

Google Cloud でクラウド二刀流エンジニアを目指すための「早わかり集中技術講座」

～ データベース編 ～

大久保 順 | データ・プラットフォーム事業開発部長
安井 通隆 | カスタマーエンジニア

2022 年 7 月
グーグル・クラウド・ジャパン合同会社

目次

クラウドニ刀流エンジニアを目指す	01
マネージド RDBMS	02
クラウド ネイティブ データベース	03
NoSQL データベース	04
まとめ	05

01

クラウドニ刀流エンジニアを目指す

本セッションのねらい

Amazon Web Services や Microsoft Azure の利用経験があるデータベース/ インフラエンジニアを対象に、Google Cloud でのデータベースの選び方の指針をご紹介します。

Google Cloud を利用して二刀流エンジニアを目指しましょう！



02

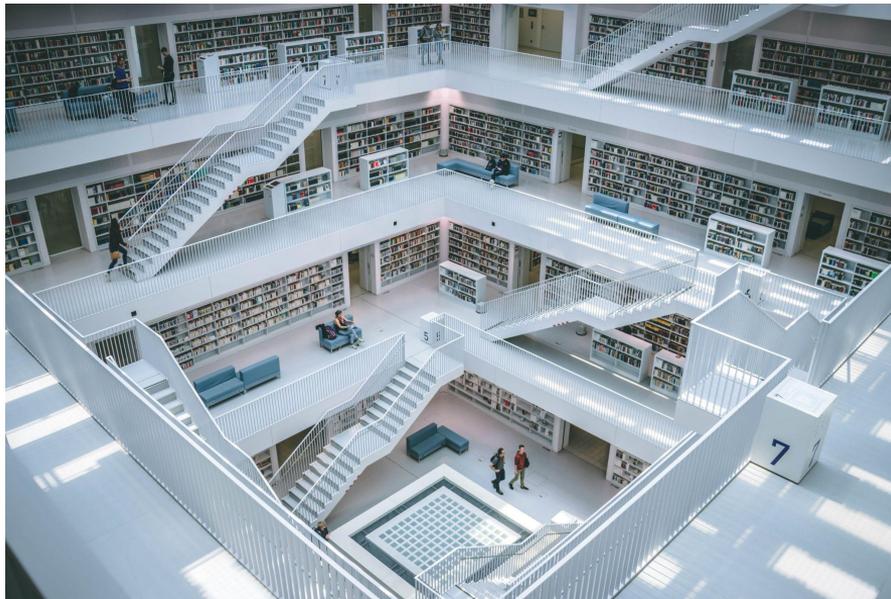
マネージド RDBMS

このセクションでの取り扱うデータベースについて

OSS や商用ソフトウェアとして提供されている **RDBMS** のマネージドサービス。具体的なソフトウェアの例としては以下の通りです。

例) MySQL, PostgreSQL, Oracle, SQL Server

オンプレミスや IaaS(Amazon EC2, Azure VM, Google Compute Engine など)を使って運用されているこれらの**既存アプリケーションの移行先**としてご利用が可能なマネージドサービス。性能向上は多くの場合、スケールアップ・ダウンで行い、一部スケールアウトが可能



主なマネージド RDBMS サービス

クラウドサービス名	AWS	Azure	Google Cloud
サービス名	Amazon RDS / Amazon Aurora	Azure Database / Azure SQL Database	Cloud SQL / AlloyDB(Preview)
サポートデータベースエンジン	MySQL / PostgreSQL / Oracle / SQL Server / MariaDB	MySQL / PostgreSQL / SQL Server / MariaDB	MySQL / PostgreSQL / SQL Server

マネージドの機能性

高可用性構成

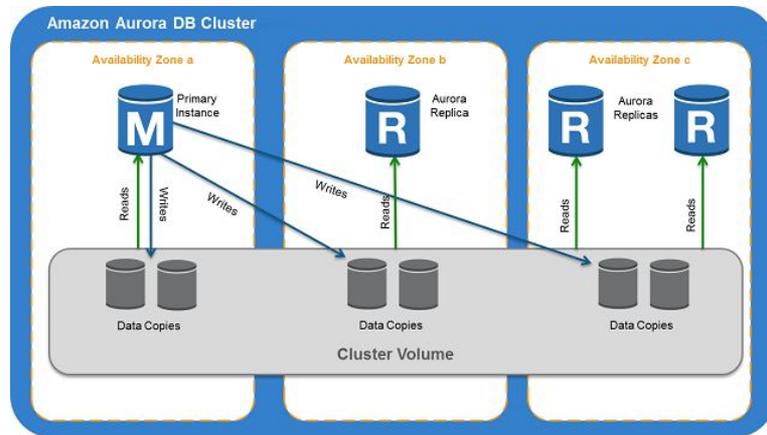
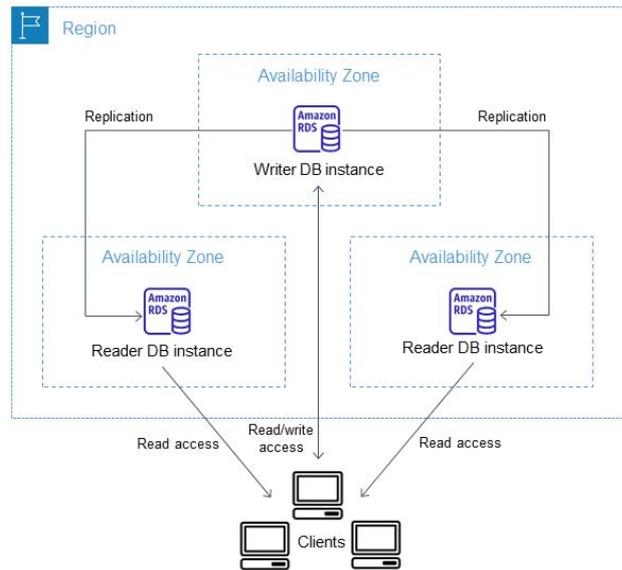
AWS

Amazon RDS の マルチ AZ

- 1つのスタンバイを備えたマルチAZ(SQL Server 以外)
- 2つの読み取り可能なスタンバイを備えたマルチAZ(MySQLとPostgreSQLのみ)

