



ClickHouseで実現する 広告成果レポートの高速化と安定化

AI Marketing solution統括本部
テクノロジー戦略本部 SREチーム

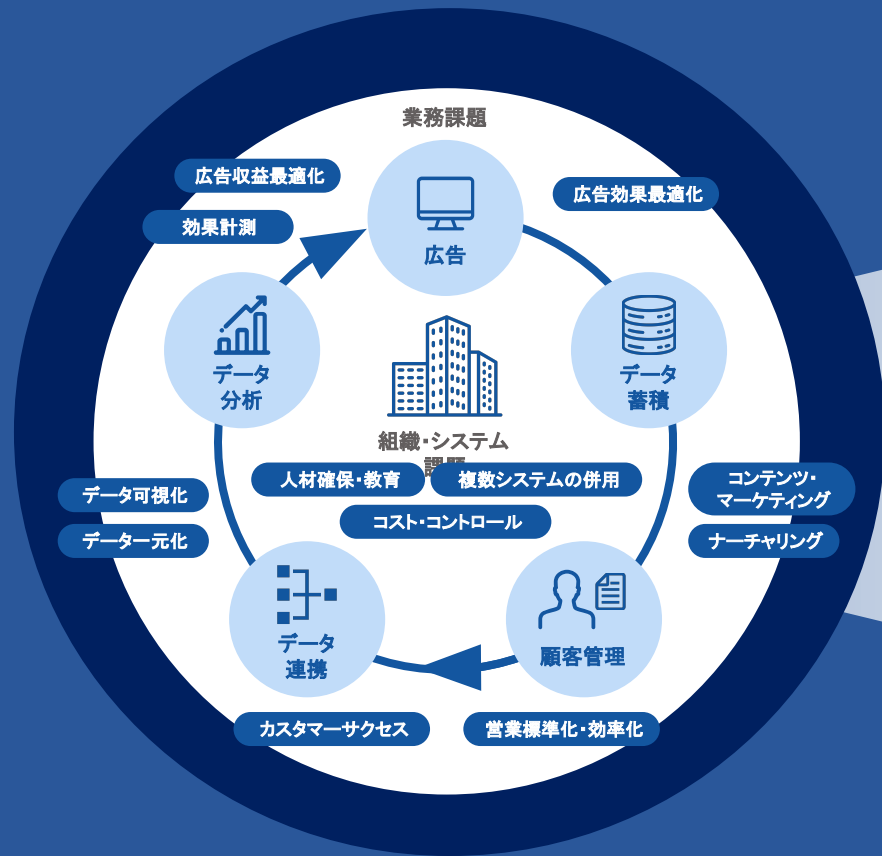
片岡崇史

自己紹介

- 片岡崇史
- SREチーム所属
- ClickHouseに関わり始めて約3年
- 最近はKubernetesを触っている

ジーニーグループが目指す姿

- ・ ジーニーグループは、複雑化・高度化するマーケティング課題の解決を総合的にご支援
- ・ マーケティングのハードルを下げ、最新のAI技術を活用した高付加価値の製品サービスを企業がよりスムーズに社会に広められる世界を目指す

