- ゲノム処理



ゲノム解読には、1 人につき数十億データポイントにもものぼる膨大な量の生データが必要です。このような分析には、速度、スケーラビリティ、セキュリティが欠かせません。Filestore は、科学的研究を実施している企業や研究機関の要望に応えると同時に、パフォーマンスに合った予測可能な料金も提示します。

## Filestore の多様なユースケース

- アプリケーションの移行
- メディアのレンダリング
- 電子設計自動化 (EDA)
- データ分析
- ゲノム処理
- ウェブ コンテンツ管理



ウェブ デベロッパーと大規模なホスティング プロバイダも、WordPress ホスティングのようなニーズなど、ウェブ コンテンツの管理と提供に Filestore を利用しています。



# ラボ

---

## Cloud Storage

先ほど説明した Cloud Storage のコンセプトをラボで実際に用いてみましょう。

このラボでは、バケットを作成し、Cloud Storage で使用できる多くの高度なオプションを実行します。アクセス制御リストを設定して、データにアクセスできるユーザーと、データに対して実行できる操作を制限します。高いセキュリティを得るために、独自の暗号鍵を指定して管理する機能を使用します。オブジェクトのバージョンニング機能を有効にしてデータの変更を追跡し、ライフサイクル管理を構成して、指定の期間が経過したらオブジェクトが自動的にアーカイブまたは削除されるようにします。最後に、先ほど説明したディレクトリ同期機能を利用し、Cloud IAM を使用してバケットを複数のプロジェクトで共有します。

# ラボの復習

---

## Cloud Storage

このラボでは、バケットとオブジェクトの作成と操作の方法を学習しました。また、 Cloud Storage の次の機能を適用しました。

- 顧客指定の暗号鍵
- アクセス制御リスト
- ライフサイクル管理
- オブジェクトのバージョニング
- ディレクトリ同期
- IAM を使用したプロジェクト間でのリソース共有

Cloud Storage の高度な機能の多くに慣れた今、以前は考えたこともなかったような、さまざまなアプリケーションでそれらを使用することを検討している方もいらっしゃるでしょう。GCP をすぐに使い始めるには、Cloud Storage をバックアップ サービスとして使用するのが簡単で一般的です。

この後、ラボのチュートリアルを参照できますが、GCP のユーザー インターフェースは変更されることがあるため、実際の環境は見た目が少し異なる場合があります。

## アジェンダ

Cloud Storage と Filestore

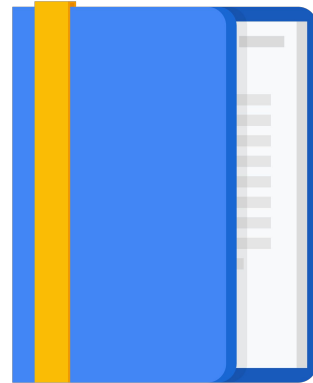
Cloud SQL

Cloud Spanner

Firestore

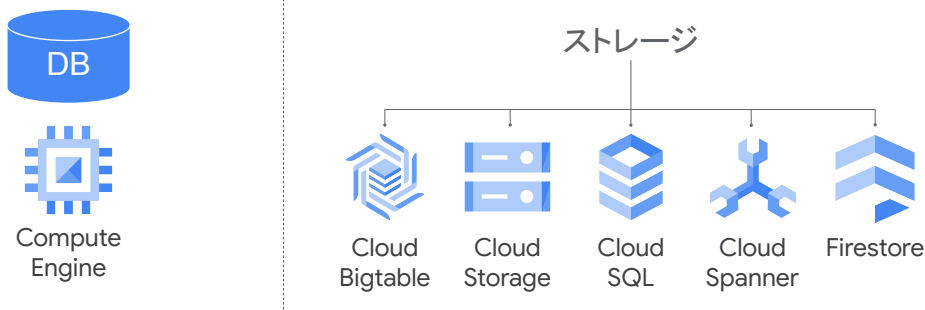
Cloud Bigtable

Memorystore



構造化データベース サービスまたはリレーショナル データベース サービスについて詳しく見ていきましょう。最初に取り上げるのは Cloud SQL です。

## 独自のデータベース ソリューションを構築するか マネージド サービスを利用



Compute Engine を使用して SQL Server アプリケーション イメージを VM にインストールできるにもかかわらず、SQL 向けの Google Cloud サービスを使用するのはなぜでしょうか。

これはまさに、独自のデータベース ソリューションを構築すべきか、それともマネージド サービスを利用すべきかという問題です。マネージド サービスの利用にはメリットがあります。それでは、Cloud SQL を Google Cloud 内でマネージド サービスとして使用する理由について学びましょう。

## Cloud SQL - フルマネージド データベース サービス (MySQL、PostgreSQL、または Microsoft SQL Server)

- パッチと更新が自動で適用される
- MySQL ユーザーは自分で管理する
- Cloud SQL は多くのクライアントをサポートしている
  - gcloud sql
  - App Engine、Google Workspace スクリプト
  - アプリケーションとツール
    - SQL Workbench、Toad
    - 標準 MySQL ドライバを使用する外部アプリケーション



Cloud SQL

Cloud SQL は、MySQL、PostgreSQL、Microsoft SQL Server の各データベースのフルマネージド サービスです。

## Cloud SQL - フルマネージド データベース サービス (MySQL、PostgreSQL、または Microsoft SQL Server)

- パッチと更新が自動で適用される
- MySQL ユーザーは自分で管理する
- Cloud SQL は多くのクライアントをサポートしている
  - gcloud sql
  - App Engine、Google Workspace スクリプト
  - アプリケーションとツール
    - SQL Workbench、Toad
    - 標準 MySQL ドライバを使用する外部アプリケーション



Cloud SQL

つまり、パッチと更新は自動的に適用されますが、MySQL ユーザーの管理については、これらのデータベースに付属するネイティブの認証ツールを使用する必要があります。

## Cloud SQL - フルマネージド データベース サービス (MySQL、PostgreSQL、または Microsoft SQL Server)

- パッチと更新が自動で適用される
- MySQL ユーザーは自分で管理する
- Cloud SQL は多くのクライアントをサポートしている
  - gcloud sql
  - App Engine、Google Workspace スクリプト
  - アプリケーションとツール
    - SQL Workbench、Toad
    - 標準 MySQL ドライバを使用する外部アプリケーション



Cloud SQL

Cloud SQL は、Cloud Shell、App Engine、Google Workspace スクリプトなど、多数のクライアントをサポートしています。また、SQL Workbench や Toad など、よく使われる他のアプリケーションやツール、または標準 MySQL ドライバを使用するその他の外部アプリケーションもサポートしています。

## Cloud SQL インスタンス

### パフォーマンス:

- 30 TB のストレージ
- 40,000 IOPS
- 416 GB の RAM
- リードレプリカによるスケールアウト

### 豊富な選択肢:

- MySQL 5.6、5.7(デフォルト)、または 8.0
- PostgreSQL 9.6、10、11、または 12(デフォルト)
- Microsoft SQL Server 2017



Cloud SQL は高パフォーマンスかつスケーラブルで、1 インスタンスにつき最大 30 TB のストレージ容量、40,000 の IOPS、416 GB の RAM を提供します。最大で 64 個のプロセッサコアへのスケールアップやリードレプリカによるスケールアウトも簡単です。