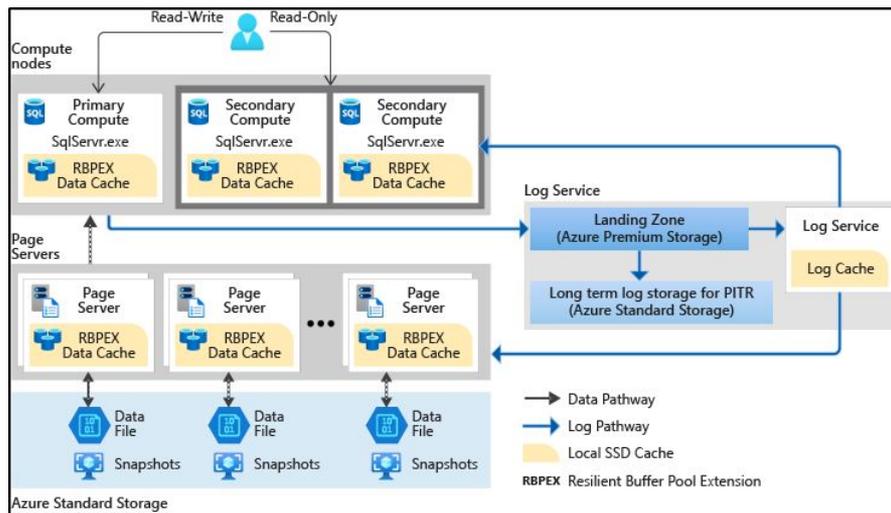
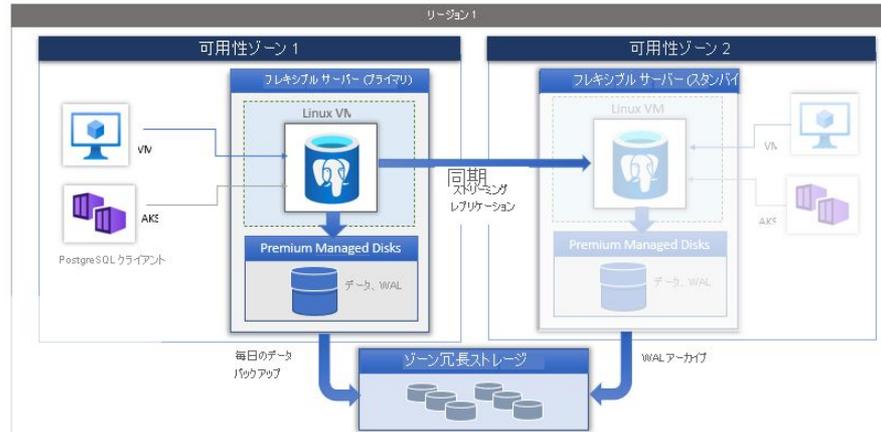
Amazon Aurora(MySQL/PostgreSQL)

- 3つのAZに分散したストレージとそれを参照するインスタンスによるマルチAZ
- 一つの読み書きインスタンスと最大5台までの読み取りレプリカ



Azure

- シングルサーバー
 - ホットスタンバイなしのHA
- フレキシブルサーバー
 - ホットスタンバイなしのHA
 - ホットスタンバイを使用した**同じゾーンかゾーン冗長HA**
- Hyperscale (SQL Server)
 - コンピューティングノード、ページサーバ、ログサービス、ストレージで構成されたアーキテクチャ
 - 最大 4 読み取りスケールアウト



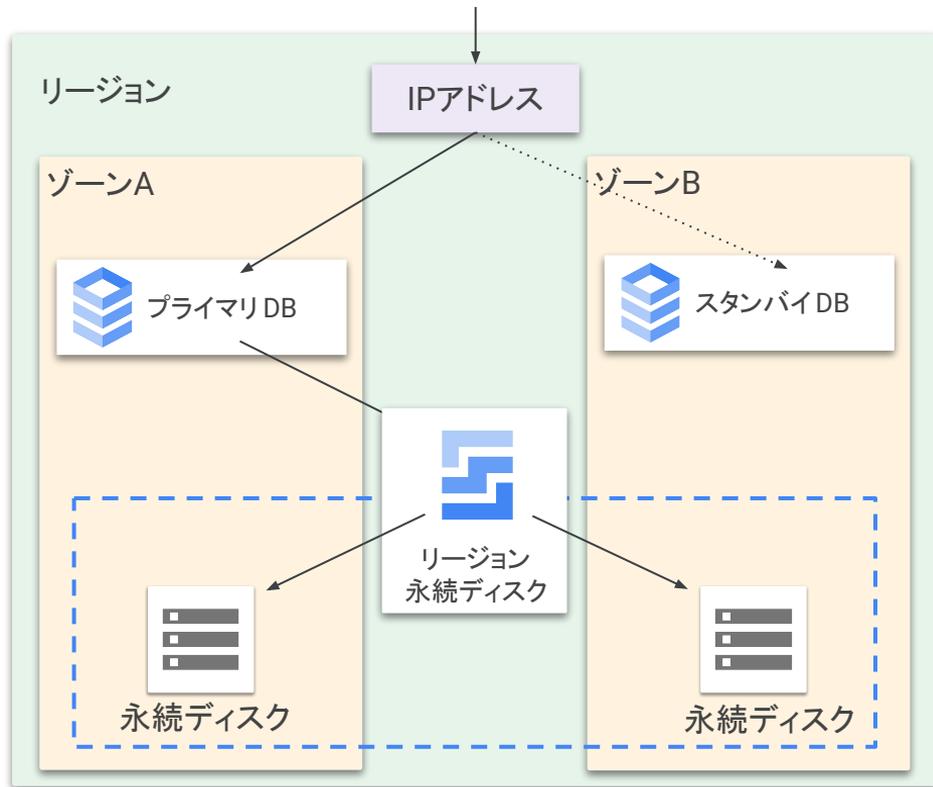
Cloud SQL の HA 構成

リージョン永続ディスクによる同期

インスタンスは、HA 構成が可能で、プライマリとスタンバイの 2 つのインスタンスから構成される。

プライマリ DB への書き込みは、リージョン永続ディスク (Regional PD) を通じ、スタンバイ DB が配置されたゾーンの永続ディスクへも **同期書き込み** が行われる。

