Cloud Ace

Google Cloud Partner
Managed Service Provider

スモールスタートで始める Cloud Spanner

自己紹介

クラウドエース株式会社

髙島　涼

Takashima Ryo

- ソーシャルゲーム業界歴 5年
- 立ち上げからサービス終了まで経験
- バックエンド全般/インフラ構築/管理画面
- GCP 4年 AWS 1年

ゲーム開発でGoogle App EngineとCloud Spannerを採用

開発と運用が驚くほど簡単なことに感動し、GCPにハマる。

フルマネージドサービスで楽しい開発・楽な運用！ がモットー

今日お話しすること

- Cloud Spannerとは？
- 従来データベースの開発/運用コスト
- 「これまで」と「これから」の開発料金
- デモ
- 開発環境
- テスト環境構成の例

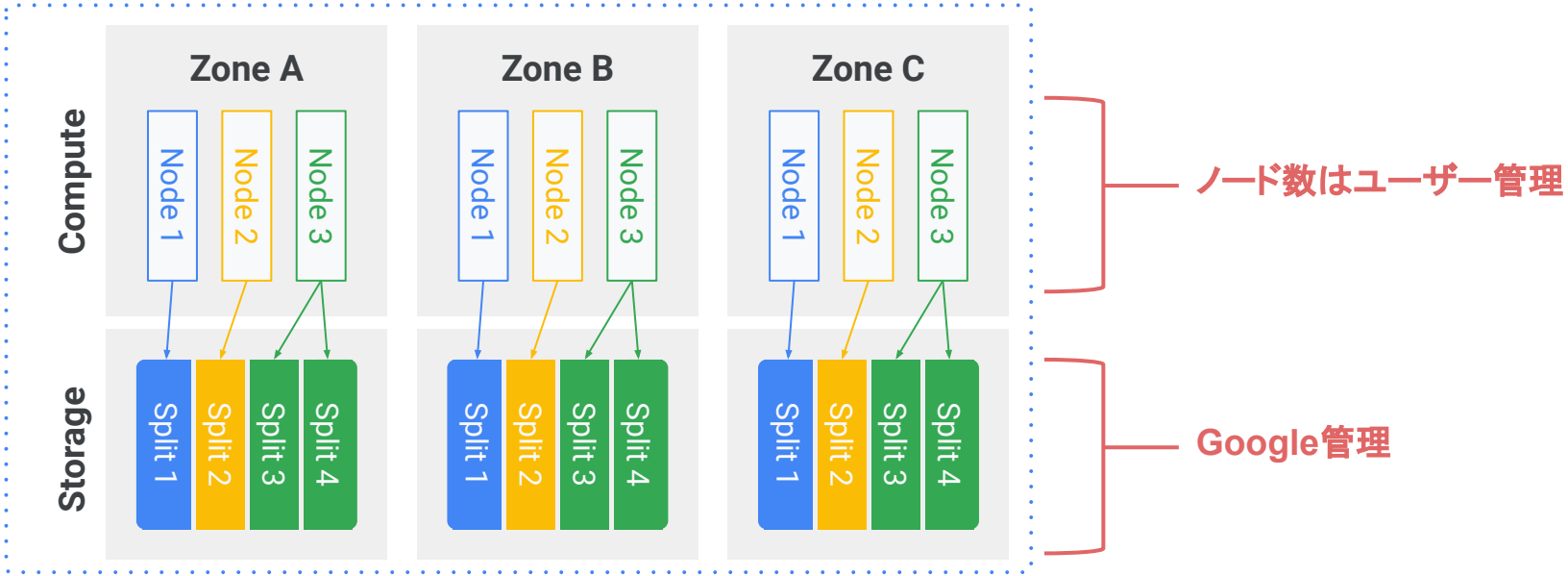
Cloud Spanner とは？

リレーショナルデータベース構造

＋

水平方向のスケーラビリティ

Cloud Spanner アーキテクチャ



従来データベースの開発/運用コスト

項目	従来データベース	Cloud Spanner
メンテナンス	✕ 夜間作業等でメンテナンスによる停止	○ Google管理のため不要
高負荷時の対応 （スパイク対応等）	✕ スケールアップによる停止	○ 無停止のスケールアウト
データベース最適化 （インデックス再構築等）	✕ 手動実施	○ 自動最適化
データベース分割 （シャーディング）	✕ アプリ改修やスキーマ再設計	○ 自動シャーディング
学習コスト	○ 開発経験やナレッジが豊富	△ 基本はリレーショナルモデルだが、キー設計など Spannerの特性を理解して設計する必要がある