## Cloud SQL インスタンス

### パフォーマンス:

- 30 TB のストレージ
- 40,000 IOPS
- 416 GB の RAM
- リードレプリカによるスケールアウト

### 豊富な選択肢:

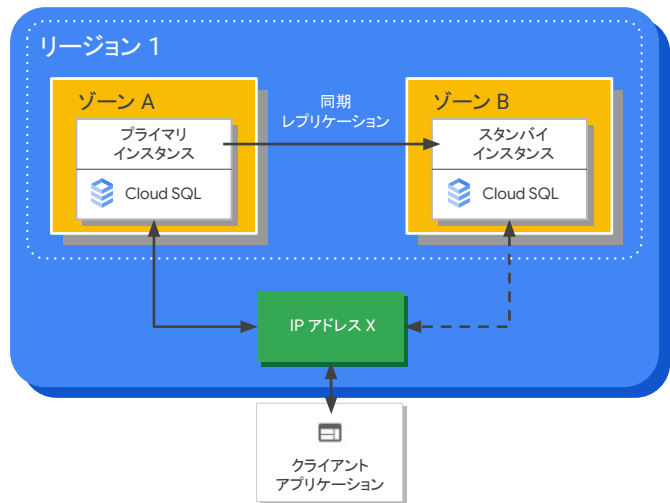
- MySQL 5.6、5.7(デフォルト)、または 8.0
- PostgreSQL 9.6、10、11、または 12(デフォルト)
- Microsoft SQL Server 2017



この収録の時点で使用可能なデータベースは、MySQL 5.6、5.7、8.0、PostgreSQL 9.6、10、11、12、SQL Server 2017 の Web、Express、Standard、Enterprise の各エディションです。

## Cloud SQL のサービス

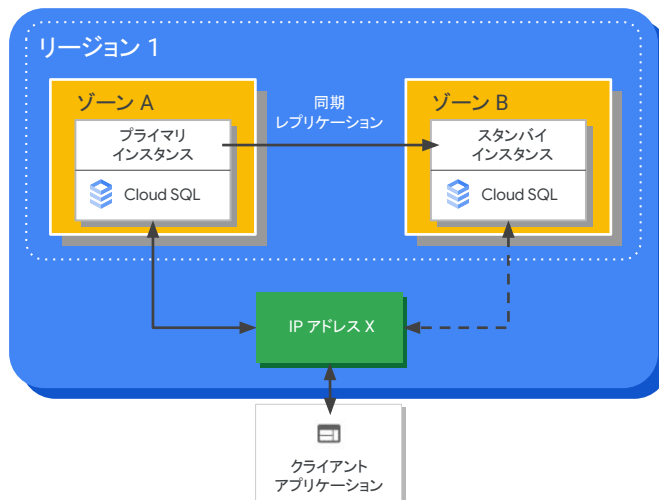
- HA 構成
- バックアップ サービス
- インポート / エクスポート
- スケーリング
  - アップ: マシンの容量
  - アウト: リードレプリカ



Cloud SQL で利用できる他のサービスに注目してみましょう。

## Cloud SQL のサービス

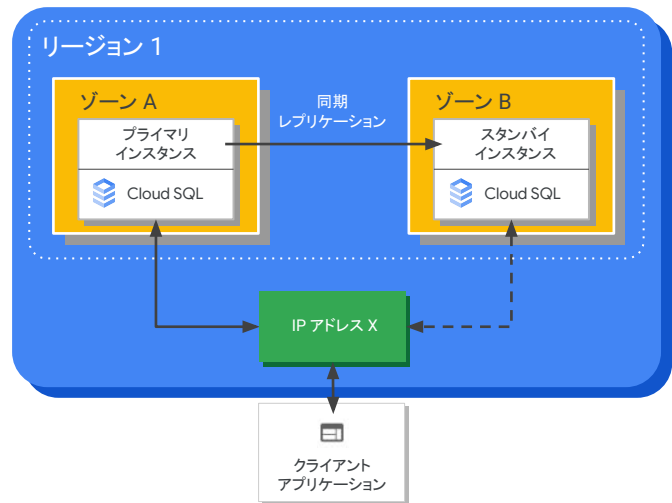
- HA 構成
- バックアップ サービス
- インポート / エクスポート
- スケーリング
  - アップ: マシンの容量
  - アウト: リードレプリカ



リージョン インスタンス内の HA 構成は、プライマリ インスタンスとスタンバイ インスタンスで構成されます。各ゾーンの永続ディスクへの 同期レプリケーション により、トランザクションが commit されたとしてレポートされる前に、プライマリ インスタンスへの書き込みのすべてが両方のゾーンのディスクに複製されます。インスタンスまたはゾーンで障害が発生した場合、永続ディスクがスタンバイ インスタンスにアタッチされて、スタンバイ インスタンスが新しいプライマリ インスタンスになります。ユーザーは新しいプライマリに再転送されます。このプロセスは、フェイルオーバーと呼ばれます。

## Cloud SQL のサービス

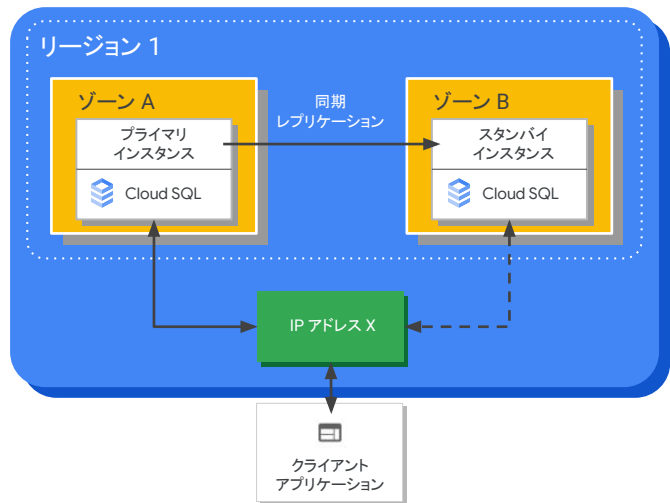
- HA 構成
- バックアップ サービス
- インポート / エクスポート
- スケーリング
  - アップ: マシンの容量
  - アウト: リードレプリカ