これによりレプリケーション遅延がなくなり、フェイルオーバーにかかる時間を大幅に短縮。

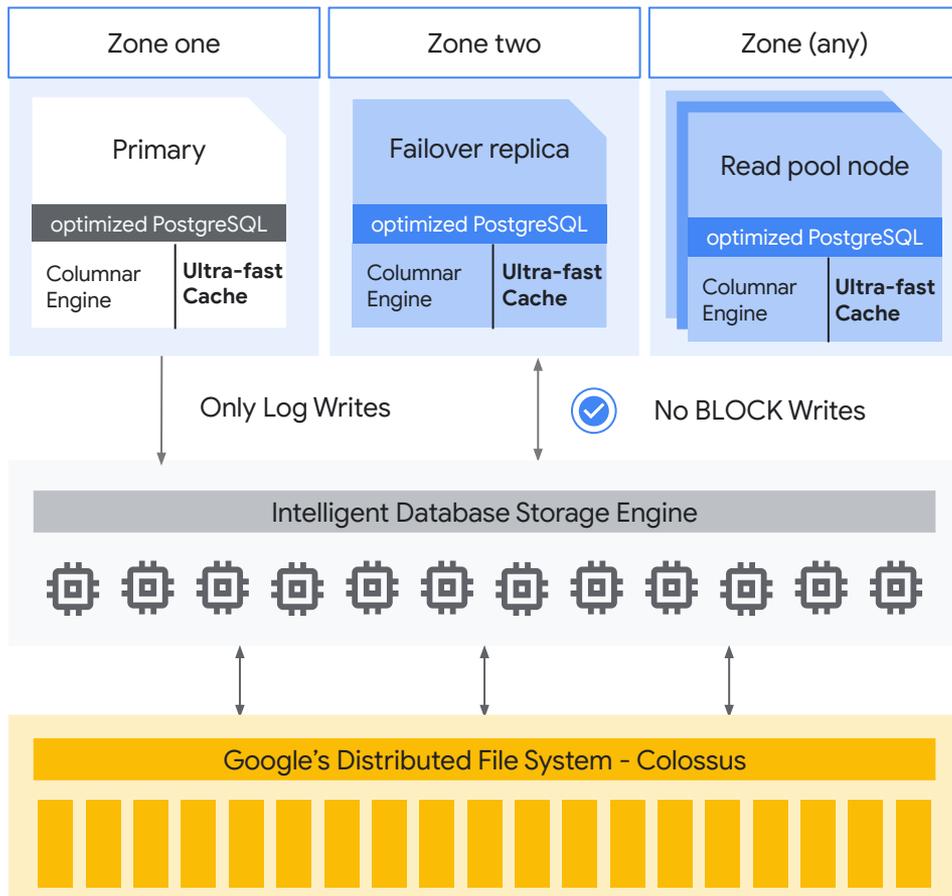


AlloyDB の高可用性

SQL を処理するコンピュー部分とストレージ部分を分離しています。

リージョナルストレージは3つのゾーンに分散して配置されています。

書き込みはプライマリインスタンスから行い、同じストレージを参照する最大 20 台のレプリカが利用可能です。同じストレージを参照することで、小さいラグは小さくすることが可能



モニタリング

各サービスのモニタリング機能

	AWS	Azure	Google Cloud
リソースモニタリング	CloudWatch	Azure メトリクス	Cloud Monitoring
パフォーマンス モニタリング	Performance Insights	Query Performance Insights	Query Insights

Cloud SQL の特徴的な機能

BigQuery から Cloud SQL の連携クエリ

BigQuery と Cloud SQL の連携機能では、データコピーを行うことなく、Cloud SQL のデータを BigQuery から利用する機能です。

Cloud SQL の MySQL と PostgreSQL で利用可能です。



Cloud SQL

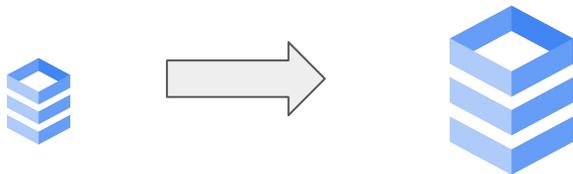


BigQuery

```
SELECT c.customer_id, c.name, rq.first_order_date
FROM mydataset.customers AS c
LEFT OUTER JOIN EXTERNAL_QUERY(
  'us.connection_id',
  ''SELECT customer_id, MIN(order_date) AS first_order_date
FROM orders
GROUP BY customer_id''') AS rq ON rq.customer_id = c.customer_id
GROUP BY c.customer_id, c.name, rq.first_order_date;
```

クエリー例

カスタムマシンタイプによるパフォーマンスの調整



簡単な操作でマシンスペックを調整

- コンソールや API などによる vCPU とメモリの細かいスケールアップ / ダウンが可能
- 主なユースケース
 - サービスの成長に合わせてスケールアップ
 - あまり利用しない週末はスケールダウン
- 最大サイズ
 - CPU 96 コア
 - RAM 最大 624 GB

マシンタイプ ▲

マシンタイプ

プリセットを選択するか、独自にカスタマイズします。パフォーマンスを改善するには、最大のテーブルを格納するのに十分なメモリを持つマシンタイプを選択します。

ハイメモリ ▼

4 vCPU、26 GB

8 vCPU、52 GB

16 vCPU、104 GB

カスタム

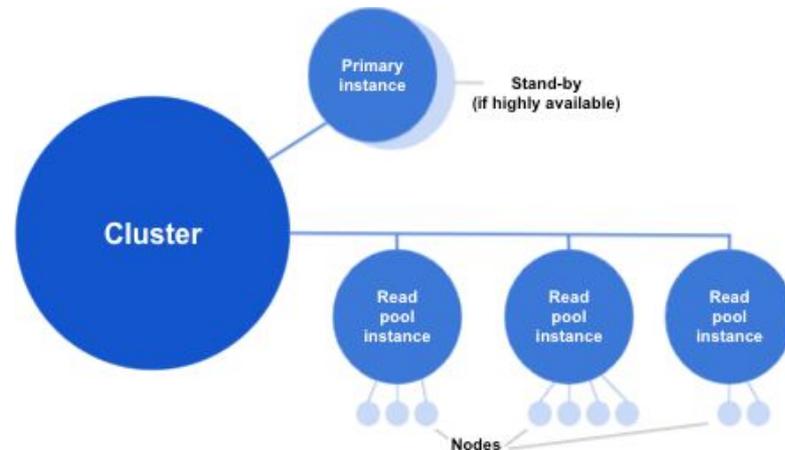
コア数 vCPU 1~96

メモリ GB 86.5~624

AlloyDB の特徴的な機能

Read Pool(読み取りプール)

- 読み取りを行うノードをまとめて管理するためのリソース
 - クラスタ内に複数の読み取りプールを作成可能
 - 読み取りプール内に複数のノードを作成
 - 読み取りプールごとに単一のエンドポイント
- 用途に応じて複数の読み取りプールを作成可能
 - 読み取りプールごとにマシンタイプやフラグを設定

