RDBMS における SQL/OLAP 機能の実装と評価

原聖宣†　　木村耕治†　　近藤洋一†
†日立製作所　ソフトウェア事業部

1.はじめに

近年、データベースが社会全体に浸透するにつれ、データベース上に蓄積されたデータを解析するための手段が求められるようになっている。SQL/OLAP はそのような要望に対応するため、ISO において定められた国際規格であり、基本的なデータ解析を単独の SQL で行う点に特徴がある。従来の RDBMS で利用されてきた集約機能と比較して、SQL/OLAP は集約範囲の指定方法や統計処理用の関数を強化しているため、区間毎の集計や基本統計値を求めることが容易にできるようになっている。

本論文では、RDBMS における SQL/OLAP の効率的な実装方法と評価について述べる。

2.SQL/OLAP 機能概要

SQL/OLAP における OLAP 機能には、集約対象の範囲（WINDOW 枠）を決める WINDOW 句と、実際に集計などを行う WINDOW 関数がある（図 1）。WINDOW 句は、検索結果行集合を分割する指定である WINDOW分割句、WINDOW 枠内の行の並びを指定する WINDOW 順序句、及び WINDOW 枠句からなる。WINDOW 枠句はカレント行を基準として値範囲や行数で WINDOW 枠の前方及び後方の境界を指定する。WINDOW 枠は、カレント行が属する分割単位内で WINDOW 枠句の指定により定まる。WINDOW 関数にはランキング、割合などの他に、合計、平均、最大最小、標準偏差、分散、相関係数といった統計処理用の集約関数があり、WINDOW 句によって詳細に指定された集計範囲を入力として関数値を求めることができる。

例えば WINDOW 句を使用した図 2 に示す SQL が投入された場合、売り上げ表の商品コードごとに分割し、集計日でソートする。そして集計日の値範囲がカレント行から 2 ヶ月前までの行を集計範囲とする。その集計範囲の売上高が AVG 関数（平均）への入力になる。

Equipment and estimation of SQL/OLAP facility on RDBMS
Kiyonori Hara†, Kouji Kimura†, Youichi Kondo†
†Hitachi, Ltd. , Software Division

```
WINDOW 句 ::= WINDOW　WINDOW 名　AS　WINDOW 指定
WINDOW 指定 ::=
（WINDOW 分割句　WINDOW 順序句　WINDOW 枠句）
WINDOW 分割句 ::= PARTITION　BY　列指定
WINDOW 順序句 ::= ORDER　BY　ソート指定リスト
                  { RANGE }
WINDOW 枠句 ::=   {       } 範囲指定
                  { ROWS  }
                  { RANK  }
WINDOW 関数 ::=   {       } OVER　WINDOW 名
                  { 集約関数 }
```

図 1　構文規則

```
SELECT 商品コード, 集計日,
       AVG(売上高) OVER 商品3ヶ月売上
  FROM 売上表
  WINDOW 商品3ヶ月売上 AS
    (PARTITION BY 商品コード ORDER BY 集計日
     RANGE 00000200.  PRECEDING)
```

<売上表>

商品コード	集計日	売上高
S101	2003−01−01	300
S101	2003−02−01	100
S101	2003−03−01	200
S101	2003−04−01	200
S202	2003−01−01	400
S202	2003−02−01	100
S202	2003−03−01	100
S202	2003−04−01	200

カレント行

<検索結果>

商品コード	集計日	売上高 3ヶ月 平均
S101	2003−01−01	300
S101	2003−02−01	200
S101	2003−03−01	200
S101	2003−04−01	166
S202	2003−01−01	400
S202	2003−02−01	250
S202	2003−03−01	200
S202	2003−04−01	133

RANGE（カレント行を基準としたWINDOW枠）

図 2　WINDOW 句を使用した検索

また、GROUP BY 句拡張機能の ROLLUP、CUBE、GROUPING　SETS によって導出される結果も集約対象の範囲とすることができる。

3.OLAP 機能の実装

WINDOW 句の実行手段として、WINDOW 分割句及び順序句に指定された情報でソートする方法を採用した。ソート後の表（作業表とする）を使用して、WINDOW 分割句の指定に従った分割単位に WINDOW 枠を移動することができる。通常、WINDOW 枠の移動及び WINDOW 枠内の検索は、作業表を利用することとした。ここで、OLAP 機能の実装にあたって工夫した 2 つの点について説明する。