Cloud SQL は自動バックアップとオンデマンド バックアップも提供します。ポイントインタイム リカバリにも対応しています。

## Cloud SQL のサービス

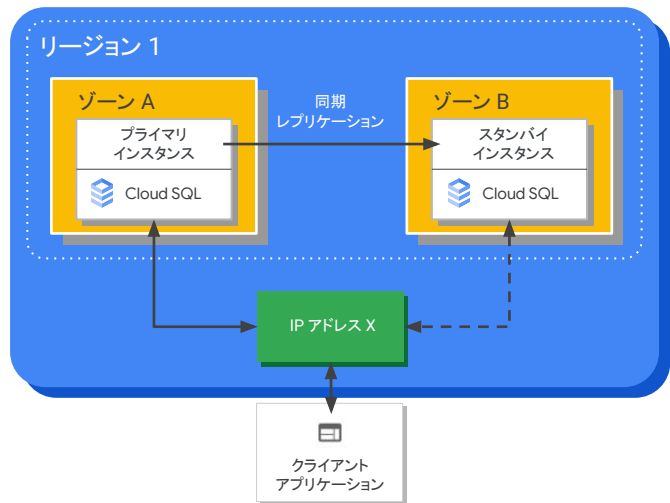
- HA 構成
- バックアップ サービス
- インポート / エクスポート
- スケーリング
  - アップ: マシンの容量
  - アウト: リードレプリカ



mysqldump を使用したデータベースのインポートとエクスポートや、CSV ファイルのインポートとエクスポートを行えます。

## Cloud SQL のサービス

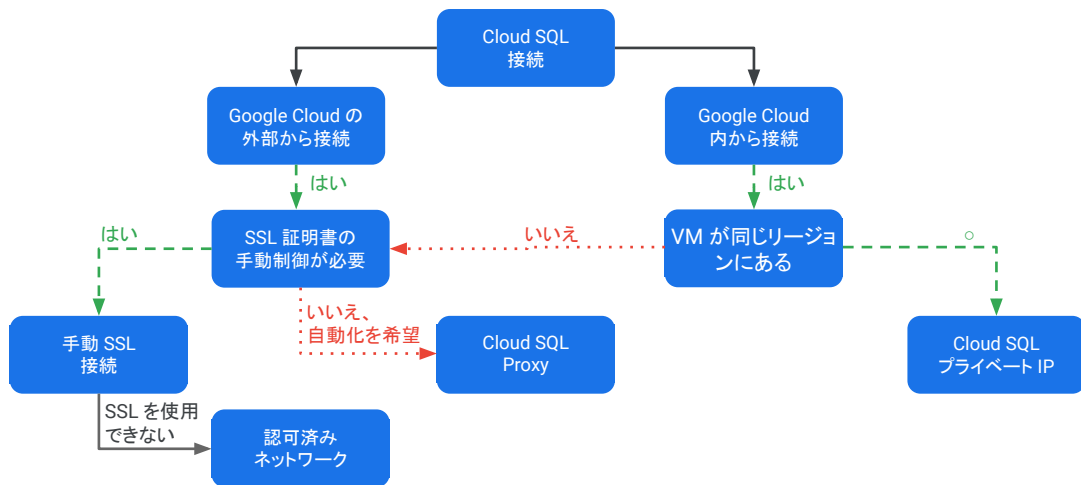
- HA 構成
- バックアップ サービス
- インポート / エクスポート
- スケーリング
  - アップ: マシンの容量
  - アウト: リードレプリカ



Cloud SQL では、スケールアップやスケールアウトも行えます。スケールアップではマシンの再起動が必要です。スケールアウトにはリードレプリカを使用します。

ただし、水平方向のスケラビリティに関心がある場合は、このモジュールで後ほど説明する Cloud Spanner の使用を検討してください。

## Cloud SQL インスタンスへの接続



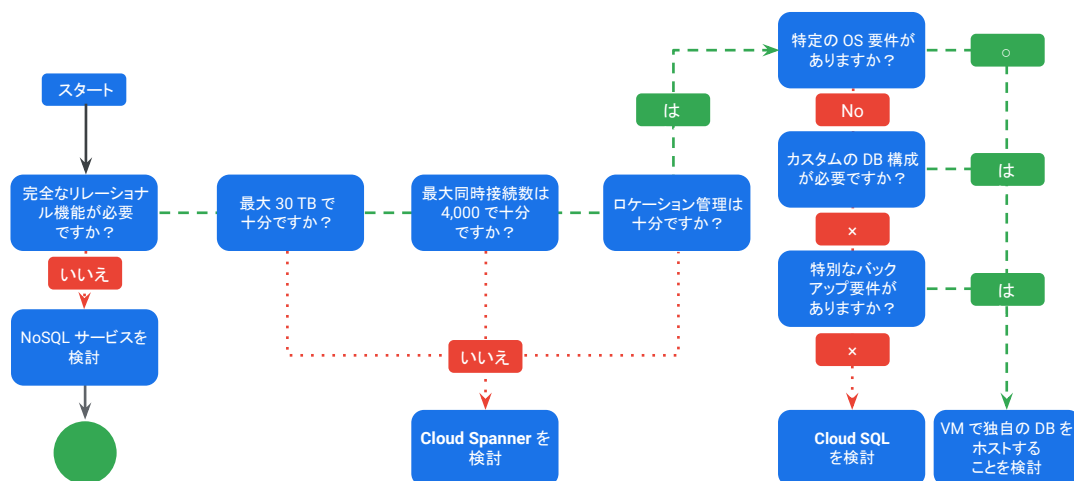
Cloud SQL インスタンスへの接続方法は、接続のセキュリティ、パフォーマンス、自動化に影響します。Cloud SQL インスタンスと同じ Google Cloud プロジェクト内でホストされ、同じリージョン内に配置されているアプリケーションを接続する場合は、プライベート IP 接続を選択すると、プライベート接続によってパフォーマンスとセキュリティが最も高い接続を実現できます。つまり、トラフィックが公共のインターネットに公開されることはありません。同じリージョンにある VM から Cloud SQL プライベート IP アドレスに接続するのは、パフォーマンスの点で推奨されるだけであり、必須ではありません。

アプリケーションが別のリージョンまたはプロジェクトでホストされる場合や、Google Cloud の外部から Cloud SQL インスタンスに接続する場合は、選択肢が 3 つあります。この場合は、Cloud SQL Proxy の使用を推奨します。これで認証、暗号化、鍵のローテーションが自動化されます。SSL 接続を手動で制御する必要がある場合は、証明書を自分で生成し、定期的にローテーションすることができます。あるいは、暗号化なしの接続を使用することもできます。その場合は、特定の IP アドレスに対して、その外部 IP アドレスを介して SQL サーバーに接続することを許可します。

それぞれの選択肢についてはこの後のラボで実習しますが、プライベート IP の詳細については、この動画のリンク セクションをご覧ください。

[<https://cloud.google.com/sql/docs/mysql/private-ip>]

## Cloud SQL の選択



最後にこのディシジョン ツリーを使用して、完全なリレーショナル機能を備えた適切なデータ ストレージ サービスを判断しましょう。

ストレージ容量が 30 TB 以上、またはデータベースへの同時接続数が 4,000 以上必要な場合、もしくはグローバルにスケールアップするためにスケーリング、可用性、ロケーション管理に対応できるアプリケーション設計が必要な場合は、このモジュールで後ほど説明する Cloud Spanner の使用を検討してください。

これらの制約が気にならない場合は、特定の OS、カスタムのデータベース構成、または特別なバックアップに関する要件がないかを確認します。いずれかの要件がある場合は、Compute Engine を使用して、独自のデータベースを VM 上でホストすることを検討してください。そのような要件がない場合は、リレーショナル データベースのフルマネージド サービスとして Cloud SQL を使うことを強く推奨します。

既存の MySQL ソリューションを使用または再実装するより、Cloud SQL をマネージド サービスとして使ったほうがよいとわかったら、MySQL から Cloud SQL への移行ソリューションをリンク セクションで参照してください。

[<https://cloud.google.com/solutions/migrating-mysql-to-cloudsql-concept>]