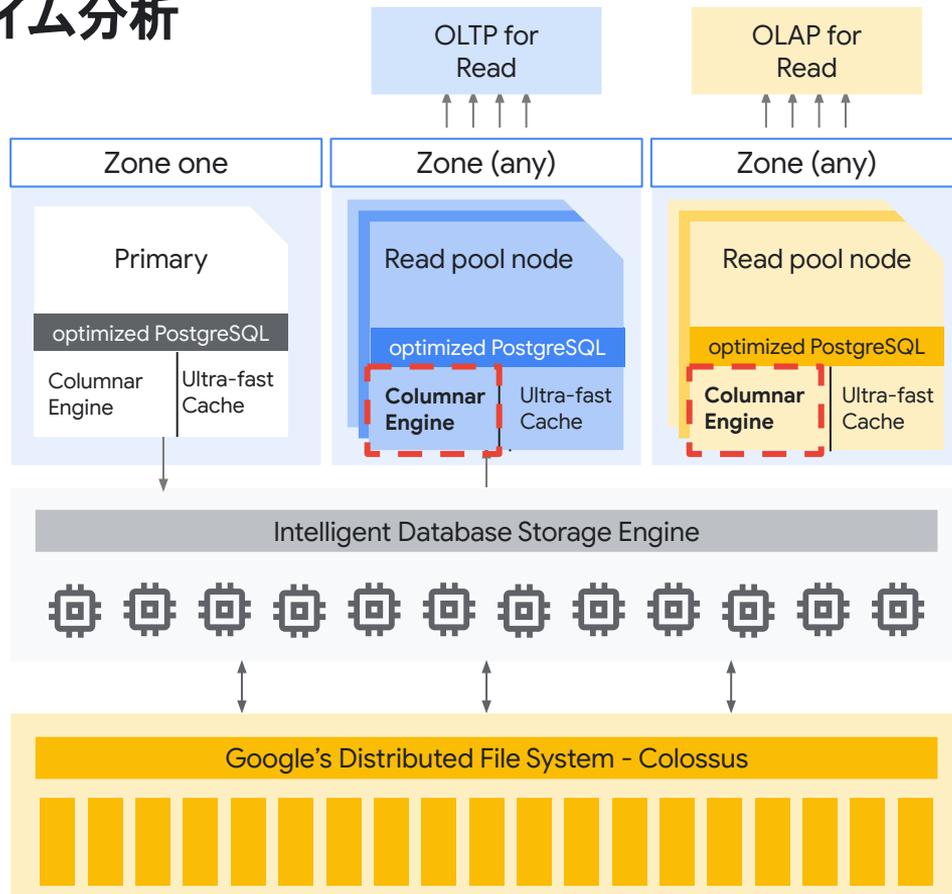


Instances in your cluster [+ ADD READ POOL](#)

alloydb-test	read-pool	read-pool-olap
Primary instance ?	Read pool instance ?	Read pool instance ?
Status ✓ Ready	Status ✓ Ready	Status ✓ Ready
High availability Highly available (multi-zone)	High availability Highly available (multiple read nodes)	High availability No
Location us-central1	Nodes 2	Nodes 1
Machine type 8 vCPU, 64 GB	Location us-central1	Location us-central1
Private IP 10. []	Machine type 8 vCPU, 64 GB	Machine type 4 vCPU, 32 GB
Flags alloydb.enable_pgaudit on	Private IP 10. []	Private IP 10. []
Flags alloydb.enable_pgaudit on	Flags No flags set	Flags No flags set
EDIT PRIMARY	EDIT READ POOL	EDIT READ POOL

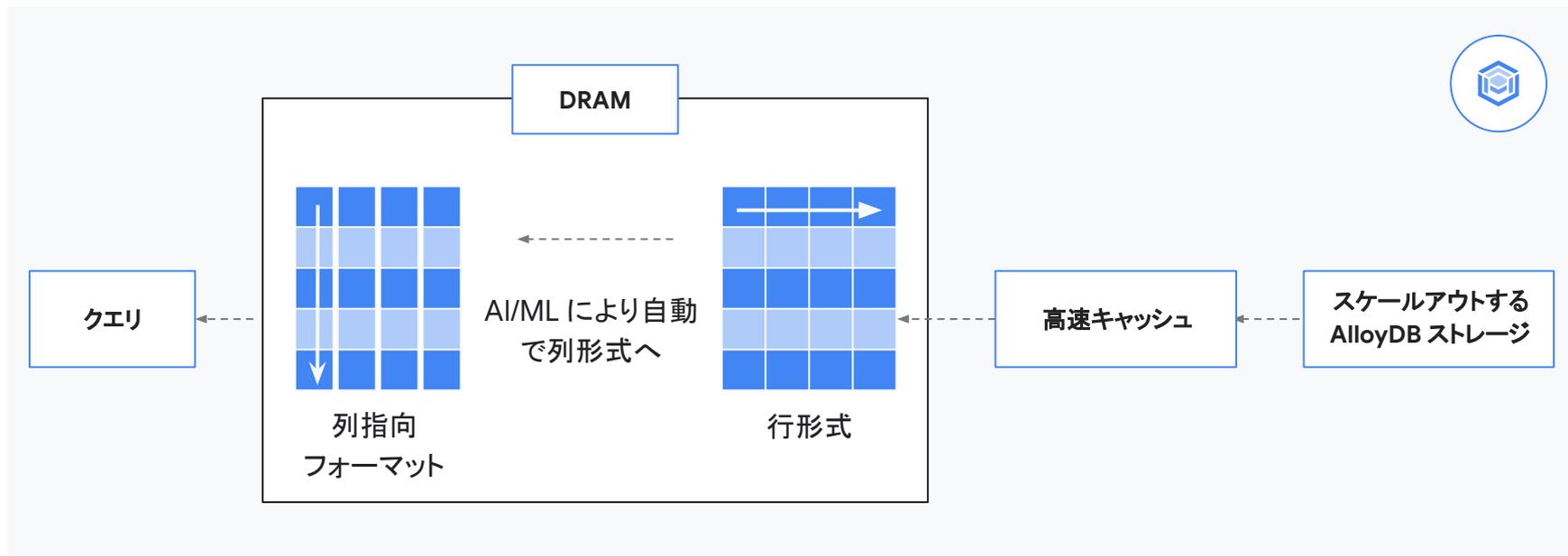
カラム型エンジンを活用したリアルタイム分析

- 分析クエリを高速化するための独自機能
 - 列指向形式に編成されたキャッシュ
 - 列指向形式のデータに対応したプランナー、実行エンジン
- 1ノードで分析処理を実行し、低レイテンシーな応答を期待可能
 - 1ノードに載せるメモリ量を調整
- 主なユースケース
 - e-commerce などのレコメンデーション
 - リアルタイムランキング(ゲームのリーダーボード、人気商品のランキング etc)
 - 最新データを使ったアドホッククエリ