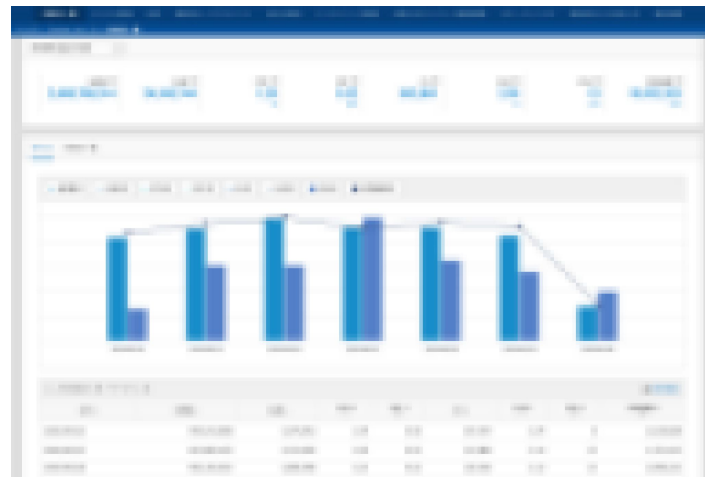


- ・ ワンストッププラットフォーム
- ・ 多様なプロダクト領域
- ・ 最新のAI技術の活用

誰もがマーケティングで 成功できる世界

- ・ 高いコスト・パフォーマンス
- ・ 業務の自動化・効率化

広告プラットフォーム(DSP)の管理画面上に表示する**広告成果のレポートDB**として利用してきた(約7年前から)

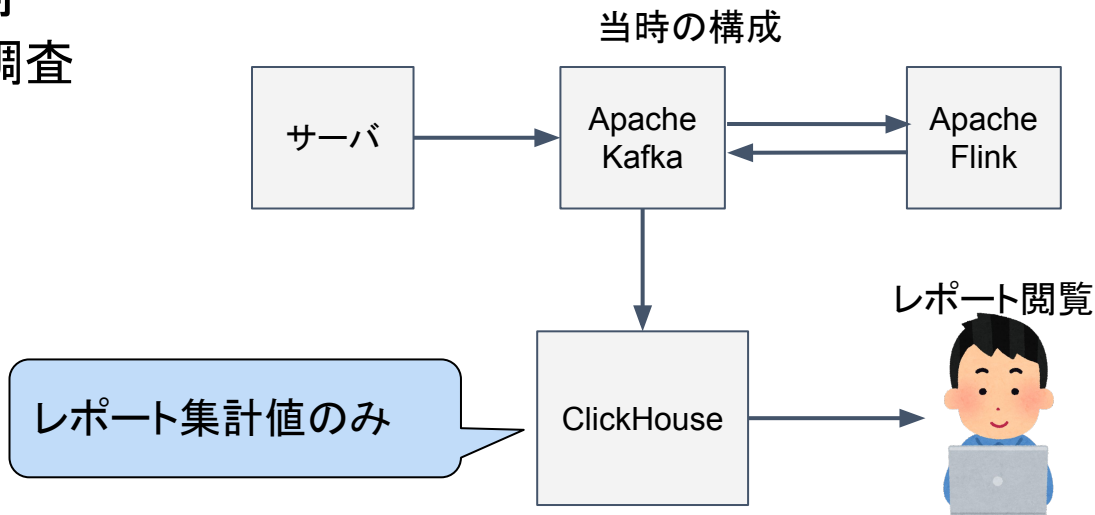


過去の発表など

- [MySQLを捨てClickHouseでレポートAPIのレスポンスを1000倍高速化した話](#)
- [ClickHouse導入事例紹介](#)
- [ClickHouseのテーブル構成を考え直してみた](#)

レポート集計はKafka/Flinkで処理していたが...

- Apache Flink周辺の障害が多発するが根本問題を解決できない
- 障害で数値が欠損したときの再集計に時間がかかる
- 成果ログの調査が面倒
 - 大量のログをgrepして調査