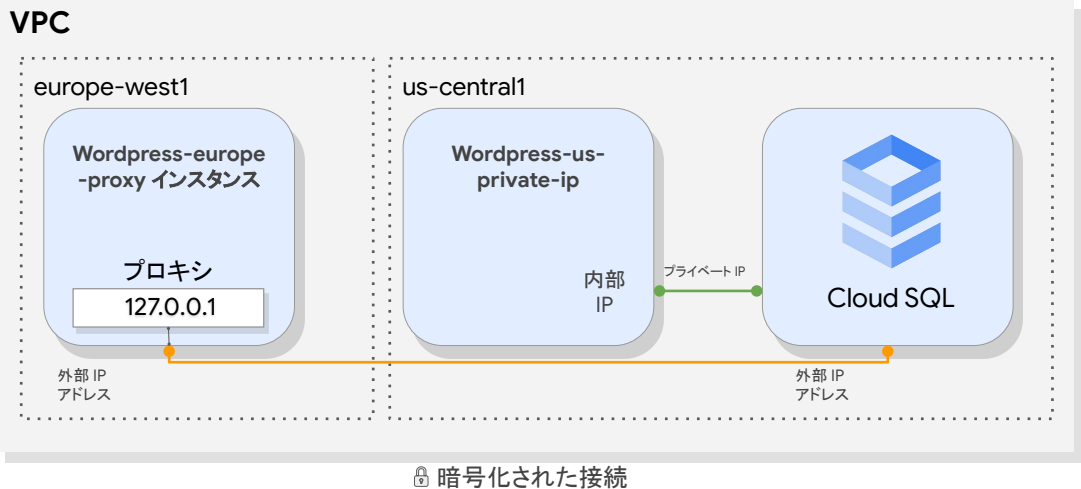


# ラボ

---

## Cloud SQL の実装

先ほど説明した Cloud SQL のコンセプトをラボで実際に用いてみましょう。



このラボでは、Cloud SQL サーバーを構成し、プロキシを介して外部接続でこのサーバーにアプリケーションを接続する方法について学びます。また、パフォーマンスとセキュリティを強化できるプライベート IP リンクによる接続も構成します。このラボのデモでは WordPress アプリを使用しますが、ここで紹介する情報とおすすめの方法は、SQL Server を必要とするどのアプリケーションにも適用できます。

このラボの終了時には、この図に示すように、WordPress フロントエンドの 2 つの動作インスタンスが 2 つの異なる接続タイプを介して SQL インスタンス バックエンドに接続されます。

# ラボの復習

---

## Cloud SQL の実装

このラボでは、Cloud SQL データベースを作成し、安全なプロキシを介した外部接続と、より安全でパフォーマンスの高いプライベート IP アドレスの両方を使用するように構成しました。

プライベート IP 経由で接続できるのは、アプリケーションと Cloud SQL サーバーが同じリージョンに配置され、同じ VPC ネットワークに含まれている場合のみです。アプリケーションが別のリージョン、VPC、プロジェクトでホストされている場合は、プロキシを使用して外部接続でその接続を保護します。

この後、ラボのチュートリアルを参照できますが、GCP のユーザー インターフェースは変更されることがあるため、実際の環境は見た目が少し異なる場合があります。

## アジェンダ

Cloud Storage と Filestore

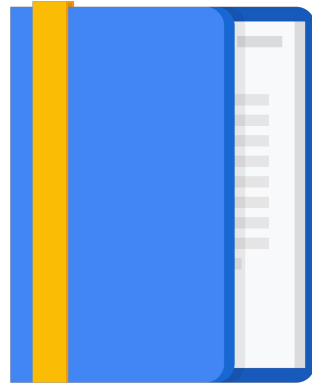
Cloud SQL

Cloud Spanner

Firestore

Cloud Bigtable

Memorystore



水平方向のスケーラビリティが必要なため、Cloud SQL で要件を満たせない場合は、Cloud Spanner の使用を検討してください。

# Cloud Spanner - リレーショナル データベース構造の利点と非リレーショナルの水平スケーリングの利点を兼備

- ペタバイト規模のスケーリング
- 強整合性
- 高可用性
- 財務 / 在庫管理アプリケーションに利用
- 月間稼働率
  - マルチリージョン: 99.999%
  - リージョン: 99.99%



Cloud Spanner は、リレーショナル データベース構造の利点と非リレーショナルの水平スケーリングの利点をクラウドで提供するために構築されたサービスです。

このサービスはペタバイト規模の容量に対応し、グローバルなトランザクション整合性、スキーマ、SQL、自動の同期レプリケーションによる高可用性を提供します。ユースケースとしては、従来はリレーショナル データベース テクノロジーで処理されていた財務アプリケーションと在庫管理アプリケーションなどがあります。

このスライドに示したとおり、月間稼働率の SLA は、マルチリージョンとリージョンのどちらのインスタンスを作成するかで異なります。ただし、最新の数値については、この動画のリンク セクションにあるドキュメントを必ず参照してください。

[\[https://cloud.google.com/spanner/sla\]](https://cloud.google.com/spanner/sla)

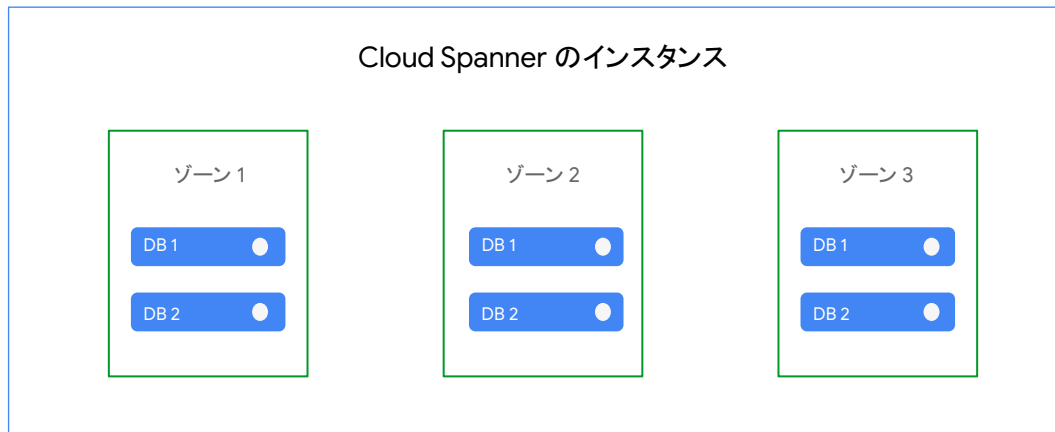
## 特性

	Cloud Spanner		リレーショナル DB		非リレーショナル DB	
スキーマ	✓	○	✓	○	✗	×
SQL	✓	○	✓	○	✗	×
整合性	✓	強	✓	強	✗	結果
可用性	✓	高	✗	フェイル オーバー	✓	高
スケーラビリティ	✓	水平型	✗	垂直型	✓	水平型
レプリケーション	✓	自動	↻	構成可能	↻	構成可能

それでは、Cloud Spanner をリレーショナルと非リレーショナルの両方のデータベースと比べてみましょう。 Cloud Spanner にはスキーマ、SQL、強整合性があり、この点はリレーショナル データベースと同じです。また、Cloud Spanner は高可用性、水平方向のスケーラビリティ、構成可能なレプリケーションを提供しますが、この点は非リレーショナルデータベースと同じです。