高速かつ予測可能なパフォーマンス

ワークロードに基づき、行ベースおよび列ベース形式に動的にデータを編成
複数レイヤーのキャッシュにより、費用対効果の高いパフォーマンスを実現



03

クラウド ネイティブ データベース

このセクションで扱うデータベースについて

クラウド環境での稼働を前提として、設計及び開発されたデータベース。

これらのサービスはクラウドで利用できるマルチゾーンを活用した**高可用性**や**スケーラビリティ**を備えている。スケーラビリティについては特に書き込み性能について、**SQL**でアクセス可能で**スケールアウトが可能なもの**を中心に扱います。

既存のアプリケーションの移行用よりは、クラウド上での動作を前提とした**新規アプリケーション**で利用されることが多い



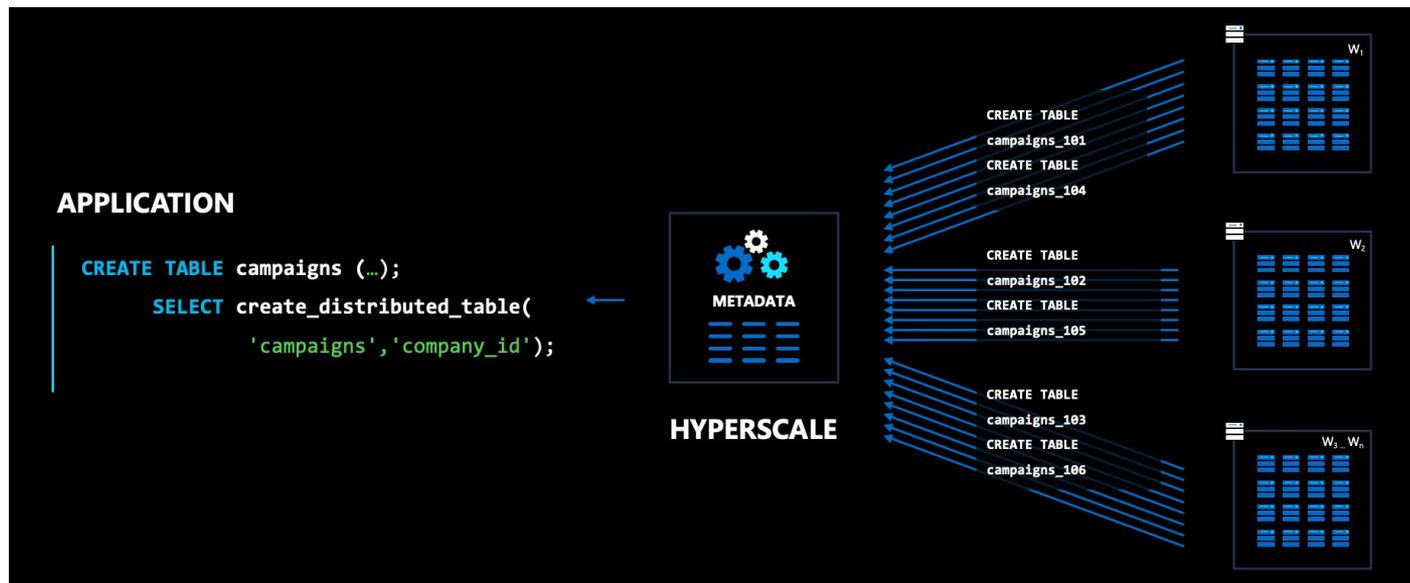
Amazon DynamoDB

特徴: キーバリュー型の NoSQL、サーバーレス、アベイラビリティ
ゾーン間で自動的にレプリケーション、操作は DynamoDB API あ
るいは PartiQL のサブセットによるアクセスも可能



Azure Database for PostgreSQL Hyperscale(Citus)

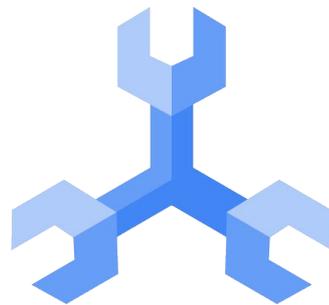
特徴: 分散テーブルによるスケールアウト可能な PostgreSQL 拡張 Citus に基づいたサービス



Cloud Spanner

特徴: リレーショナルなデータベースのようにテーブル構造に対して SQL での問い合わせとトランザクションをサポートし、**同時に** NoSQL のような水平スケーラビリティを持つ

- SQL は Google 標準 SQL (拡張機能を含む ANSI 2011) と PostgreSQL 互換をサポート
- トランザクションは行、およびテーブルにまたがるアトミックな読み取りと書き込みをサポート
- リクエストの負荷とデータサイズに基づいて、自動的にシャーディングを行うことでパフォーマンスを最適化します



Cloud Spanner

可用性

各サービスの可用性についての SLA

	単一リージョン構成	複数リージョン構成
DynamoDB	99.99 %	99.999 %
Hyperscale(Citus)	99.95 %	-
Cloud Spanner	99.99 %	99.999 %

料金モデル

料金モデルの比較

	料金モデル(主なもの)
DynamoDB	オンデマンド型キャパシティモード / プロビジョンド型キャパシティモード
Hyperscale(Citus)	ノードのサイズと数
Cloud Spanner	コンピューティング容量