ClickHouseをレポートDBとしてだけでなく、ログからの集計にも利用する

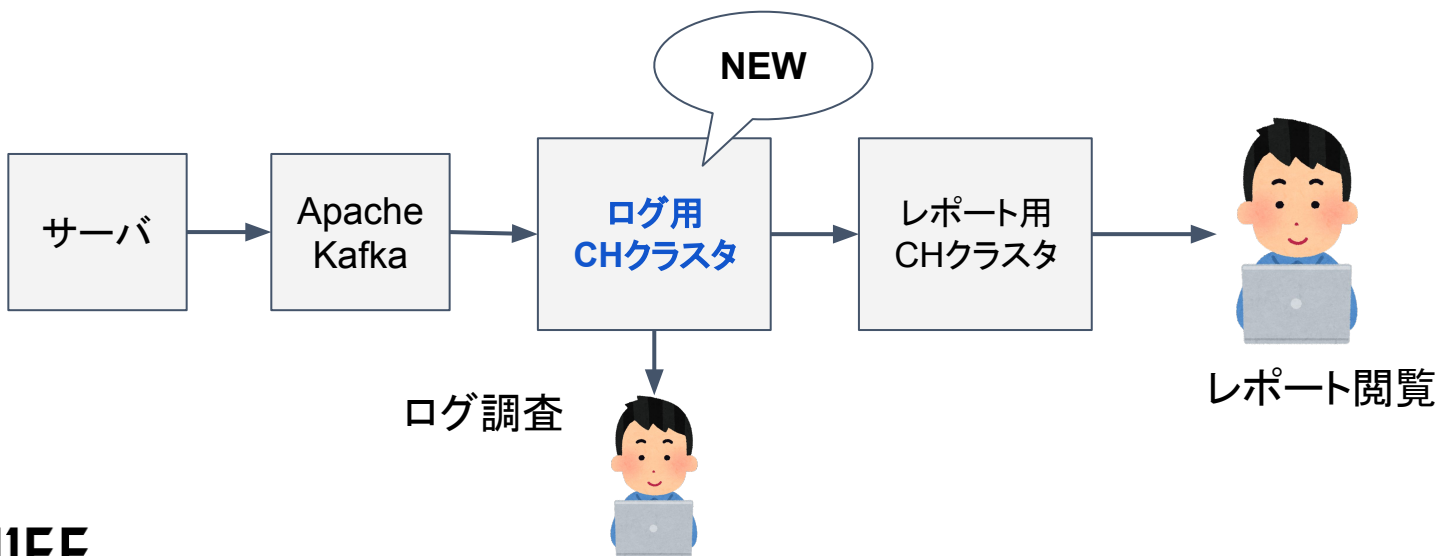
全てBigQueryに乗り換えることも選択肢の一つだったがコストの観点で断念

- 大量のログを扱う
- DSP管理画面から無制限にレポート表示させたい

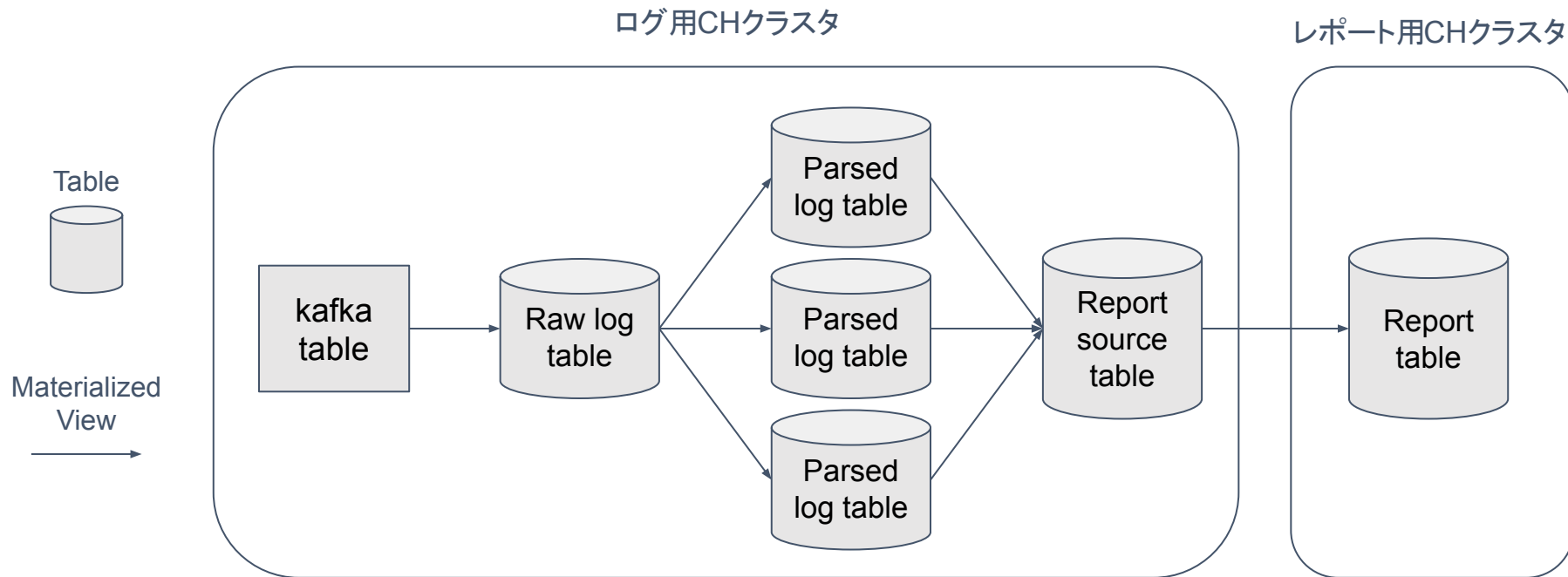
- Apache Flink周りの障害に悩まされることがなくなった
- 再集計も比較的簡単&かかる時間も短くなった
- ログ調査がSQLを使ってできるようになった
- サーバ台数が減ってコストカット
 - Flinkサーバ全台削減
 - Kafkaサーバ半数減
- ログが出力されてからレポートに反映されるまでが速くなった
 - 遅延40分以上から3分以内に
- 集計からレポート閲覧までClickHouseのみで完結できるようになった
 - 全てDDLのみで表現されている