[従来] テーブルシャーディング

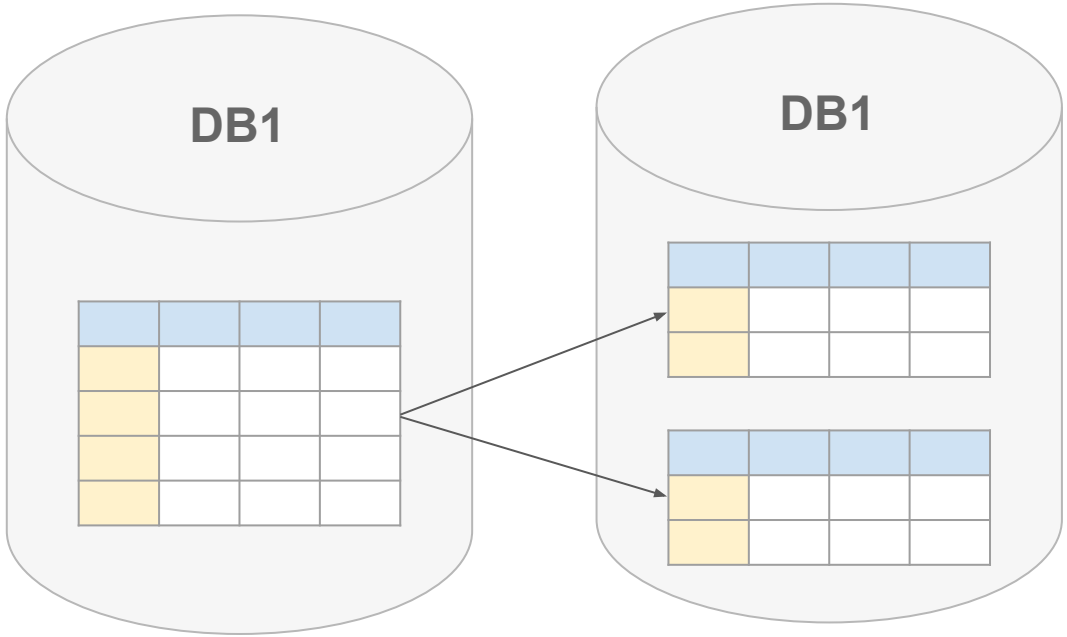
分割基準

- 主キーのレンジ
- 主キーのプレフィックスの一桁
- 利用頻度の高い行、低い行

なんらかのロジックでテーブルを分割する

Pros ：ロックされる範囲は局所的

Cons：DB容量が分散されるわけではない
データが肥大化していくうちに
インデックスがメモリに乗り切らない



[従来] データベースシャーディング

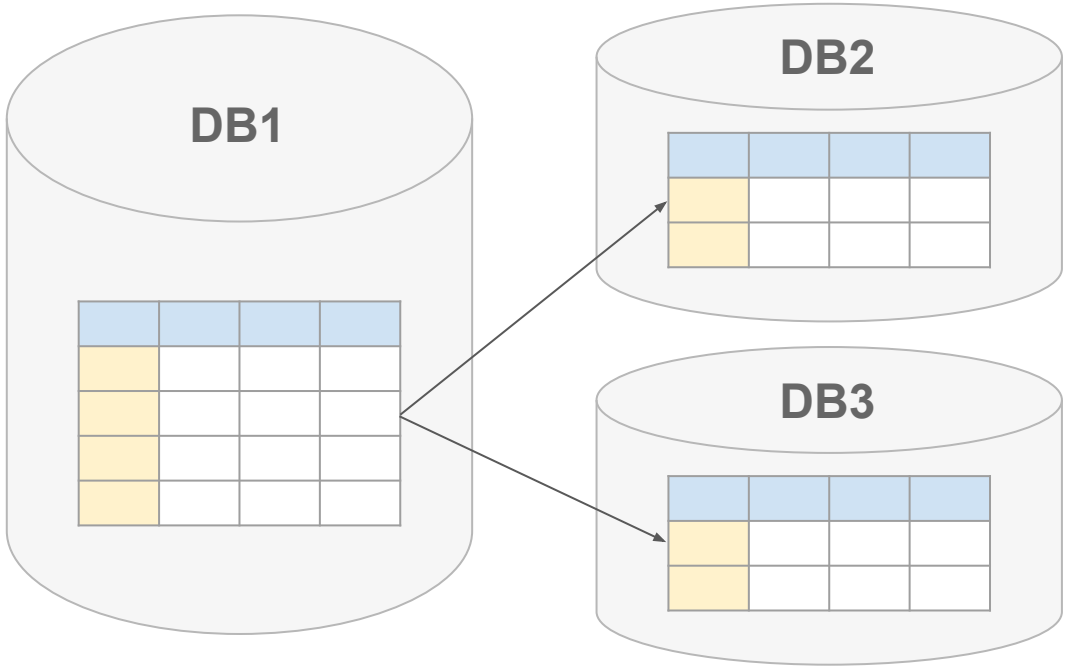
分割基準

- 主キーのレンジ
- 主キーのプレフィックスの一桁
- 利用頻度の高い行、低い行

なんらかのロジックでテーブルを分割する

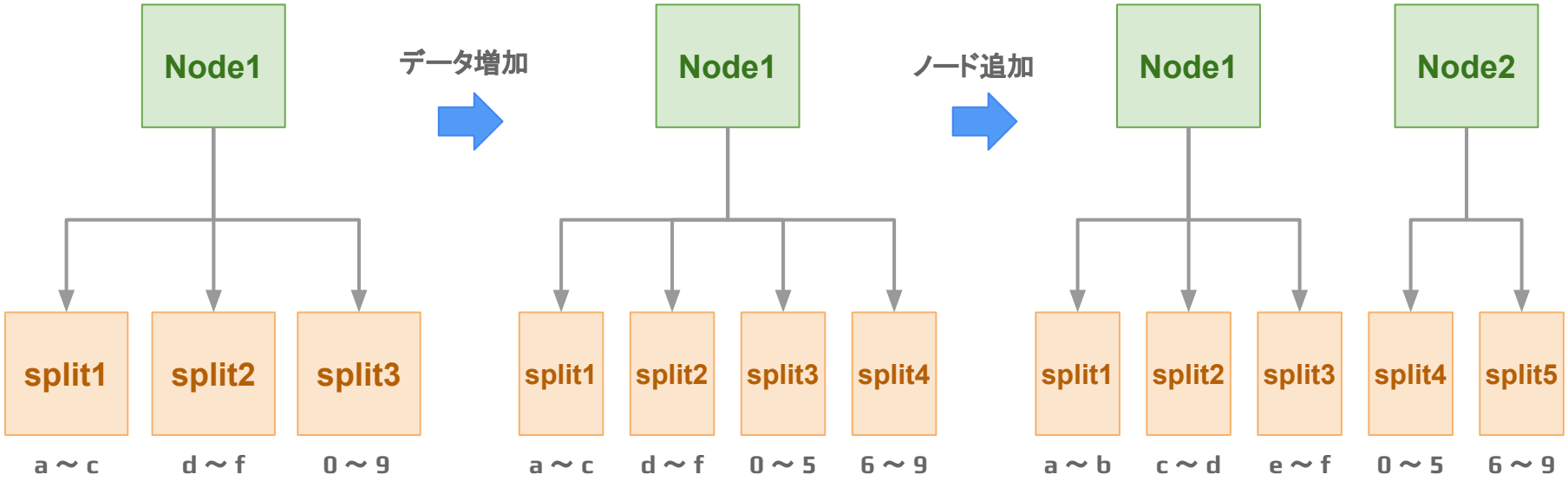
Pros ：CPU / メモリ/ 容量が分散される

Cons：トランザクションが DBを跨げない
データ不整合の懸念
DBを気にした仕様を変更や
アプリ側のロジックで頑張る
プログラムの複雑さ増大