Cloud Spanner の特徴的な機能

Key Visualizer はテーブル アクセスの偏りを可視化する

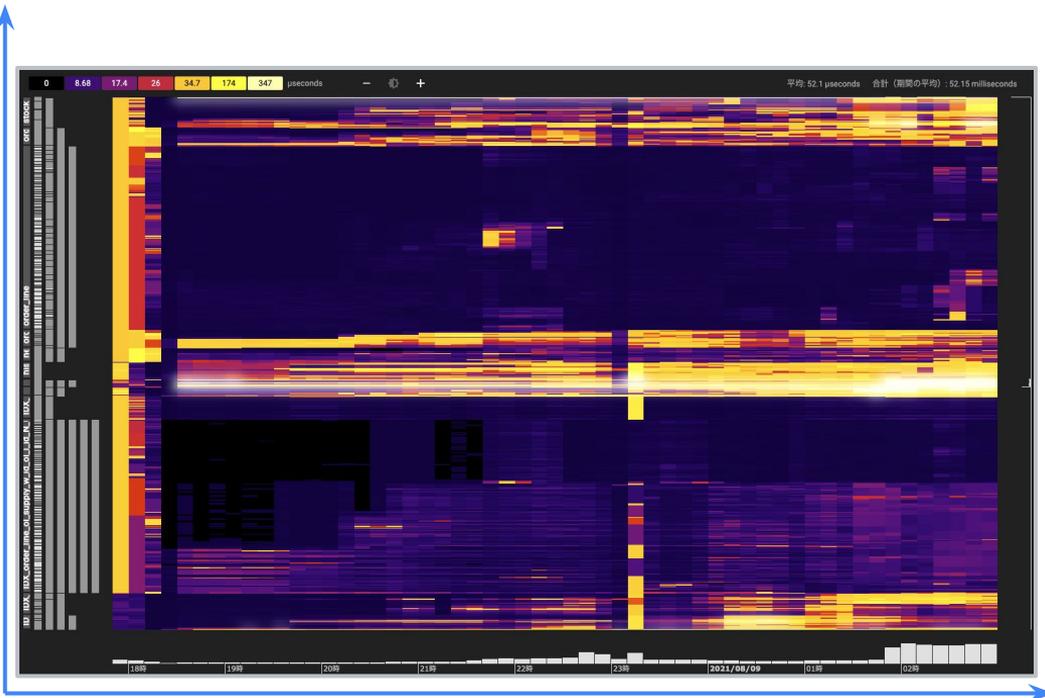
Key Visualizer は、データベースを構成する全テーブルの各行範囲が、どのようにアクセスされているかを可視化するツール。）

テーブルに対して、偏ったアクセスがないかを可視化することができる。

以下の項目について、色の明らいたところほど値が大きく、暗いところほど値が小さいことを意味する。

- 消費された CPU 時間
- 読み込まれた行数
- 書き込まれた行数
- など

テーブル全体のキー空間
(インデックス含む)



時間

関連Docs: [Key Visualizer の概要](#)

<https://cloud.google.com/spanner/docs/key-visualizer?hl=ja>

Cloud Spanner の構成



(シングル) リージョン構成

日本国内には東京と大阪の2リージョンから選択。もちろんその他海外のリージョンも選択可能。可用性は 99.99%^{*1}

マルチリージョン構成

複数のリージョンを同時に利用することにより、可用性を 99.999%^{*1} に向上させ、またリージョン障害や災害など地理位置に依存した障害にも耐えることができる。

日本では、東京及び大阪(及びソウル^{*2})を同時に利用することで実現

*1) SLA として提供される月間稼働率の数値。詳細はこちらに記載 <https://cloud.google.com/spanner/sla>

*2) ソウルには実際のデータを保存しないが、リージョン障害発生時においてもデータの整合性担保ができるよう通信を行う。

04

NoSQL データベース

このセクションでの取り扱うデータベースについて

NoSQL データベースは広く非リレーショナル データベース全般 (= RDBMS 以外)を指しますが、その種類は多岐に渡ります。

本セクションでは、用途が近いと考えられるデータベースの種類ごとに分類を行っていますが、これらの分類はあくまでも一例であるのご理解下さい。

(他の視点や分類を否定するものではありません)

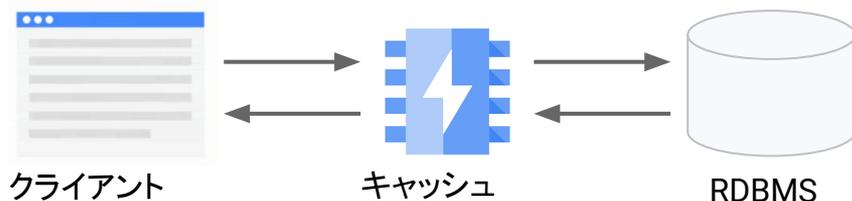


インメモリー データベース

インメモリー データベースの主な用途

利点

- キャッシュとして配置することでレイテンシーを低減する

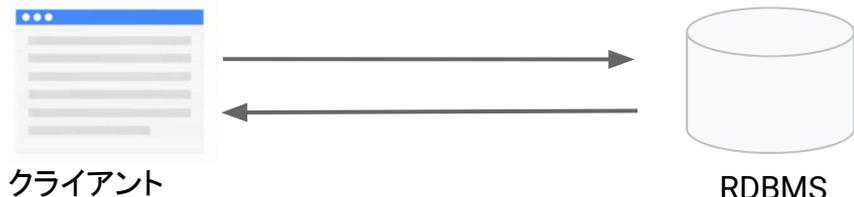


主なユースケース

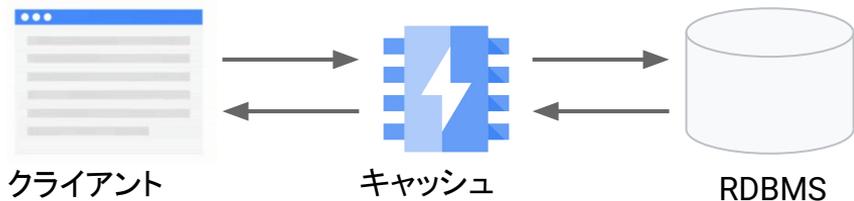
- 読み取り負荷の高いアプリケーション
- リアルタイム性の求められるアプリケーション
- 読み取り速度の向上

オープンソース データベースの **Redis** と **Memcached** がこの用途で多く利用されている

インメモリー キャッシュによるレイテンシーの軽減イメージ



- ストレージアクセスだと遅い
- 大量のリクエストがくると、過負荷になる可能性



- 常に高速な応答が期待可能
- RDBMS への負荷を巻き取る

インメモリ キャッシュがない場合

- ストレージアクセスが発生した場合、低速なレスポンスになる可能性
(ディスクからの読み込みはメモリからの読み込みと比較すると圧倒的に遅い)
- RDBMS への読み取り負荷が増大し、スループットが頭打ちになる可能性

インメモリ キャッシュがある場合

- 高速な応答が必要なデータをキャッシュにおき、ミリ秒未満の応答を実現
- RDBMS と対照的にメモリに最適化されたクエリの実行になっており速い応答が期待できる(RDBMS ではディスクにデータがある前提で操作が行われる)