レポート集計基盤刷新後の構成

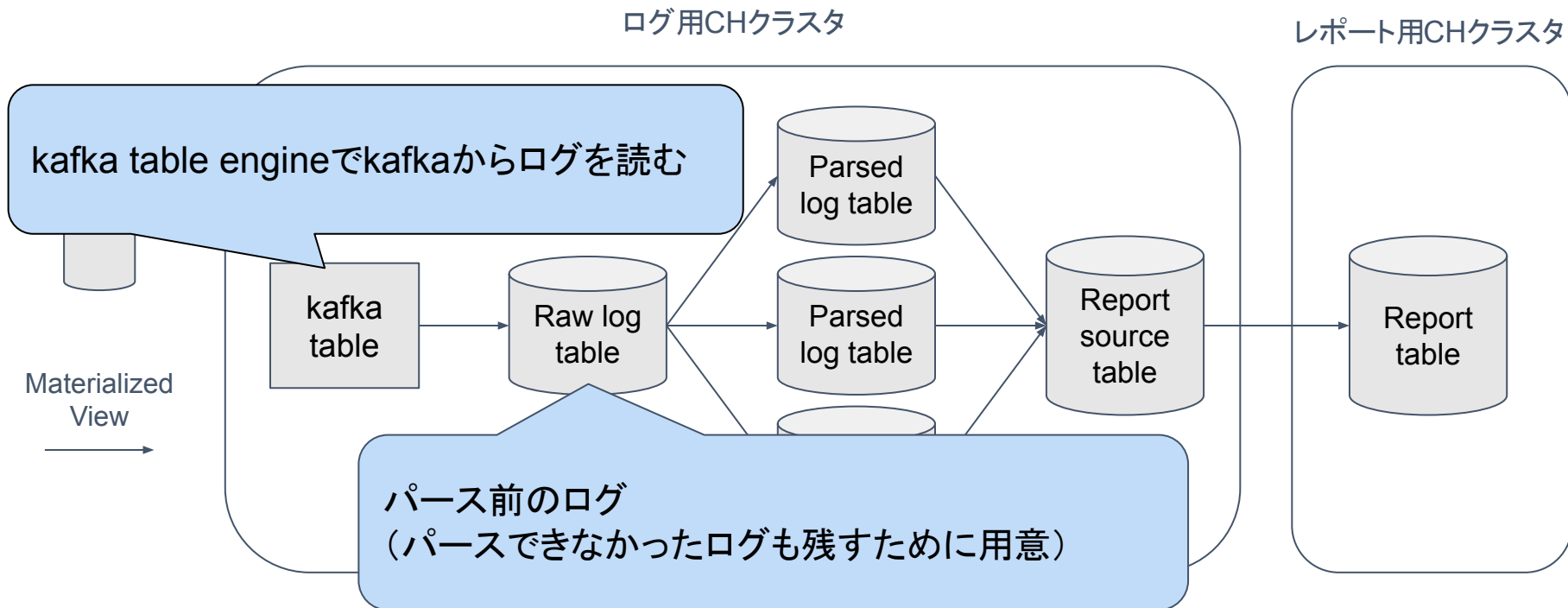
- ログ用CHクラスタはApache Kafkaからログを取得、パースしてログテーブルに蓄える
- ログからレポート行を作り、レポート用CHクラスタのレポートテーブルへ挿入して集計