Spannerの自動シャーディング

データ増加によるスプリット追加（自動シャーディング）とノード追加時のSplit割り当ての例



⬆️主キーのprefixによって各splitにデータが配置される。

「これまで」と「これから」の開発料金

「これまで」の開発料金

Q：でもお高いんでしょう？

A：はい。1ノード約10万円です

Q：開発期間中も？

A：はい。1ノード約10万円です

2021年7月までは・・・！！！

Processing Unit

Preview

1ノード = 1000PU以下の構成でコストカット

100PU、200PU、300PU … 1000PU、2000PU、3000PUの単位で増減可能

100PUになるとどうなるか

- 価格 1/10
- 性能 1/10
- ストレージサイズ 1/10

➜ **可用性はそのままに中小規模や開発で大幅なコストカットが可能**

東京リージョンの1000PU 構成で約100,000円/月

東京リージョンの100PU 構成で約10,000円/月（1時間あたり13円）

（※1000PU未満のインスタンスは現在PreviewでSLA対象外です）

名前	ID	構成	処理ユニット	ノード	ストレージの利用率
the-current-exp	the-current-exp	us-east4	1,000	1	0 B / 2 TB
the-new-exp	the-new-exp	us-east4	100	0	0 B / 205 GB

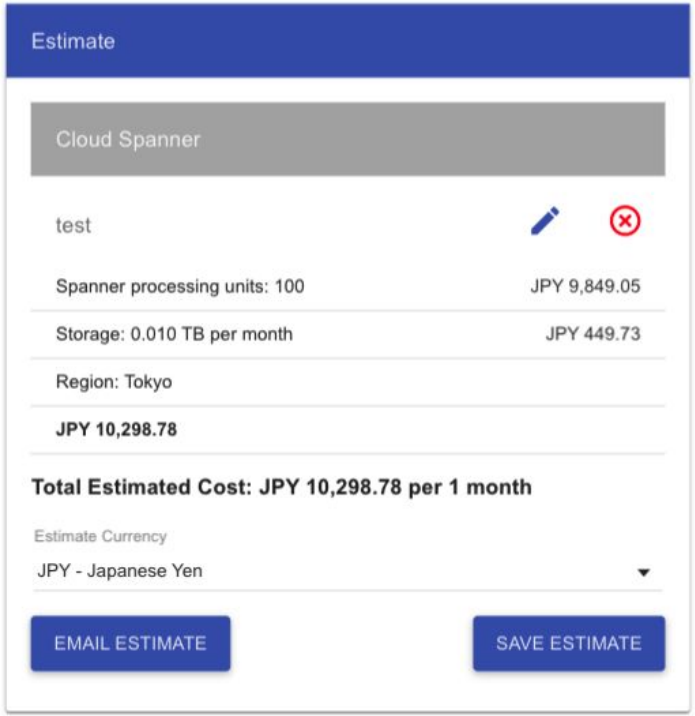
「これから」の開発料金の例

Processing Unit	使用ストレージ量	料金
100PU	10GiB	10,298円 / 月
200PU	20GiB	20,597円 / 月
500PU	50GiB	51,493円 / 月
1000PU (SLA対象)	100GiB	102,987円 / 月

課金対象

1. インスタンスのコンピューティング容量
2. データベースで使用しているストレージ量
3. バックアップで使用しているストレージ量
4. 使用したネットワーク帯域幅の量（下り）