キャッシュ以外のユースケース

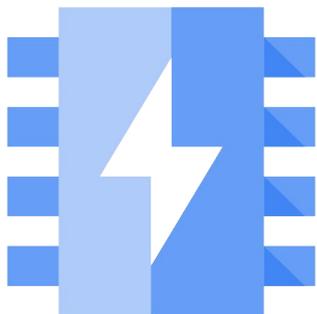
- セッション管理
 - ログイン情報の管理
- リーダーボード
 - ゲーム内の順位表、ランキング
 - Redis が持つソート済みセット型 (Sorted Set) を使うと簡単に実装できるため
- pub/sub
 - チャットアプリケーション
 - センサーデータの管理

即時の永続化が必要ない(万一揮発してもクリティカルではない)
他に永続化されたデータから再構成できる性質の
データを扱うことに向いているデータストア

主なインメモリー データベース サービス

	AWS	Microsoft Azure	Google Cloud
サービス名	ElastiCache	Azure Cache for Redis	Memorystore
対応エンジン	Redis (5.0.6, 6.0, 6.2) Memcached (1.4.x, 1.5.x, 1.6.x)	Redis (4.0.x, 6.0.x)	Redis (6.x, 5.0, 4.0, 3.2) Memcached (1.5.16)
SLA	99.5% (Single-AZ) 99.9% (Cross/Multi-AZ)	なし~99.999% (サービスティアによる)	99.9%
料金体系	インスタンス料金 (CPU, Memory, 台数)	サービスティア (Basic / Standard / Premium / Enterprise / Enterprise Flash) キャッシュサイズ ゾーン冗長 geo レプリケーション	サービスティア (スタンドアロン/ 高 可用性) プロビジョニングされた容量 レプリカ数

Cloud Memorystore



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

Redis エンジン版の機能性比較

	AWS ElastiCache for Redis	Microsoft Azure Azure Cache for Redis	Google Cloud Memorystore for Redis
スケールアウト方式	リードレプリカ Redis Cluster	Redis Cluster	リードレプリカ
高可用性	レプリカへのフェイルオーバー Multi-AZ 構成	レプリカへのフェイルオーバー ゾーン冗長 geo レプリケーション	レプリカへのフェイルオーバー ゾーン冗長
セキュリティ	暗号化 (in transit, at rest) IAM による認証	暗号化 (in transit, at rest)	暗号化 (in transit, at rest) IAM による認証 Cloud Audit Logging
永続化	RDB / AOF への書き出し	RDB / AOF への書き出し	RDB スナップショット (Preview)
運用監視	CloudWatch	Azure Monitor	Cloud Monitoring

ドキュメント データベース

ドキュメント データベースの主な用途

特徴

- スキーマレスでデータ構造が自由なので、ビジネスの変化に応じて柔軟にデータを格納することが可能
- JSON や XML などのオブジェクト(=ドキュメント)を扱いやすい

主なユースケース

- ログの格納・分析
- 機能拡張・変更が頻繁に発生する Web アプリケーション

主なドキュメント データベース サービス

	AWS	Microsoft Azure	Google Cloud
サービス名	DocumentDB	Cosmos DB (注) マルチモデルデータベース サービス	Firestore
API	MongoDB 互換	MongoDB 互換 (注) 他に SQL/Core, Cassandra, Gremlin, Table など様々な API から DB 作成時に選択	Firestore ネイティブモード Datastore モード
SLA	99.9%	99.99~99.999% (項目・構成ごとに 詳細な SLA 定義)	99.99% (Regional) 99.999% (Multi-Region)
料金体系	インスタンス料金 (CPU, Memory, 台数) ストレージ (容量, I/O) 等	データベース操作 (プロビジョニング 済み, サーバーレス) ストレージ (容量) 等	データベース操作 (read / write / delete) ストレージ (容量) 等

Firestore (ネイティブ モード / Datastore モード)

NoSQL の一種であるドキュメント ストアを API コール課金のサーバーレス DB として提供。Firestore との親和性が高く、モバイルアプリから直接接続が可能であり、DB への意識はおろかサーバーサイド要らずで開発を高速化。

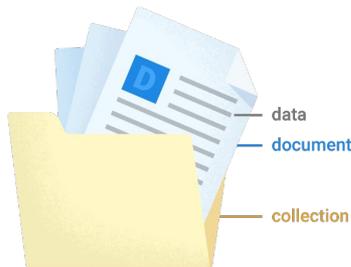


Firestore 由来の **Realtime Database**
Google Cloud の **Cloud Datastore**
(App Engine の)

この両者をベースに、Google Cloud のインフラ上に構築された、新しいサーバーレス データベース。
ネイティブ モードと
互換性維持の **Datastore モード**がある。

Cloud Firestore (ネイティブモード) のデータモデル

- データの格納単位は「ドキュメント」
- 複数のドキュメントをまとめた「コレクション」



コレクション — users

ドキュメント — aloveface

データ (フィールド) {
first : "Ada"
last : "Lovelace"
born : 1815

ドキュメント — aturing

データ (フィールド) {
first : "Alan"
last : "Turing"
born : 1912

/ > users > aloveface		
最上位	users	aloveface
+ コレクションを開始	+ ADD DOCUMENT	+ コレクションを開始
⋮ users >	⋮ aloveface >	+ フィールドを追加
	aturing	born: 1912 first: "Ada" last: "Lovelace"

キーバリューストアと その他の NoSQL データベース

キーバリューストアの主な用途

特徴

- キーとバリュー (値) のペアとしてデータを格納
- リレーショナルモデルのようにスキーマ構造が固まっていないため、半構造化・非構造化データも扱いやすい
- シンプルな構造ゆえに、スケーラビリティやレイテンシーに優れる
- 複数テーブル・レコード間のデータ一貫性を担保するトランザクション処理は得意ではない

主なユースケース

- 同時アクセスが多いモバイルやWeb、IoT アプリケーション
- 業務要件の変化などに伴い、スキーマの変更が頻繁に発生するユースケース

主なキーバリューストア

	AWS	Microsoft Azure	Google Cloud
サービス名	DynamoDB	Cosmos DB (注) マルチモデルデータベース サービス	Cloud Bigtable
API	DynamoDB API PartiQL	Table API (注) 他に SQL/Core, Cassandra, Gremlin, MongoDB など様々な API からDB 作成時に選択	Cloud Bigtable API Apache HBase 互換 API
SLA	99.99% (Standard) 99.999% (Global Tables)	99.99~99.999% (項目・構成ごとに 詳細なSLA 定義)	99.9% (Single Cluster) 99.99% (Multi-Cluster) 99.999% (Multi-Cluster with 3 or more regions)
料金体系	プロビジョン済み容量(r/w) オンデマンドモード(r/w) ストレージ容量等	データベース操作(プロビジョニング 済み, サーバーレス) ストレージ(容量)等	ノード数 ストレージ容量等

