



Cloud Bigtable を使いこなす！ KARTE で秒間 13 万イベントを0.x 秒以内に捌く秘訣

日鼻 旬

株式会社プレイド・Senior Manager Tech Lead

スピーカー自己紹介



日鼻 旬

株式会社プレイド

Senior Manager Tech Lead

2012 年に日本アイ・ビー・エム株式会社に新卒入社し、Web アプリケーション開発・自動テストなどの品質向上等に従事。

2019 年に株式会社プレイドに入社。
KARTE のリアルタイム解析基盤の運用や新基盤への刷新プロジェクトをリード。

Agenda

- KARTE とは？リアルタイム解析基盤とは？
- 現基盤の課題と対策（新解析基盤のアーキテクチャ概要）
- 新解析基盤での Cloud Bigtable フル活用方法
- 今後の展望



KARTE とは？

リアルタイム解析基盤とは？



Customer Experience Platform

karte.io

The screenshot displays the KARTE Customer Experience Platform interface, which tracks and analyzes user behavior on a website. The interface shows several user cards with avatars and activity logs:

- User 71:** Facebook広告から流入しました (流入した今). Activity: Facebookから流入, キャンペーン広告を見た, 3回以上購入, 30.
- User 94:** 商品詳細ページを閲覧しています (閲覧した今). Activity: ichiro.tanaka@sample.com, じっくり検討中, 10分以上滞在しているお客様.
- User 64:** 新着アイテムのページを閲覧しています (閲覧した今). Activity: 本村 一郎, ガジェット系商品をよく見る, お気に入りから流入, 大学生.
- User 56:** よくあるご質問 (質問した今). Activity: vis-12345.

Overlays and callouts provide additional context:

- Top Right Callout:** Webサイトの訪問者の行動を顧客ごとにリアルタイムに解析 (Analyze the behavior of website visitors in real-time by customer).
- Product Recommendation Card:** 冬の主役、アウター OUTER COLLECTION. Text: 絶対、欲しい! ずっと使えるアイテムだけを揃えました。この機会をお見逃しな! CHECK.
- Design Rules Card:** 初めの方へ. Text: 基本サービスやご利用方法をご紹介します。お得な特典や役立つ情報が満載です。 DESIGN RULES. UIデザイナーが知っておくべき 7つのデザインルール. Text: UIデザイナーだけでなくなかなか上達しない、今よりスキルアップしたいという方必読です。 いますぐ確認する.
- Bottom Center Callout:** 一人一人に合わせた顧客体験を提供 (Provide a customer experience tailored to each individual).
- Bottom Right Card:** ご希望の物件は見つかりませんでしたか? 検索条件を少し変更するだけで、あなたの理想の物件に出会える可能性が高くなります。条件を変えて検索してみてください。 X 閉じる.