テーブル構成



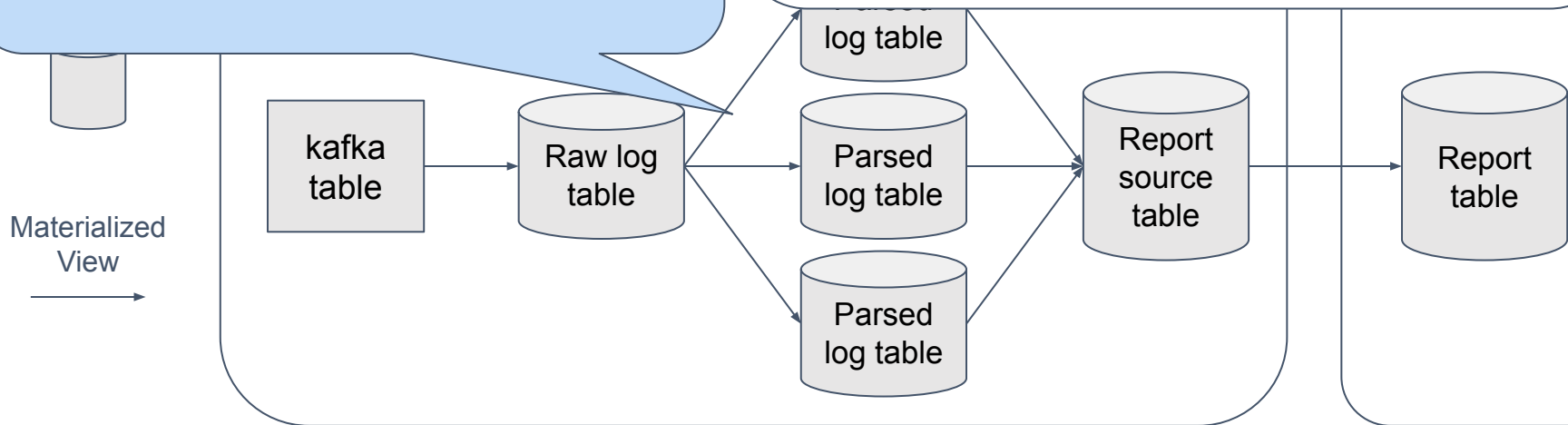
テーブル構成



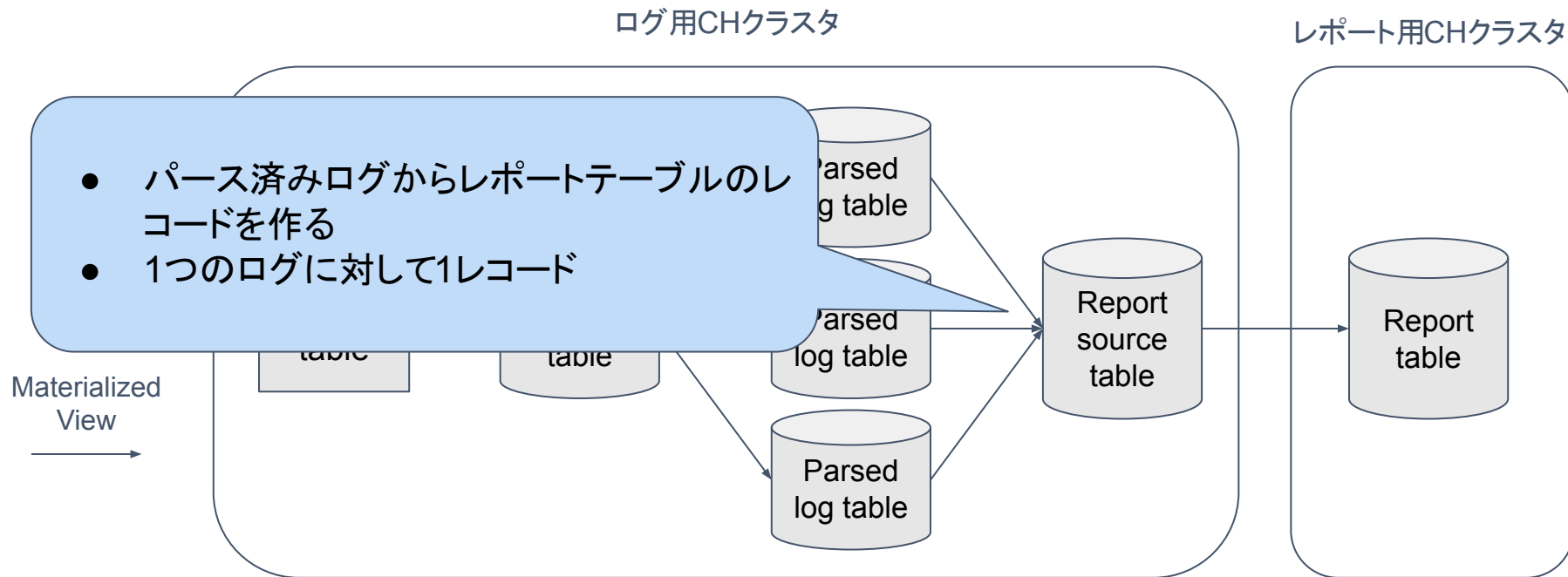
テーブル構成

- Materialized Viewでパースしてパース済みログテーブルへ
- ClickHouse組み込みのJSONパース関数を使う

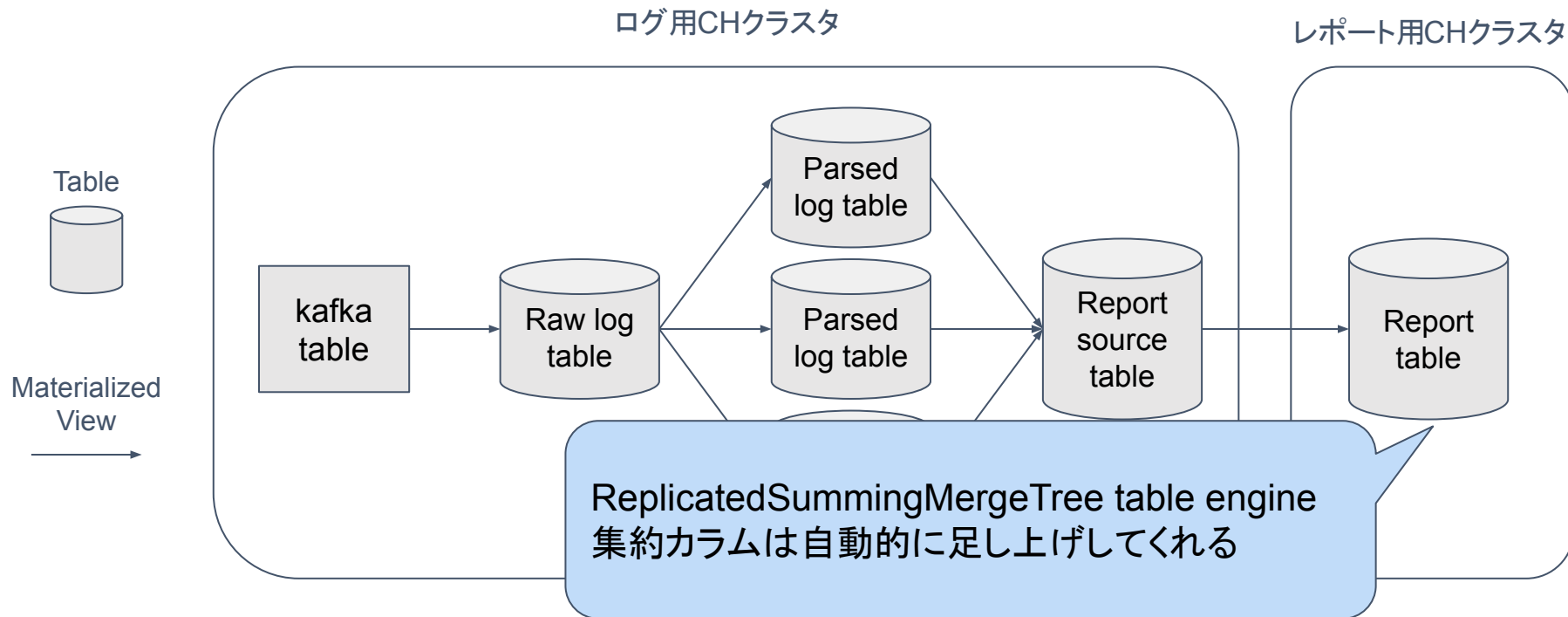
```
CREATE MATERIALIZED VIEW mv_parse_click
TO parsed_click_log_table
SELECT
    visitParamExtractUInt(record, 'xxx_id')
    AS xxx_id,
    visitParamExtractString(record, 'yyy_id')
    AS yyy_id,
    ...
FROM raw_log_table
WHERE kind = 'CLICK'
```



