

ヘテロジニアスなIn-databaseデータ解析・機械学習基盤

— GPUを活用した高速PostgreSQLの実現と事業化 —

海外 浩平 (HeteroDB)
 柏木 岳彦 (HeteroDB)
 遠藤 克彦 (慶應義塾大学)

◆GPU/SSDといった新世代の半導体の力を引き出し、誰もが使いやすいデータ解析・機械学習の基盤を実現

◆製品の特長

1Uノードで従来のDWH専用機に匹敵する性能
 PostgreSQLと統合され、周辺ソフトウェアを“そのまま”利用することができる。

◆ターゲット市場・用途

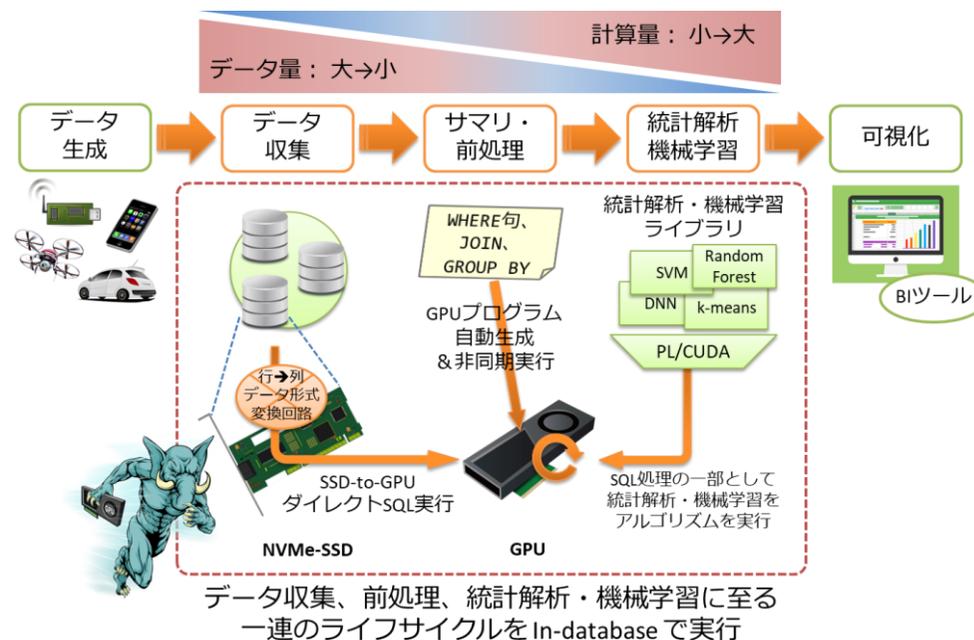
旧世代DWH(Netezza/Greenplumなど)の代替
 高速ログ分析ソリューション(協業案件)
 金融向け市場分析ソリューション(協業案件)
 製造業向け異常検知ソリューション(協業案件)

◆Value Proposition

従来は高価なDWH専用機や小規模なクラスタを構成していたシステムが、シンプルなシングルノード構成の PostgreSQL で実現可能に。

- ① シングルノード化により、アプリケーション開発やシステム管理が圧倒的に楽になる。
- ② “データのある場所”で統計解析・機械学習を実行できるため、データ移動を省略できる。
- ③ OSSとコモディティHWの組合せで高性能を実現。商用DBと比べ圧倒的な低コスト。

➔ 中堅～大企業向けに、協業パートナーとソリューション化が進行中。2018年度冒頭製品リリース。



問い合わせ先: contact@heterodb.com / <http://heterodb.com/>

【補足資料】 中核機能の一部、ビジネスモデル

中核機能① SSD-to-GPUダイレクトSQL実行

SQL最適化ステージ

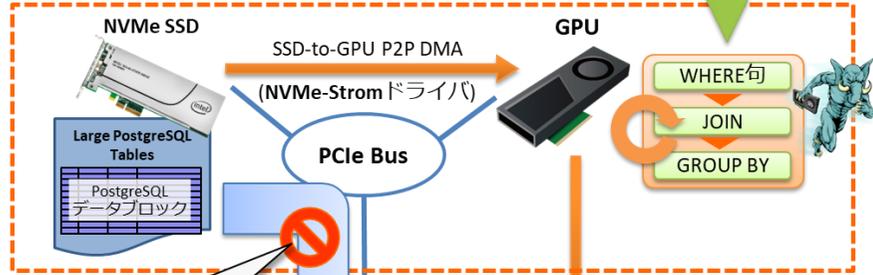
```
SELECT cat, count(*), avg(X)
FROM t0 JOIN t1 ON t0.id = t1.id
WHERE YMD >= 20120701
GROUP BY cat;
```

SQL-to-GPU
プログラム
ジェネレータ

GPUコード
実行時
コンパイル

PG-Stromが追加で提供する
SQL処理ロジック。
拡張モジュールにより提供。

SQL実行ステージ



従来のデータフロー
本来は不要なレコードであっても、一度CPU/RAMへロードしなければ、要・不要を判断できないため、データサイズは大きくなりがち。

SSD-to-GPUダイレクトSQL
SSDから転送したレコードを、GPUの数千コアを用いてSQLワークロードを実行。CPUがロード/処理すべきデータ量を削減し、I/Oを高速化。

SQL処理に特化させる形でSSD/GPUの能力をフルに引き出し、従来のDBで大きなボトルネックだったI/Oを大幅に高速化。

中核機能② In-database機械学習

PL/CUDA関数の定義

```
CREATE FUNCTION my_logic(matrix, matrix)
RETURNS matrix
AS $$
LANGUAGE 'plcuda';
```

SVM (support vector machine)、Random Forest、k-means等のアルゴリズムを実装

SQLによる後処理
✓ テーブルJOIN ✓ ORDER BY
✓ Window関数 ✓ GROUP BY
etc....

GPU Kernel
ユーザ定義のCUDAコードブロック

実行結果の書き戻しとSQLによる加工
アルゴリズムをGPUで並列実行

Storage

一般的なCREATE FUNCTION構文を用いた関数定義

In-databaseで高度に計算集約的な機械学習アルゴリズムを実行。データをエクスポートせず結果だけを得る。今後はPython連携などを通じ深層学習フレームワークと連携

ビジネスモデル – 間接販売による弱点の補強

HeteroDBの強みと弱み

- ✓ 強み: 独自技術による圧倒的なデータ処理能力
 - ✓ 弱み: 営業網、保守体制、ソリューションの幅
- Slerとの協業で弱点を補完。自社は中核技術と得意分野に注力

未踏AD期間を通してSler/代理店xx社にアプローチうち、大手2社、ベンチャー2社との協業が進行中。