Customer Experience Platform

karte.io

145 億 UU

累計ユーザー数 ※1

2.43 兆円

年間解析流通金額 ※2

135,000 over

秒間トラッキング数 ※3

0.x 秒/解析

解析速度

180 + PB

月間解析データ量

8 + PB

蓄積データ量

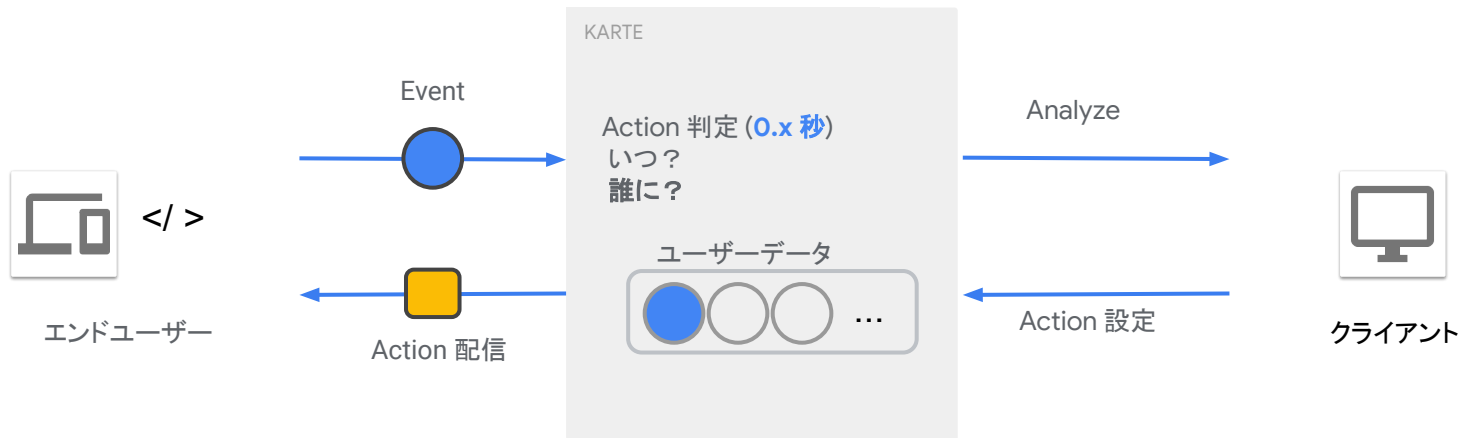
※1 ローンチ～2022 年 3 月までの解析ユニークユーザー数の実績

※2 EC 領域における解析流通金額。2021 年 3 月～2022 年 2 月までの
単年の実績

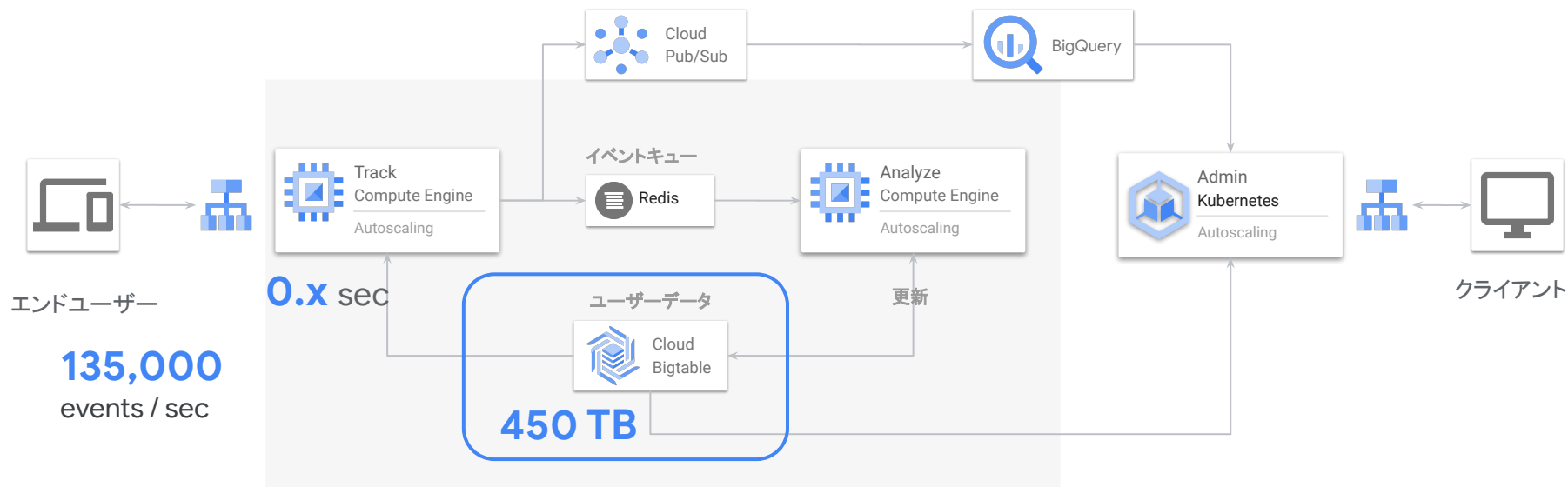
※3 秒間解析イベント数

(閲覧、購入、クリックなど全計測イベントが対象。2022 年 3 月最大値)

