

ポイント

- 暗号化したまま計算できる技術
- クラウドを安全に活用できるデータ処理基盤
- 専門知識が不要なクエリ記述ルール

課題とソリューション

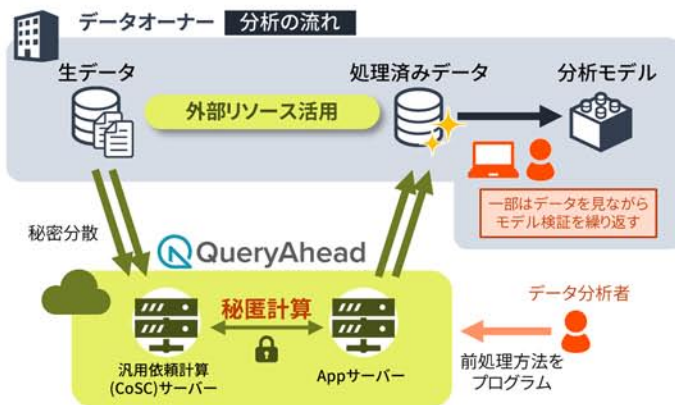
機密性の高いデータを分析する現場では、データの外部持ち出し禁止といったセキュリティ制約を課されることが多い。そのため、作業場所と利用可能なコンピューティングリソースが限定され、無駄な時間とコストが発生している。作業場所を選ばず、スケーラビリティがあるクラウドサービスを利用できれば、大幅な時間とコストの削減を期待できるが、上述のセキュリティ制約がそれを阻んでいる。

QueryAheadはこの課題を解決すべく、秘匿計算技術をベースに開発しているクラウドサービスである。秘匿計算技術とは暗号化したまま計算を可能にする技術である。これによってデータを秘匿化したまま処理し、外部でのデータ復元ポイントを排除することで、情報の安全性を確保する。その結果、セキュリティを担保しつつクラウドを活用したデータ処理を実現できる。

さらに、秘匿計算の専門知識を持たない一般のデータアナリストや開発者が、通常のプログラミング感覚で秘匿クエリを発行可能とする平易な記述ルールを採用している。

全体構成

データオーナーが蓄積している生データの量は膨大である。そこで、データの整形・集約といった前処理でQueryAheadを利用する構成例を下図に示す。従来は、不便なオンサイト作業で膨大なデータをさばく必要があったのに対し、外部リソースを活用することで、その作業を効率化可能である。秘匿計算によって、外部には復元ポイントが無くデータの安全性も確保されている。



データ分析業務における課題



- ① 人的リソースの課題
貴重なデータアナリスト人材の配置に制約
- ② コンピューティングリソースの課題
オンサイトで限られたスペックのコンピュータしか使用できない

物理的制約によりデータ利活用が **高コスト・非効率**に

ソリューション

クラウドを安全に活用できるデータ処理基盤

🔒 秘匿計算技術 → 外部にデータ復元ポイント無し

- ① 外部からのデータ操作が可能になり **人材の配置が自由に**
- ② 外部計算リソースの活用により **大規模データの利活用が容易に**

人的 / コンピューティング
リソースの解放による **コスト削減+効率化**

クエリ記述例

データ分析者や開発者が慣れ親しんでいるSQL的な操作を、数行の記述で秘匿計算として実行可能となっている。

```

q = (Query('result')
     .select(date,
             kari_code,
             kashi_code,
             kari_amount,
             kashi_amount,
             (kari_code + kashi_code) * 10)
     .where(date == _date('2019/8/27'))
     .from_('journal'))
app_client.req_query(q)
    
```

SELECT句: 基本演算の記述

WHERE句: 日付に一致する行を抽出

FROM句: 日付に一致する行を抽出

Pythonプログラムとして記述・実行可能

主な対応機能

- | | |
|------------------------|-----------|
| 基本演算 (加算, 減算, 乗算) | JOIN |
| SELECT句 (列の選択, 列同士の演算) | ORDER BY |
| WHERE句 (等価評価による抽出) | GROUP BY |
| FROM句 (テーブルの選択) | 基本統計機能 など |