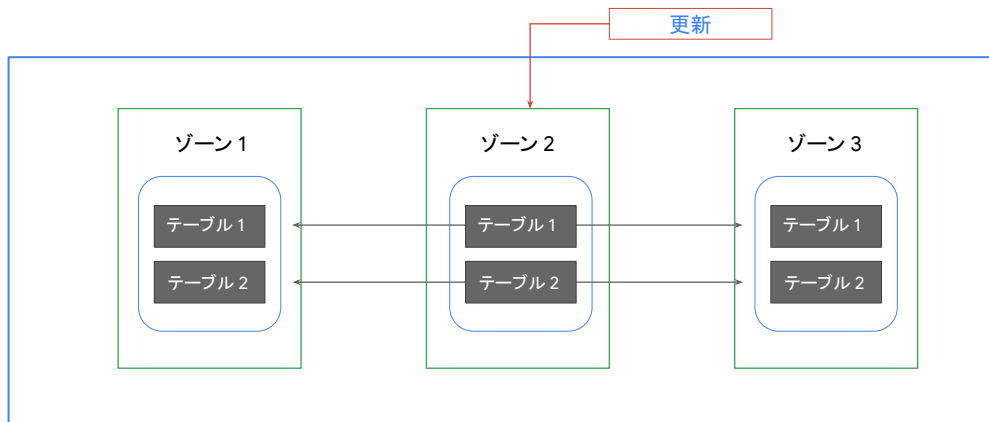
先ほども触れましたが、Cloud Spanner はリレーショナルと非リレーショナルの利点を兼ね備えています。これらの機能を利用することにより、金融サービス業や小売業のトランザクションおよび在庫管理用に整合性のあるシステムを構築するといった、ミッション クリティカルなユースケースに対応できます。この仕組みを理解するため、 Cloud Spanner のアーキテクチャを見ていきましょう。

## Cloud Spanner のアーキテクチャ



Cloud Spanner のインスタンスは、データを N 個のクラウドゾーンに複製します。これらのゾーンは 1 つのリージョンに含めるか、複数のリージョンに分散させることができます。データベースの配置は構成可能です。つまり、データベースを配置するリージョンを選べます。このアーキテクチャによって高可用性とグローバルな配置が実現されます。

## Google のグローバル ファイバー ネットワークによるゾーン間の同期的なデータレプリケーション

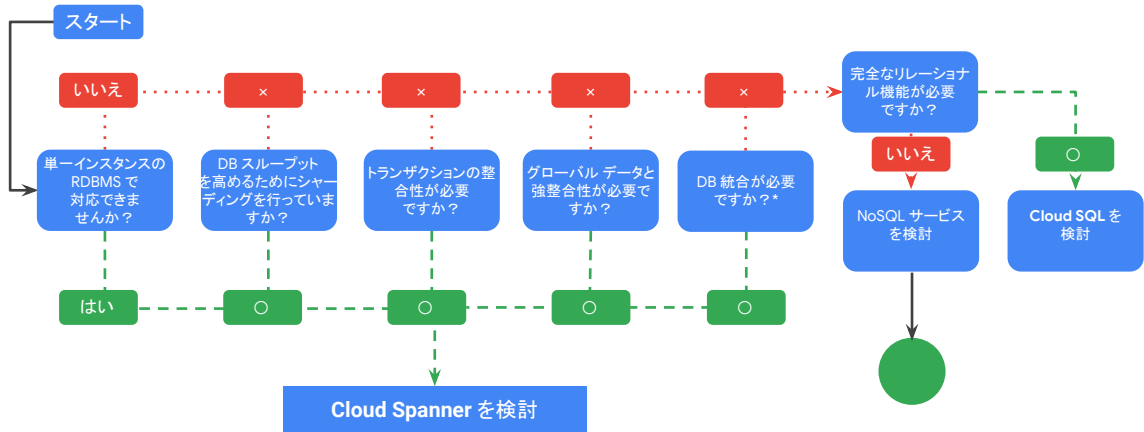


データのレプリケーションが、Google のグローバル ファイバー ネットワークを使用してゾーン間で同期されます。原子時計を使うことにより、データをいつ更新してもアトミック性が確保されます。

Cloud Spanner の詳細についての説明はここまでにして、このモジュールの目的は Cloud Spanner をいつ使用すべきかを理解することなので、ディシジョン ツリーを見ていくことにしましょう。



## Cloud Spanner の選択



リレーショナル データベースで対応できない、スループットを高めるためにデータベースをシャードイングしている、トランザクションの整合性が必要とされる、グローバル データと強整合性が必要とされる、またはデータベースを統合する必要がある場合は、 Cloud Spanner の使用を検討してください。

これらの要件がなく、完全なリレーショナル機能も不要な場合は、次に説明する Cloud Firestore のような NoSQL サービスの使用を検討してください。

既存の MySQL ソリューションを使用または再実装するより、Cloud Spanner をマネージド サービスとして使ったほうがよいとわかったら、MySQL から Cloud Spanner への移行ソリューションをリンク セクションで参照してください。

[\[https://cloud.google.com/solutions/migrating-mysql-to-spanner\]](https://cloud.google.com/solutions/migrating-mysql-to-spanner)

## アジェンダ

Cloud Storage と Filestore

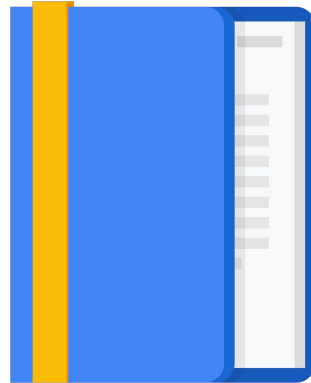
Cloud SQL

Cloud Spanner

Firestore

Cloud Bigtable

Memorystore



アプリケーション向けのスケーラビリティの高い NoSQL データベースを探している場合は、Firestore の使用を検討してください。

# Firestore - NoSQL ドキュメント データベース

- データの保存、同期、クエリを簡素化
- モバイル、ウェブ、IoT のアプリをグローバルに提供
- ライブ同期とオフライン サポート
- セキュリティ機能
- ACID トランザクション
- マルチリージョンのレプリケーション
- 強力なクエリエンジン



Firestore

Firestore は、高速でサーバーレスのフルマネージド型クラウドネイティブ NoSQL ドキュメント データベースです。モバイルアプリ、ウェブアプリ、IoT アプリのデータの保存、同期、クエリをグローバル規模で簡単に行えます。クライアント ライブラリの形でライブ同期とオフライン サポートが提供され、セキュリティ機能、Firebase および Google Cloud とのインテグレーションにより、真にサーバーレスなアプリの開発が促進されます。

Firestore は ACID トランザクションにも対応しているので、トランザクション内のオペレーションの一部が失敗し、再試行できない場合は、トランザクション全体が失敗します。

さらに自動マルチリージョン レプリケーションと強整合性により、障害が発生した場合でも、データの安全性と可用性が維持されます。Firestore では、パフォーマンスを低下させずに、NoSQL データに対して高度なクエリも実行できます。このため、データを柔軟に構成できます。

# Firestore - 次世代の Datastore

Datastore モード(新しいサーバー プロジェクト):

- Datastore アプリケーションとの互換性
- 強整合性
- エンティティ グループの制限なし

ネイティブ モード(新しいモバイルおよびウェブアプリ):