リアルタイム解析基盤とは？



アーキテクチャ上のポイント



なぜ Cloud Bigtable なのか？

高スケーラビリティ

累計 145 億ユーザー

全データ量 450 TB

低レイテンシー

平均 20 ~ 50 msec

1 ユーザーのサイズ: 50 KB 程度

低コスト

Cloud Spanner と比べて割安

ストレージ料金



現基盤での課題と対策

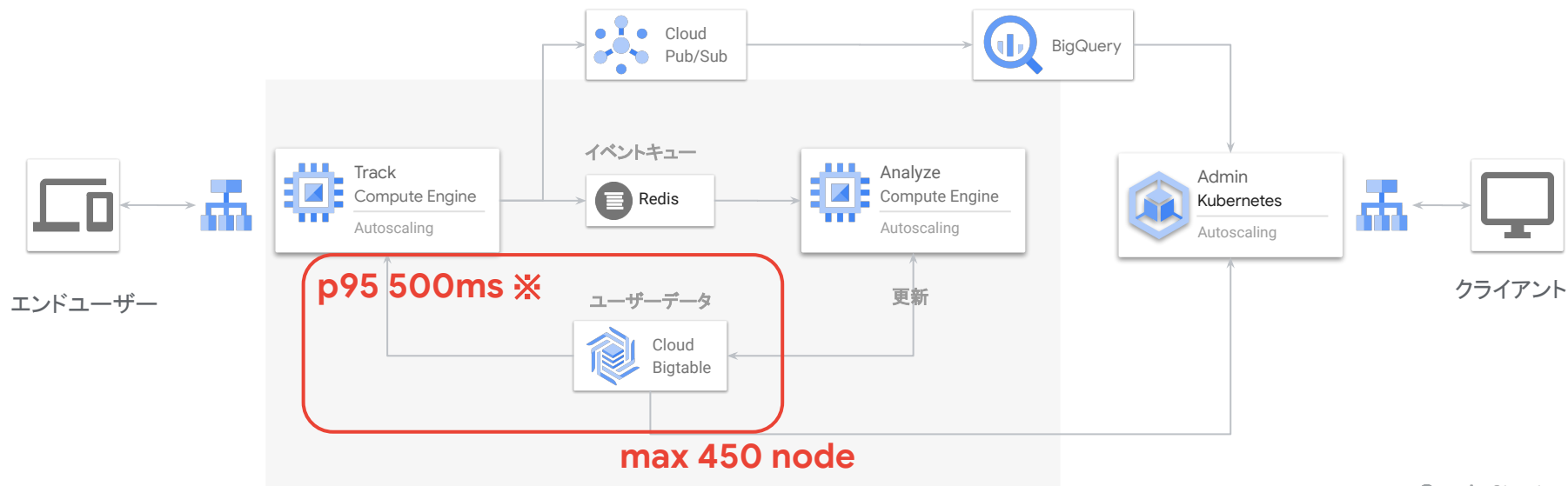
2015 年から運用・改善して見えてきた課題

01 | これ以上大幅なパフォーマンス・コスト改善が見込めない

02 | 結果整合性(古いユーザーデータでの解析が起きうる)