まず、WINDOW 枠を移動する度に WINDOW 枠内を検索しなくても、差分を求めることで高速化を実現した点である。WINDOW 枠内の先頭

の行を前方境界、最後の行を後方境界と呼ぶ。カレント行が次の行に移動する（WINDOW 枠が移動する）時に、WINDOW 枠によっては前方境界が変化しない場合がある。その場合を次のようにして考える。カレント行が n 行目（n=1,2,…）から n+1 行目に移動する時に、n 行目を基準とする WINDOW 枠内で、関数の入力となる値の集合を Rn とする。n+1 行目の関数の入力値を R’とした場合、Rn+1 は Rn に R’を加えた集合となる。Rn を入力とした時の関数値を F(Rn)と表現すると、 F(Rn+1)を求めるには通常、WINDOW 枠が変化するため、Rn+1 全体を検索する必要がある。だが前方境界が変化しない場合は、前回の関数値 F(Rn)に、後方境界が変化することによる差分 F(R’)を新たに加えることで求めることができる。つまり F(Rn+1) = F(Rn) + F(R’)が成立する。差分 F(R’)を求めれば作業表を検索する時のオーバーヘッドを削減できるので、前方境界が変化しない場合は差分を求める方法を適用した。標準偏差などのように複雑な計算をする関数についても、計算式上の中間結果について差分を求めることで、同じく前回の関数値を利用することができる。

前方境界が変化する場合でも、差分を求める方法を適用することは可能である。ただし、前方境界の差分を関数値から差し引くことになるため、精度の問題などで適用する関数の種類を絞る必要がある。

次に WINDOW 句の実行順序に関する最適化について説明する。WINDOW 句のソートキーは分割句と順序句により決定するため、WINDOW 句の実行順序を調整することによりソート処理回数を削減することができる。今、二つの WINDOW 指定 W1、W2 があったとし、W1 のソートキーが W2 のソートキーに先頭からソート方向も含めて一致する場合、W1 は W2 に含まれるということにする。そうすると、複数の WINDOW 句に包含関係を持たせることができて、この包含関係により WINDOW 句を分類することができる。各分類における最大の WINDOW 句に必要なソートを一回行えば、その分類に含まれる他の WINDOW 句はソートが不要である。従って、各分類における WINDOW 句の実行順序は、包含関係の大きい方から実行すればよいことになる。また、各分類の実行順序は WINDOW 句の入力となるリレーションの持つソートキー及び WINDOW 句の後に処理する ORDER BY 句のソートキーと比較して決定することで、ソート回数を削減することができる。

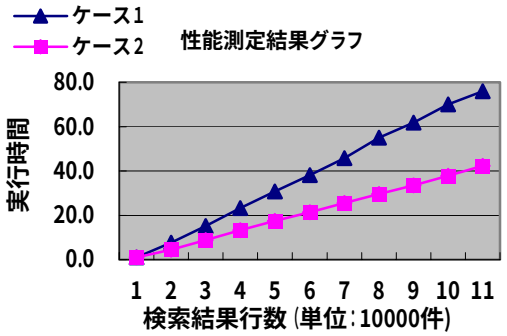
4.性能評価

WINDOW 句を使用した検索の性能評価として、以下の 2 ケースの SQL 実行時間を測定し、結果をグラフにまとめる。（図 3）

ケース 1：WINDOW 枠内をカレント行ごとに検索して関数値を導出するパターン。

ケース 2：WINDOW 枠の差分から関数値を導出するパターン。

上記 2 ケースの SQL は WINDOW の範囲指定を除き同一である。ケース 2 の方が、検索結果件数の増加に伴う実行時間の増加率が小さい事から、高速化の効果が現れていることが分かる。



実行時間はケース 1 の 10000 件を基準とした相対値

図 3　WINDOW 句性能測定結果グラフ

5.まとめ

データベース上に蓄積されたデータの基本的な解析を単独の SQL で実現できる OLAP 機能を実装した。

WINDOW 句の実行時、WINDOW 分割句及び順序句に指定された情報でソートする方法を採用した。

さらに性能を確保するため、実行順序の最適化においてソート処理を削減する工夫や、WINDOW 枠の差分を求めることで高速化をはかる工夫を行った。

参考文献

[1]ISO/IEC 9075-1,9075-2,9075-5:1999/Amd.1:2000, Database Languages - Amendment 1: On-Line Analytical Processing
[2]ISO/IEC 9075-1:1999/Information technology – Database Languages – SQL – Part1: Framework (SQL/Framework)
[3]ISO/IEC 9075-2:1999/Information technology – Database Languages – SQL – Part2: Foundation (SQL/ Foundation)
[4]JIS X 3005-1:2002 データベース言語 SQL 第 1 部: 枠組み(SQL/Framework)
[5]JIS X 3005-2:2002 データベース言語 SQL 第 2 部: 基本機能(SQL/ Foundation)