- 強整合性のストレージ レイヤ
- コレクションとドキュメントのデータモデル
- リアルタイム アップデート
- モバイル クライアント ライブラリとウェブ クライアント ライブラリ

Firestore の実体は次世代の Datastore です。Firestore は Datastore モードで動作できるため、Datastore との下位互換性があります。Datastore モードで Firestore データベースを作成すると、Datastore システムの動作を維持しながら Firestore の改善されたストレージ レイヤにアクセスできます。

これにより、Datastore にあった以下の制限事項がなくなります。

- クエリは結果整合性ではなく、強整合性になりました。
- 25 エンティティ グループまでというトランザクションの制限がなくなります。
- 1 秒あたり 1 個というエンティティ グループへの書き込み制限がなくなります。

ネイティブ モードで動作する Firestore には、次のような新機能があります。

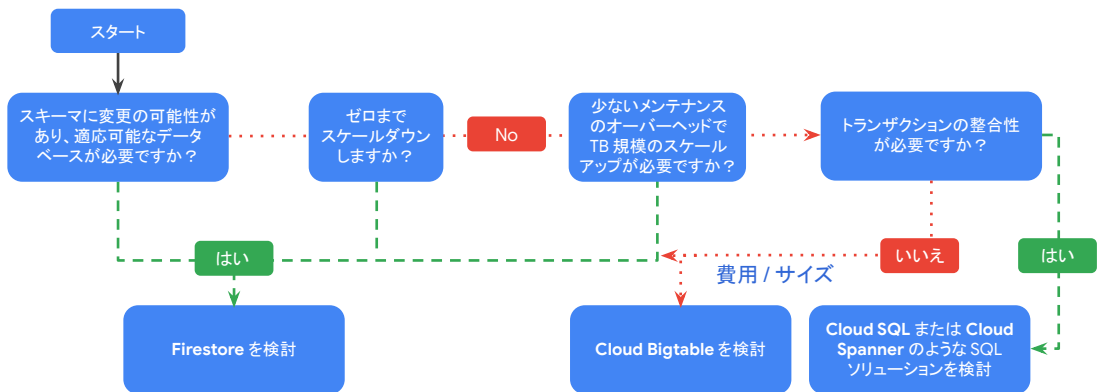
- 新しい強整合性に優れたストレージ レイヤ
- コレクションとドキュメントのデータモデル
- リアルタイム アップデート
- モバイル クライアント ライブラリとウェブ クライアント ライブラリ

Firestore は Datastore と下位互換性がありますが、新しいデータモデル、リアルタイム アップデート、モバイルおよびウェブ クライアント ライブラリの機能には下位互換性がありません。Firestore のすべての新機能にアクセスするには、Firestore をネイティブ モードで使用する必要があります。一般的に、新しいサーバー プロジェクトの場合は Datastore モードで、新しいモバイルアプリやウェブアプリの場合はネイティブ モードで、Firestore を使用すると考えてください。

Firestore は次世代の Datastore であるため、Datastore のすべての API およびクライアント ライブラリと下位互換性があります。既存の Datastore ユーザーには、将来、Firestore へのライブアップグレードが自動的に実施されます。詳細については、この動画のリンク セクションをご覧ください。

[<https://cloud.google.com/datastore/docs/firestore-or-datastore>、  
<https://cloud.google.com/datastore/docs/upgrade-to-firestore>]

## Firestore の選択



最後にこのディシジョン ツリーを使用して、Firestore が自分のデータに適したストレージ サービスかどうかを判断しましょう。

スキーマが変更される可能性があり、適応可能なデータベースが必要である、ゼロまでスケールする必要がある、または、少ないメンテナンスのオーバーヘッドでテラバイト規模までスケールアップする必要がある場合は、Firestore の使用を検討してください。

また、トランザクションの整合性が不要な場合は、費用またはサイズによっては Cloud Bigtable の使用も検討できます。

Cloud Bigtable については、次に説明します。

## アジェンダ

Cloud Storage と Filestore

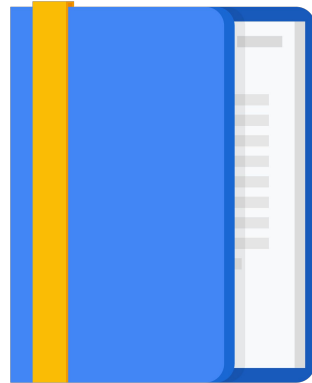
Cloud SQL

Cloud Spanner

Firestore

Cloud Bigtable

Memorystore



トランザクションの整合性が不要でない場合は、Cloud Bigtable の使用を検討することをおすすめします。

# Cloud Bigtable - NoSQL ビッグデータ データベース サービス

- ペタバイト規模
- 10 ミリ秒未満の一貫したレイテンシ
- 優れたスループットを提供するシームレスなスケーラビリティ
- アクセス パターンを学習して適応
- アドテック、フィンテック、IoT に最適
- ML アプリケーション向けのストレージ エンジン
- オープンソースのビッグデータ ツールとのインテグレーションが容易



Cloud  
Bigtable



APACHE  
HBASE

Cloud Bigtable は、レイテンシが非常に低いフルマネージド NoSQL データベースであり、ペタバイト規模のデータに対応できます。シームレスにスケーリングして優れたスループットを提供し、学習によって特定のアクセス パターンに適応します。Cloud Bigtable は、検索、アナリティクス、マップ、Gmail など、Google の多数のコアサービスを支えているのと同じデータベースです。

Cloud Bigtable は低いレイテンシで高い読み取り / 書き込みスループットを実現できるため、IoT、ユーザー分析、財務データ分析など、データ操作向けにもデータ分析向けにも優れたツールです。機械学習アプリケーション用のストレージ エンジンとしても優れています。

Cloud Bigtable は、Hadoop、Cloud Dataflow、Cloud Dataproc などの一般的なビッグデータ ツールと簡単に統合できます。さらに、オープンソースで業界標準の HBase API もサポートしており、導入に関して開発チームに不安を与えません。Cloud Dataflow と Cloud Dataproc については、このコースで後ほど学習します。HBase API の詳細については、この動画のリンク セクションをご覧ください。[<https://hbase.apache.org/>]



## Cloud Bigtable ストレージ モデル

「follows」列ファミリー

行キー	follows			
	gwashtington	jadams	tjefferson	wmckinley
gwashtington		1		
jadams	1		1	
tjefferson	1	1		1
wmckinley			1	