ボトルネックはユーザーデータの Read 処理

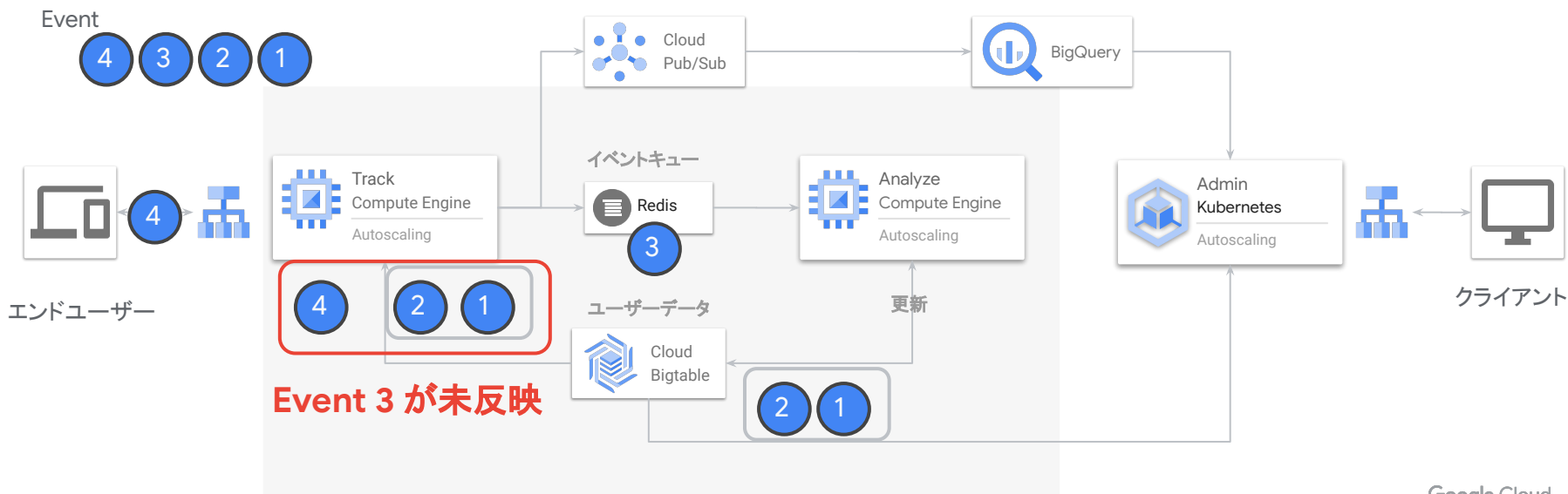
- パフォーマンス：ユーザーデータ取得処理が支配的
- コスト：低レイテンシー維持のためコスト増加



