

分散データベース上でのリダクション処理

新田 健二 † 矢口 堯 †
長尾 洋也 † 首藤 一幸 †

1. はじめに

データの解析では、データベース管理システムからデータを取得し、それを解析することになる。ここでもし、解析対象のデータよりも解析結果のほうが小さいならば、データが置かれている場所の近くで解析を行い、その結果のみを取得することで、取得のための通信を削減できる。これはつまり、リダクション処理である。

本研究では、分散データベース管理システムである Apache Cassandra¹⁾ の機能を拡張し、データの存在するノード上で指定した演算を実行し、元データの代わりに演算結果を返して集計する機能を実装する。この機能により、通信量の削減を達成する。

2. リダクション処理の設計・実装

Apache Cassandra はオープンソースの分散 key/value 型データストアである。Cassandra にはマスターとなるノードは存在しないため、データ群を取得する際には以下の手順で行う。

クライアントはまずクラスタを形成するノードのうちの一つをプロキシとしてアクセスする。その後、プロキシを間に介して必要なデータを保管しているノード（データノード）からデータを取得する。そして、プロキシに全ての必要なデータが集まったところでクライアントにその結果を返すことで、クライアントに

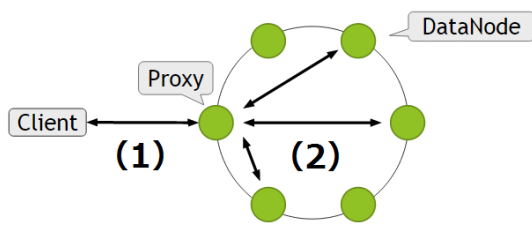


図 1 Cassandra のデータアクセス

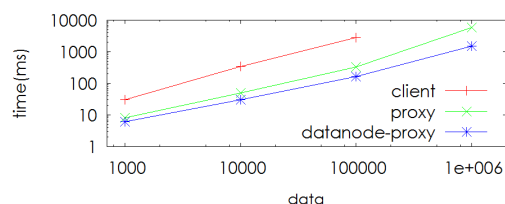


図 2 データ数と実行時間

データが届くことになる（図 1）。

広範囲のキーの値、つまり大量のデータを利用して演算を行う場合、(1) クライアント・プロキシ間、(2) プロキシ・データノード間の通信量は増大する。そこで本研究では (1), (2) の通信負荷の削減のため、二種類の機能を実装する。一つ目は、指定されたキーの値を全て取得する代わりに、プロキシがデータを収集した後、演算してからクライアントに結果のみを返す機能、二つ目は、データノード上であらかじめ特定の演算を行い、結果を求めるために必要な値のみをプロキシに返し、プロキシでそれらを集計して結果を返す機能である。

3. 評価

実装した機能による演算の処理時間を図 2 に示す。今回は指定したキー範囲の値の総和をとる機能を、それぞれクライアント側、プロキシ、データノードとプロキシで演算を行うように実装し、それぞれを実行時間 (ms) で比較した。取得するデータ数は 1000, 10000, 100000, 1000000 とした。

それぞれの手法を実行時間で比較すると、データの数に関わらず、データノード及びプロキシで演算を行った場合に最も実行時間が短くなっていることが確認できる。また、クライアントによる演算とプロキシでの演算の差がプロキシとデータノード及びプロキシの差よりも大きなものとなっているが、これはプロキシからクライアントへ集めた全てのデータを送る段階が非常に高負荷となるためと考えられる。

† 東京工業大学
Tokyo Institute of Technology

4. ま と め

本研究では, Cassandra を拡張して特定の演算を実行してから結果として返す機能を実装, 評価し, 性能の向上を確認した. 今後の課題としては, 演算の実行によるクラスタ全体の負荷の調査, より一般的な枠組みへの適用がある.

参 考 文 献

- 1) The Apache Cassandra Project
<http://cassandra.apache.org/>