複数のバージョン

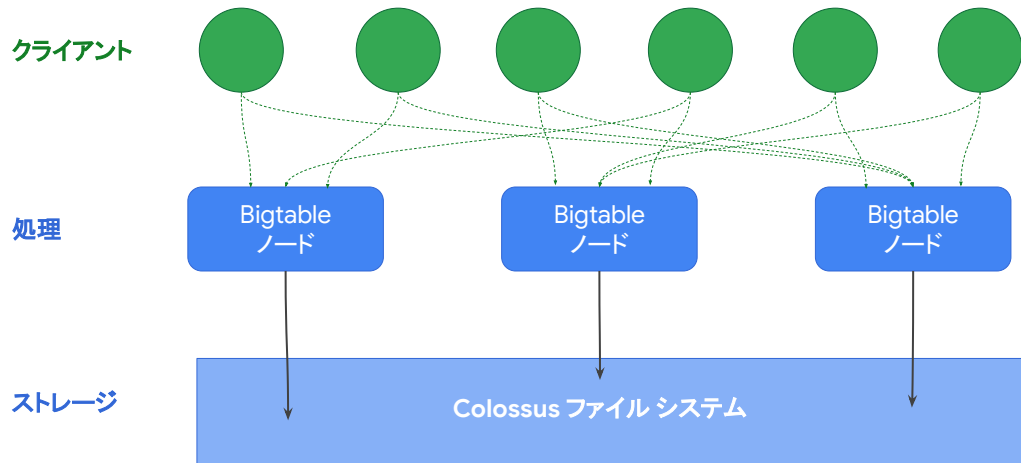
Cloud Bigtable では、大規模にスケーリング可能なテーブル(並べ替えられた Key-Value マップ)にデータが格納されます。テーブルは、行(通常は、各行が単一のエンティティを記述する)と列(各行の個々の値が格納される)で構成されます。各行は単一の行キーでインデックスに登録されており、相互に関連する列は通常、列ファミリーとしてグループ化されます。各列は列ファミリーと列修飾子(特定の列ファミリー内で一意の名前)の組み合わせによって識別されます。

それぞれの行と列が交差する場所には、タイムスタンプの異なる複数のセル(バージョン)を含めることができるため、保存されたデータが時間の経過とともにどのように変更されたかを記録できます。Cloud Bigtable テーブルはスパースです。つまり、データが格納されていないセルが領域を消費することはありません。

ここに示した例は、米国大統領の架空のソーシャル ネットワークです。各大統領は他の大統領の投稿をフォローできます。要点を説明します。

- このテーブルに含まれる列ファミリーは 1 つだけです(follows ファミリー)。このファミリーには複数の列修飾子が格納されています。
- 列修飾子がデータとして使用されています。この設計は、Cloud Bigtable テーブルのスパース性と、データの変更に応じて新しい列修飾子を追加できるという事実を活かしたものです。
- ユーザー名が行キーとして使用されています。ユーザー名がアルファベット全体に均等に分散しているものと仮定すると、データアクセスはテーブル全体にわたって、ある程度均一に分散されます。

## 処理とストレージの分離

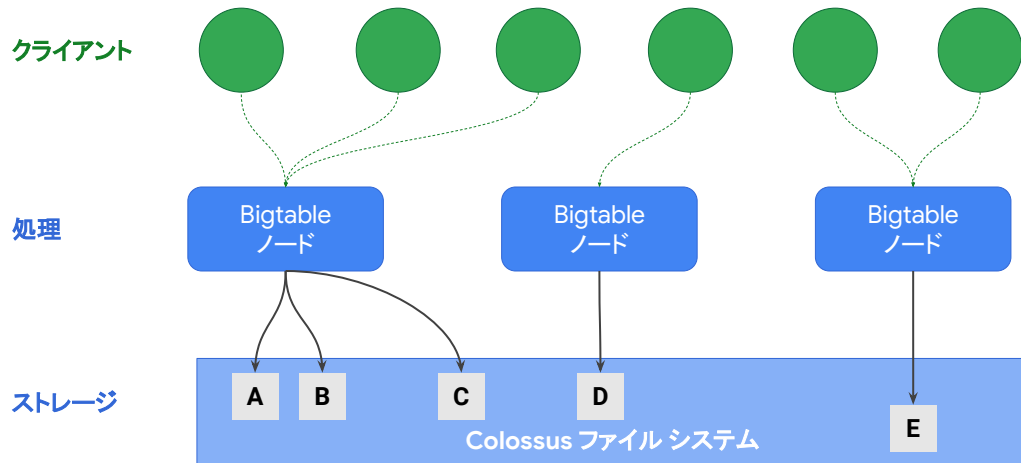


この図は、Cloud Bigtable の全体的なアーキテクチャを簡素化して示しています。ここでわかるとおり、フロントエンド サーバープールとノードによって行われる処理が、ストレージから分離されて扱われます。

Cloud Bigtable テーブルは、連続する行ブロック(タブレット)にシャーディングされるため、クエリの負荷分散が実現されます。HBase API を使ったことがある人向けに言えば、タブレットは HBase リージョンに相当します。

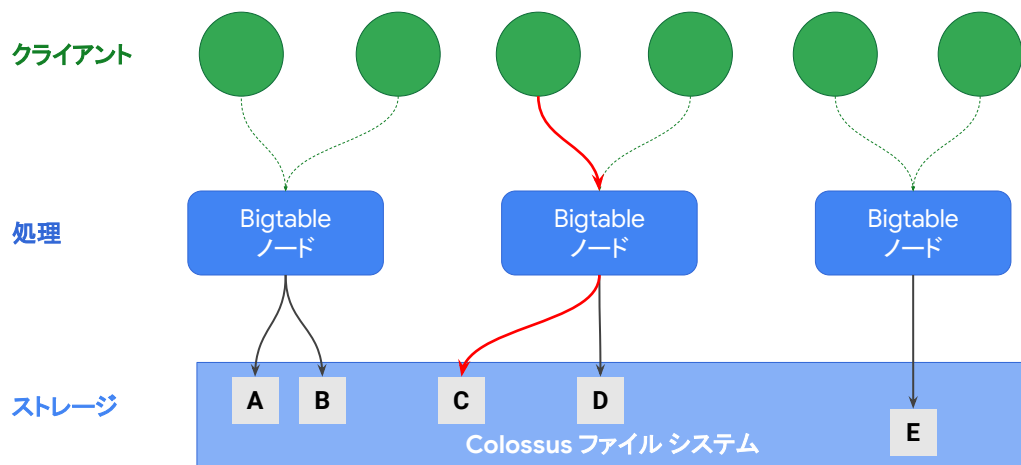
タブレットは、Google のファイル システムである Colossus に SSTable 形式で保存されます。SSTable は、永続的な順序付きの、キーから値への不変マップとなっています。ここで、キーと値はどちらも任意のバイト文字列です。

## アクセス パターンの学習



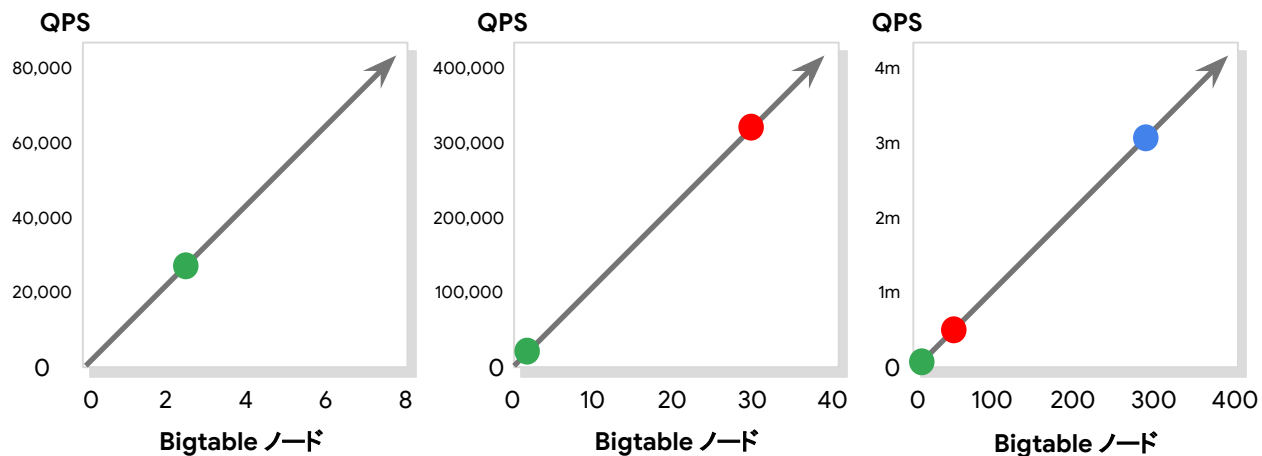
先ほども触れましたが、Cloud Bigtable は学習によって特定のアクセス パターンに適応します。特定の Bigtable ノードがデータの特定のサブセットに繰り返しアクセスしていると仮定しましょう。