※ Compute Engineからみたレスポンスタイム

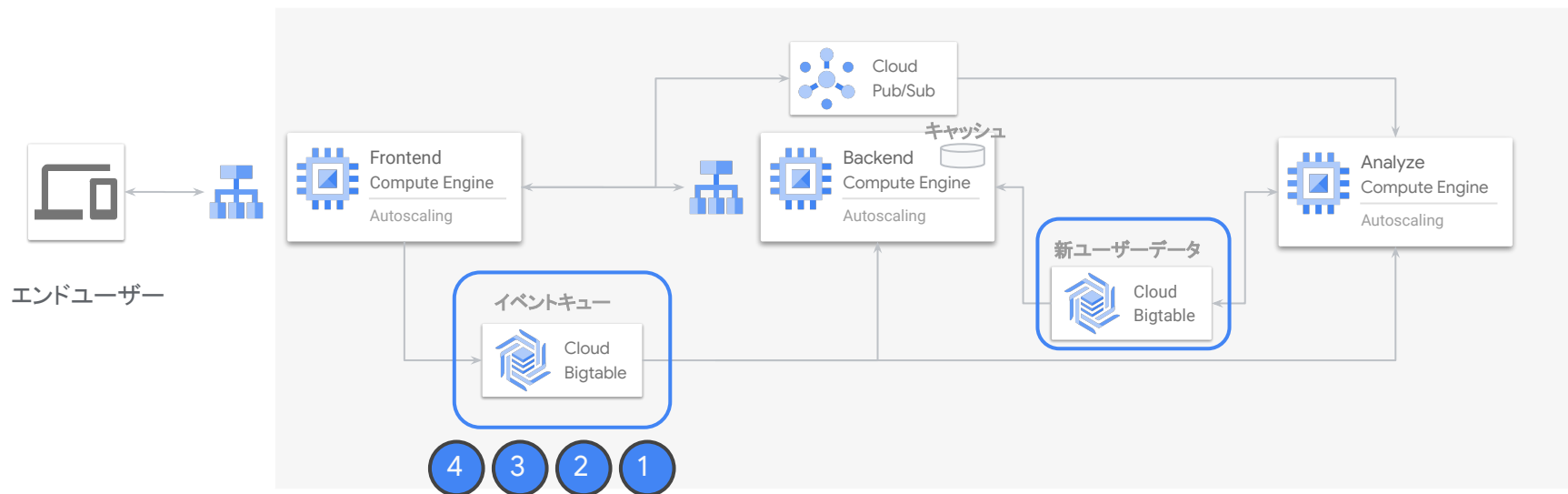
KARTE リアルタイム解析における結果整合性とは？

- 更新処理待ちのイベントがユーザーデータに加味されない時がある
- 更新処理待ち時間を減らすには限界がある



新解析基盤 のアーキテクチャ

- ユーザーデータのスキーマ再設計 + キャッシュ戦略
- 強整合性の実現
 - まずイベントキューに書き込む
 - 更新待ちイベントはキューから取得し常に最新状態を維持



新解析基盤におけるキーポイント

01 | Cloud Bigtable (ユーザーデータ) のスキーマ再設計

02 | Cloud Bigtable を Queue として使い、強整合性を実現



新解析基盤での Cloud Bigtable フル活用方法

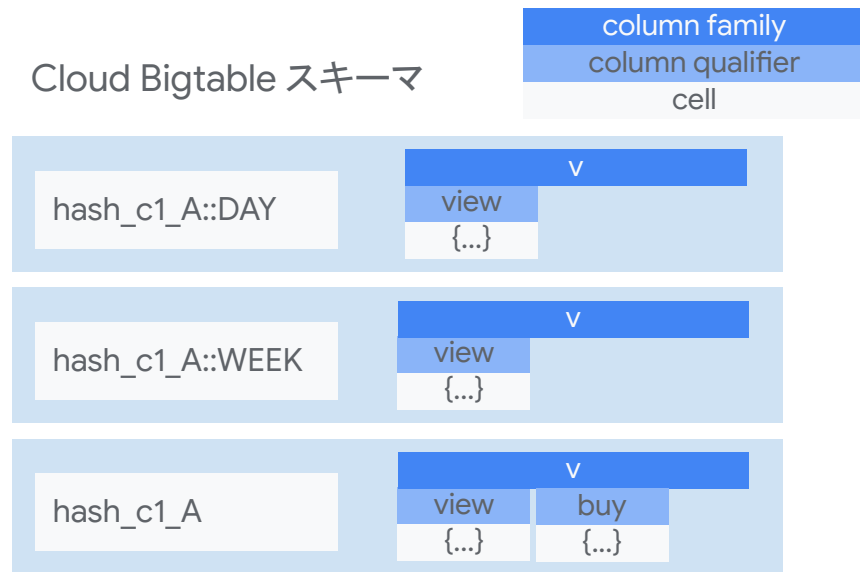
ユーザーデータ (Cloud Bigtable) のデータ構造 (Before)

- Application 上のユーザーデータモデルをそのまま Cloud Bigtable スキーマにも投影
- Read 時はプレフィックスによる Scan が発生