料金計算ツール



スモールスタートできても

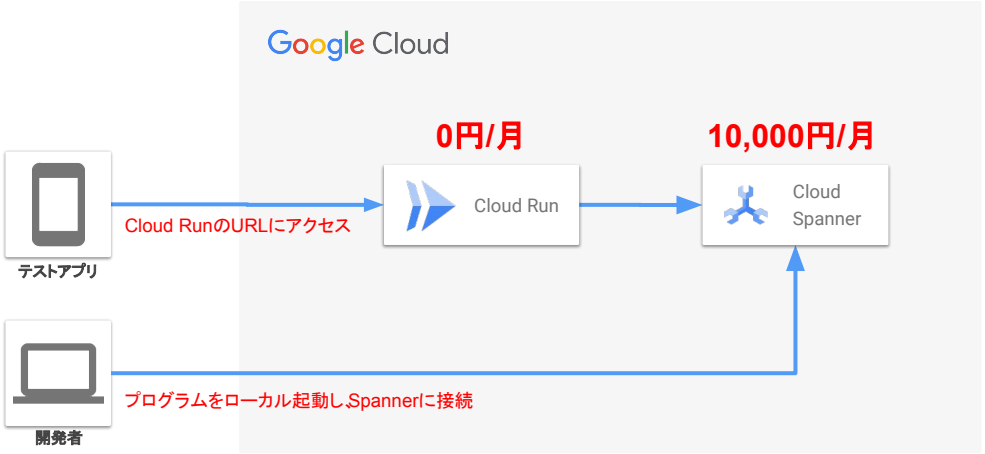
Spannerは難しいんでしょう？

デモ

1. Spannerインスタンスの作成
2. データベース作成
3. テーブル作成
4. データインサート
5. データのクエリ
6. スケールアウト/スケールイン
7. モニタリング・バックアップ

開発環境

テスト環境構成の例



※Cloud Runはゼロスケール構成であれば、開発期間中は無料枠に収まります。

※Cloud Spannerは東京リージョン最小スペックで約10,000円の計算

※その他のプロダクトを使用した場合は別途料金が発生します。

無料で試したい！

ローカルエミュレータ

エミュレータの起動

```
$ gcloud beta emulators spanner start
```

制限事項

- 1つのRead Writeトランザクションのみが許可され、並行トランザクションは Abortされる
- 全てのデータはメモリに保存されるため、永続性をサポートしていない
  - startup.shなど独自に用意する必要あり
- エラーコードが本番と違う可能性がある
- そのほか制限事項あり...

CLIクライアント

[cloudspannerecosystem/spanner-cli]

Spanner用のインタラクティブなコマンドラインツール

並行するRead Writeトランザクションなどコマンドラインでトランザクションの挙動を確認することが可能

右図は emulatorでのクエリ実行例

```
[~] PROJECT_ID=emu-project
[~] SPANNER_INSTANCE_NAME=emu-instance
[~] SPANNER_DB_NAME=emu-instance
[~] export SPANNER_EMULATOR_HOST=localhost:9010
[~]
[~] gcloud spanner instances create $SPANNER_INSTANCE_NAME \
--config=asia-northeast1  \
--description="Emulator Instance" \
--nodes=1
Creating instance...done.
[~]
[~] gcloud spanner databases create $SPANNER_DB_NAME \
--instance=$SPANNER_INSTANCE_NAME
Creating database...done.
[~]
[~] spanner-cli -p $PROJECT_ID -i $SPANNER_INSTANCE_NAME -d $SPANNER_DB_NAME
Connected.
spanner> CREATE TABLE Singers (
      ->   ID STRING(MAX) NOT NULL,
      ->   FirstName STRING(MAX) NOT NULL,
      ->   LastName STRING(MAX) NOT NULL,
      ->   CreateAt TIMESTAMP NOT NULL OPTIONS (allow_commit_timestamp=true),
      -> ) PRIMARY KEY (ID);
Query OK, 0 rows affected (0.00 sec)

spanner> begin;
Query OK, 0 rows affected (0.00 sec)

spanner(rw txn)> INSERT INTO Singers (ID, FirstName, LastName, CreateAt) VALUES
              -> ("2ee52ae7" , "Alice" , "Smith", "2020-11-16T03:31:59.13049Z");
Query OK, 1 rows affected (0.00 sec)

spanner(rw txn)> commit;
Query OK, 0 rows affected (0.00 sec)

spanner> SELECT * FROM Singers;
+----------+-----------+----------+--------------------------+
| ID       | FirstName | LastName | CreateAt                 |
+----------+-----------+----------+--------------------------+
| 2ee52ae7 | Alice     | Smith    | 2020-11-16T03:31:59.13049Z |
+----------+-----------+----------+--------------------------+
1 rows in set (550.8us)
```

ご検討中の皆様へ

autoscalerを使用すれば、**ダウンタイムなしでノードのオートスケール**が可能で
データベースもオートスケールする時代になってきています。

人件費が大幅に削減できるため、運用コストを含めた料金を
総合的に試算するとSpannerはお手頃です。(**「人」**の方がはるかに高い・・・)

既存データベースの**料金と運用コスト**をぜひ比べてみてください。

フルマネージドサービスで

楽しい開発・楽な運用！

Thank you !!