テーブル構成



テーブル構成



- MySQL table engineで接続することもできるがClickHouseのテーブルとJOINするとパフォーマンスがでない
- 代わりにClickHouseの**Dictionary**を使う
 - キーから属性値へのマップ
 - MySQLをデータソースとするオンメモリキャッシュとして

```
CREATE DICTIONARY advertisers (  
    id UInt32,  
    name String  
)  
PRIMARY KEY id  
SOURCE (MYSQL (  
    -- MySQL テーブルへの接続情報  
))  
LIFETIME (300)  
LAYOUT (HASHED());
```

```
SELECT  
    dictGet(  
        'advertisers',  
        'name',  
        100) AS advertiser_name
```

- レポートテーブルのクエリパフォーマンス向上
 - カラムの圧縮アルゴリズムの変更を検討する
 - 圧縮されたカラム(Gorilla,ZSTD)のdecompressに時間がかかっている(CPU負荷)
 - Projection機能を使う
 - テーブルから集約した射影テーブルを作ってくれる機能
 - 管理画面が発行するクエリが悪くなくて射影テーブルを使ってくれない
- ログ形式をバイナリにする
 - JSONよりもパフォーマンス向上が期待できる

- ClickHouseをレポートDBだけでなくレポート集計基盤としても使う
 - 低コスト
 - 低遅延
 - 高い安定性
- 利用しているClickHouseの機能
 - SummingMergeTree, Materialized View, Dictionary, etc...
- レポートテーブルのクエリパフォーマンスが課題



多数のプロダクトを軸に成長している企業です！

自分にぴったりの環境でスキルアップし

日本発の世界的なテクノロジー企業を

一緒に創りませんか？