Cloud Bigtable

NoSQL の一種である**ワイド カラムストア**を提供

低レイテンシな読み書きを、高スループットで提供可能な、スケーラブルなキーバリューストア

データ分析系の製品との連携性も高く
また Apache HBase との互換性もある



—列の値が入っていたり入っていなかったりする、スカスカのテーブルのこと

スパースなテーブルを効率よく扱うことができるため、**例えば数千や数万列からなるテーブルにデータを格納して処理することが可能**

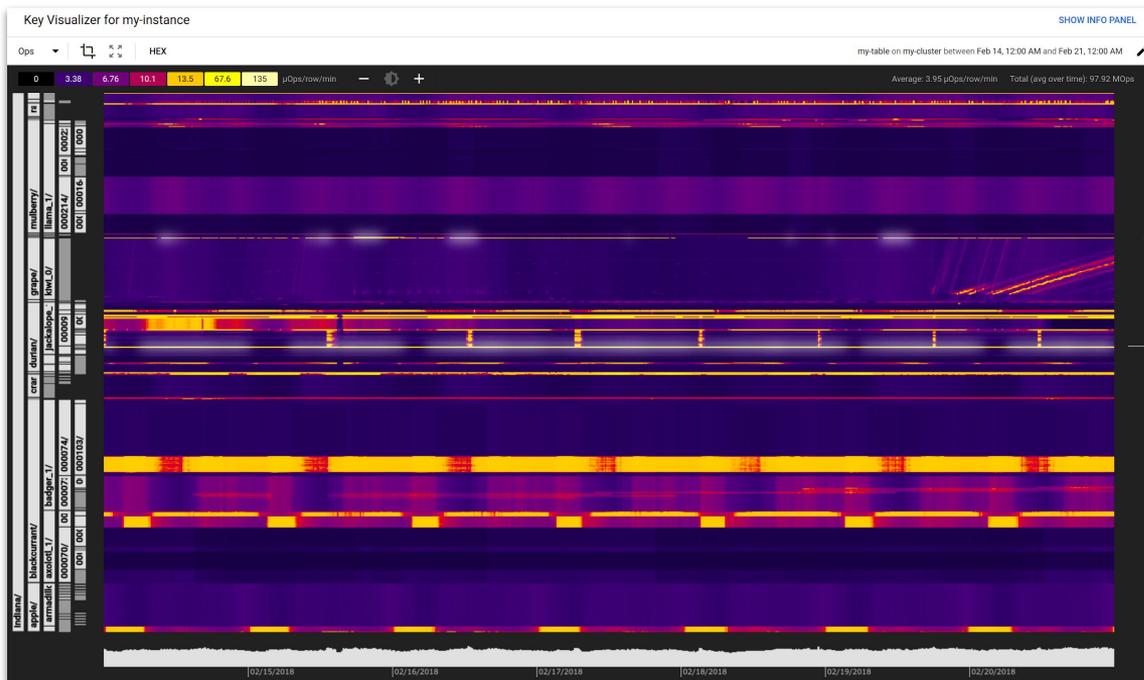
テーブルに大量の列を保持する事になりがちな、IoT センサー値の処理基盤、機械学習のデータ基盤といったシステム用のデータベースに適している

低レイテンシが強く求められるシステムにも向いている

Row Key								
1234500	bob	1	100	4	1	100		
1234515	alice	2	0	124	2	0	124	
1234529	carol	1	50	33	1			
1234543	steve	2	10000	2		10000		
1234551	oscar	2	1000					
1234578	dave	3	700		3		124	3

Key Visualizer によるパフォーマンスの分析

キー空間



時間

色が暗いところほどアクセスが少なく、色が明るいところほどアクセスが多いことを表している。

これによりどの行キーにアクセスが集中しているか、どのようなワークロード傾向があるかなどを判断できる。

その他の NoSQL データベース

AWS

- Amazon MemoryDB for Redis
→ 永続化に重点を置いたRedis 互換 DB
- Amazon Keyspaces
→ Apache Cassandra 互換
- Amazon Neptune
→ グラフデータベース
- Amazon Timestream
→ 時系列データベース
- Amazon QLDB
→ 台帳データベース

Azure

- Azure Cosmos DB
→ データベース作成時にAPI を選択
SQL/Core
Table API
MongoDB 互換
Apache Cassandra 互換
Gremlin (グラフデータベース)
等

Google Cloud

- OSS パートナー マネージド サービス



05

まとめ

マネージド データベース選択の感覚はつかめましたか？

- ✓ クラウド利用のメリットを最大限に享受するためにも
マネージド データベースを積極的に使う点は変わらない
- ✓ AWS や Azure での経験はそのまま応用できる
リレーショナル、分散、NoSQL など、
ワークロードの性質に応じて適切なデータベースを選ぶ
- ✓ OSS や 商用 DB エンジンとの互換性を重視するなら
Cloud SQL / AlloyDB / Memorystore
- ✓ より高いスケーラビリティや可用性を求めるなら
Cloud Spanner / Firestore / Cloud Bigtable
などのクラウド ネイティブ DB を選択



Google Cloud におけるマネージド データベースの選択肢

ユーザー数や経験者も非常に多い OSS の DB や商用 DB を
マネージド サービス化

クラウド環境用に設計された
Google Cloud 独自の DB のこと

OSS 及び商用のマネージド DB



Memorystore
for Redis
for Memcached

キャッシュ



Cloud SQL
for PostgreSQL
for MySQL
for SQL Server

リレーショナルデータベース (SQL データベース)



AlloyDB
Preview
for PostgreSQL
(完全互換)



Cloud Spanner
PostgreSQL
SQL 方言のみ互換



Cloud Bigtable
KVS / ワイドカラム

NoSQL データベース



Firestore
ネイティブモード
Datastore モード

ドキュメント DB



BigQuery

DWH

サーバーレス



mongoDB.



OSS パートナー
マネージド サービス

DB と一緒に使う周辺サービス

データベース移行サービス



Database Migration Service

サーバーレス CDC サービス



Datastream



Thank you.