## データを移動せずに再調整



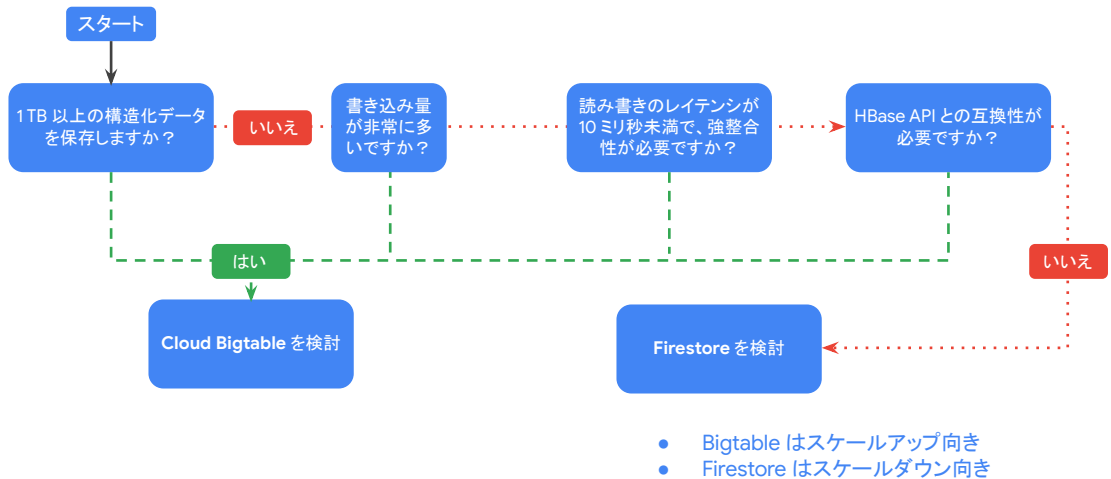
そのような場合は、この図のように、Cloud Bigtable がインデックスを更新して、他のノードがこのワークロードを均等に分散できるようにします。

## スループットが直線的にスケーリング



スループットが直線的にスケーリングするため、ノードを 1 つ追加することによってスループットのパフォーマンスが直線的にスケーリングします。このスケーリングが数百のノードまで続きます。

## Cloud Bigtable の選択



まとめると、1 TB 以上の構造化データを保存する必要がある、書き込み量が非常に多い、読み書きレイテンシが 10 ミリ秒未満で強整合性が必要である、または HBase API と互換性のあるストレージ サービスが必要な場合は、Cloud Bigtable の使用を検討してください。

これらの要件がなく、スケールダウンを適切に行えるストレージ サービスを探している場合は、Firestore の使用を検討してください。

スケーリングについて言えば、作成可能な最小の Cloud Bigtable クラスタは 3 ノードで構成され、1 秒あたり 30,000 回のオペレーションを処理できます。これらのノードがオペレーションを受け付ける状態にある場合、アプリケーションがそれらを使用しているかどうかに関係なく、料金が発生することに注意してください。

## アジェンダ

Cloud Storage と Filestore

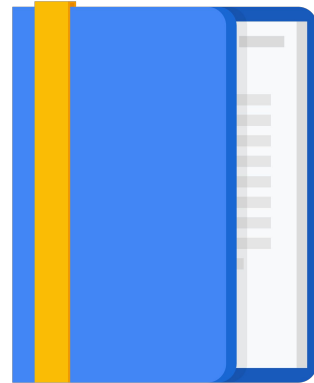
Cloud SQL

Cloud Spanner

Firestore

Cloud Bigtable

Memorystore



Memorystore の概要を簡単に説明します。

# Memorystore - フルマネージド Redis サービス

- インメモリ データストア サービス
- アプリの構築に専念
- 高可用性、フェイルオーバー、パッチ適用、モニタリング
- ミリ秒未満のレイテンシ
- 最大 300 GB のインスタンス
- 12 Gbps のネットワーク スループット
- 簡単なリフト & シフト



Memorystore

Google が管理するスケーラブルで安全な高可用性インフラストラクチャ上に構築された Memorystore for Redis は、フルマネージドのインメモリ データストア サービスです。スケーラビリティと可用性が高くセキュアなこの Redis サービスを利用すると、Google Cloud 上のアプリケーションで優れたパフォーマンスを実現できます。複雑な Redis デプロイを管理する必要もありません。これにより、コードの記述に多くの時間を費やせるようになるので、優れたアプリの構築に専念できます。

Memorystore は、高可用性の実現、フェイルオーバー、パッチ適用、モニタリングなど、複雑なタスクを自動化することもできます。高可用性インスタンスが 2 つのゾーンに複製され、99.9% の可用性 SLA を提供します。

アプリケーションに必要なミリ秒未満のレイテンシやスループットを簡単に実現できます。最も低いティアと最も小さいサイズから開始し、アプリケーションの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できます。Memorystore は、最大 300 GB のインスタンスと 12 Gbps のネットワーク スループットをサポートします。

Memorystore for Redis には Redis プロトコルとの完全な互換性があるので、コードを変更しなくても、インポートとエクスポートの機能を使用してアプリケーションをオープンソースの Redis から Memorystore にリフト & シフトすることができます。既存のツールやクライアント ライブラリがすべてそのまま機能するので、新しいツールを習得する必要はありません。



# まとめ

---

## ストレージとデータベース のサービス



このモジュールでは、Google Cloud が提供するさまざまなストレージ サービスとデータベース サービスについて説明しました。具体的には、Cloud Storage (フルマネージド オブジェクト ストア)、Cloud SQL (フルマネージド MySQL および PostgreSQL データベース サービス)、Cloud Spanner (トランザクション整合性、グローバルなスケーリング、高可用性を備えたリレーショナル データベース サービス)、Firestore (フルマネージド NoSQL ドキュメント データベース)、Cloud Bigtable (フルマネージド NoSQL ワイドカラム型データベース)、Memorystore (Redis 対応フルマネージド インメモリ データストア サービス) について学びました。

ここでの目標は、インフラストラクチャの観点から、利用できるサービスと、それらが使用される状況について理解することでした。完全なデータ戦略を策定することはこのコースの範囲外ですが、Google Cloud でのデータ エンジニアリングと機械学習に関するコースの中でデータ戦略を取り上げています。