ユーザーデータモデル



Cloud Bigtable スキーマ



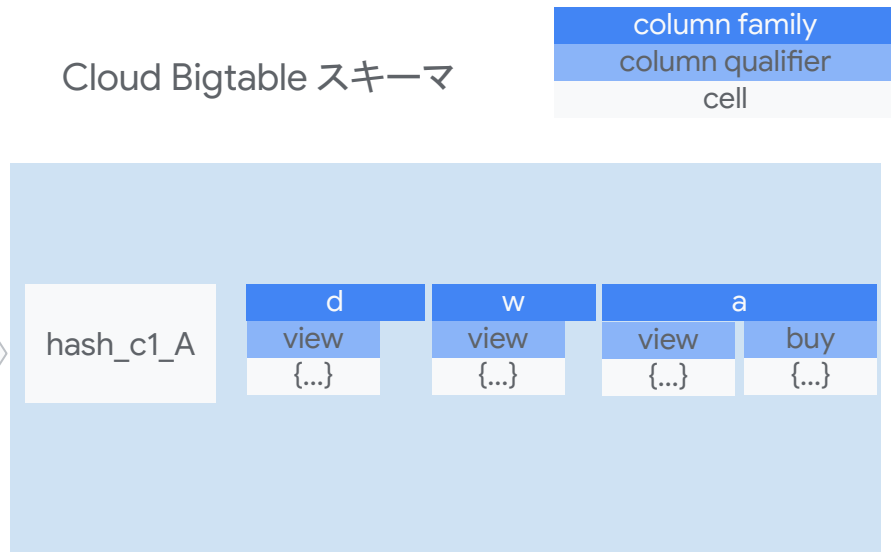
ユーザーデータ (Cloud Bigtable) のデータ構造 (After)

- Read 効率最大化のため、1 行にまとめ、Column Family にて期間データを分割
- イベントごとの統計情報をアプリケーション側で圧縮して保存

ユーザーデータモデル



Cloud Bigtable スキーマ



スキーマ変更による効果 & 副作用

Read 速度向上

response time が改善
p95 500ms -> 350ms

コスト削減

Cloud Bigtable CPU 使用率低下
max 450 node -> 300 node

ホットスポット増加

1行あたりのデータサイズ増加
Bot などによる大量アクセス時にホット
スポットが起きやすくなる
Application 側での対策

- Bot 自動検知 & ブロック

Lessons Learned

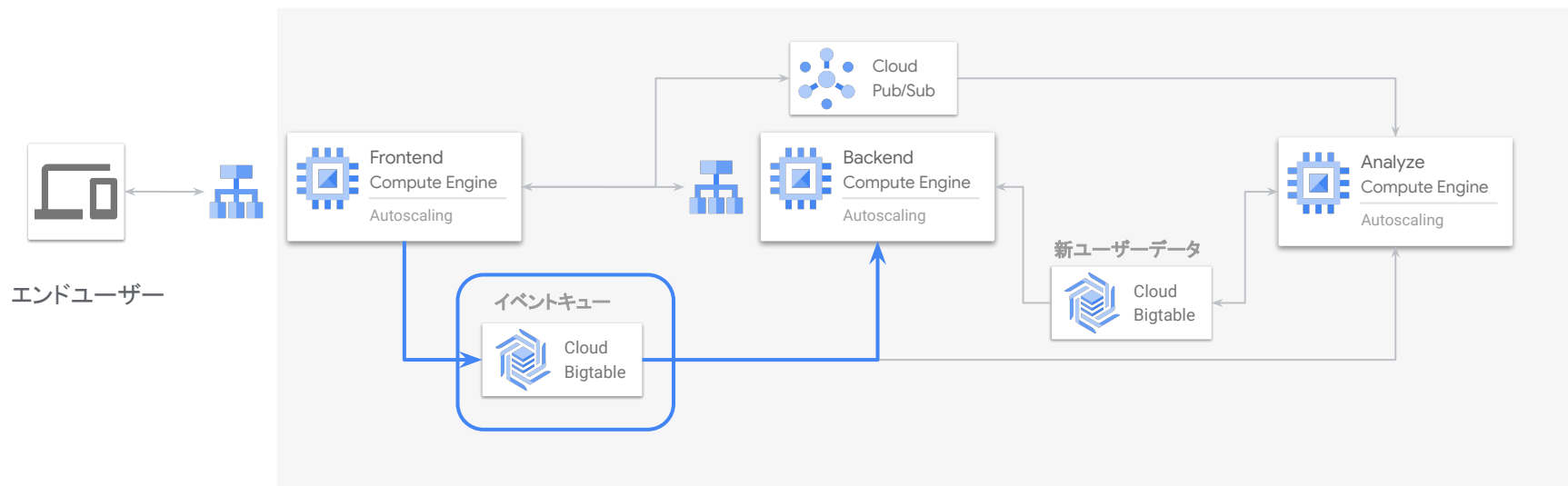
- 01 | データモデルを Cloud Bigtable に投影する
スキーマ設計時に最適化の余地が生まれる
- 02 | 同時に取得するデータは 1 行にまとめた方が
パフォーマンス・コスト的にも効率が良い
- 03 | ただし、1 行のデータサイズが大きすぎると過負
荷状態が発生しやすくなるので注意



強整合性の実現 (Cloud Bigtable as a Queue)

課題はイベント発生順序担保 & 低レイテンシー維持

- イベントキューへの Write のオーバーヘッドは最小限に抑えたい（10 ms 以下）
- イベントキューからの Read も同様に 10 ms 以下に抑えたい



なぜ Cloud Bigtable を Queue として使うのか？

- 既存の Queue 用の Product では低レイテンシーと 高スケーラビリティの両立は難しい
- Cloud Spanner の可用性は魅力だが、Cloud Bigtable レベルの低レイテンシーかは未知数

Product	低レイテンシー	高スケーラビリティ	可用性	コスト
Redis Queue	◎	✕	✕	◎
Cloud Pub/Sub (with ordering)	✕	◎	◎	○
Cloud Bigtable	◎	◎	○	○
Cloud Spanner	?	◎	◎	△

イベントキュー用の Cloud Bigtable スキーマ設計

- ユーザーデータの timestamp を元に最新状態まで Row Range Scan する
- ガーベッジ コレクション ポリシーを用いて古いデータは自動削除される

イベントデータ



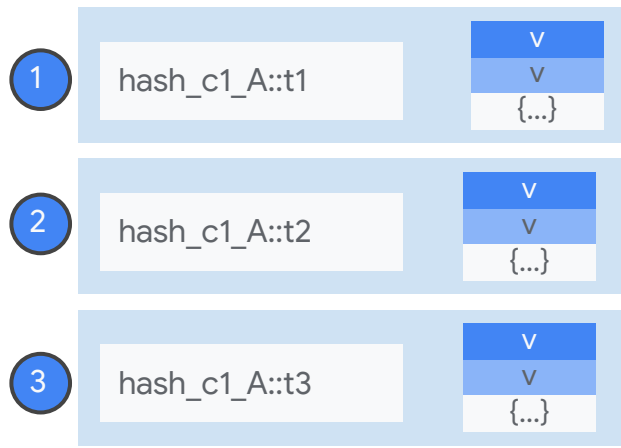
user_id : A

client_id : c1

イベントデータ



Cloud Bigtable スキーマ with TTL



column family
column qualifier
cell

Row Range Scan
from : hash_c1_A::t1
to : hash_c1_A::now

パフォーマンスは問題なく、可用性向上も検討中

Write パフォーマンス

データサイズ : 1 KB程度

response time

- avg : 5 ms
- p95 : 10 ms
- max: 120 ms

Read パフォーマンス

Scan データ量 : 1 ~ 3 Row × 1 KB

response time

- avg : 5 ms
- p95 : 10 ms
- max: 300 ms

可用性

現状、単一クラスター構成
レプリケーションでは書き込み
順序の維持はサポートしてない

Lessons Learned

- 01 | Cloud Bigtable は低レイテンシー・高スケーラビリティを備えた Queue として十分に使える
- 02 | Cloud Bigtable でのレプリケーション下では強整合性は担保されなくなるので注意が必要

まとめ

01 | Cloud Bigtable ではスキーマ設計がとにかく重要

02 | Read 効率を高めるため 1 行にデータをまとめる方法は有効

03 | スキーマ設計次第で Queue としても使える



今後の展望

パフォーマンス 10 倍・コスト 1 / 10 を目指して

01 | 新解析基盤に完全移行する

02 | Cloud Bigtable だけではなく、
他のサービスも適材適所で使い倒していく

03 | Google Cloud には、他の Cloud Service では
代替しえない Database サービスを提供し
続けることを期待

Thank you.

