

ORDB におけるプラグイン関数並列実行の実現と評価

4 A a - 5

近藤 隆憲 下川 隆義 亀城 嘉人 正井 一夫

日立製作所ソフトウェア開発本部

1.はじめに

日立製作所では、RDBMS、HiRDB の開発を行っている。HiRDB は、Shared Nothing 方式を採用するなど、高い並列性によるデータ操作の高速化を実現するためのアーキテクチャを採用している。現在、筆者らは HiRDB をオブジェクト拡張した HiRDB Universal Server (以下、HiRDB US) を開発している。

データベース管理システム (以下、DBMS) においてある決まった処理を行うことを定義・実行するものとして、ストアド・プロシジャがある。(以下、このように DBMS において一連の処理手続きを登録・実行するものをルーチンと呼ぶ。) HiRDB では、ストアド・プロシジャをサポートしている。しかし、そこには解決すべき課題があった。

その課題とは、SELECT 結果の演算などデータの後処理については、SQL 受付サーバと呼ばれる一つのサーバにおいてのみ実行されるため並列性がない、ということである。このような場合には、HiRDB が持つ並列処理が可能であるという利点が生かされないということになる。

そこで、HiRDB US においては、プラグイン機構の導入を行い、SQL 受付サーバより他のサーバへ実行要求を出すようにし、ルーチン実行の並列化を可能にした。

また、プラグイン関数は、コンパイルされたオブジェクト・モジュールを DBMS に組み込み、それを実行するため、ルーチン処理の実行速度向上も実現することができた。

以下で、プラグイン機構の実現方式を示し、その評価を行う。

2.HiRDB US について

2.1.HiRDB US のサーバ構成

HiRDB US のサーバ構成を説明する。

HiRDB US は図 1 に示した分散マルチサーバ方式のソフトウェア構造を持っており、サーバを各プロセッサに分散配置することによって並列性を実現している。各サーバの役割について次に説明する。

- (a)SQL 受付サーバ クライアントから投入された SQL を受け取り、解釈、最適化して各 DB 処理サーバに処理を要求する。また、DB 処理サーバからの処理結果を収集してクライアントに返却する。

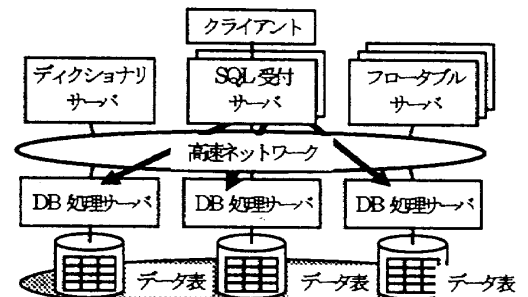


図 1 HiRDB US のサーバ構成

- (b)DB 処理サーバ 表を格納したデータ格納単位ごとに存在し、そのデータに専任でアクセスを行う。また、ソートなどの演算処理を行う。
- (c)フローダブルサーバ 表を持たない DB 処理サーバであり、負荷分散を目的としてプロセッサに配置され、ソート、ジョインなどの CPU ボトルネックとなる処理を担当する。これによって、Shared-Nothing 方式の欠点であるデータの偏在にともないサーバ負荷が偏る問題の解決をはかる。

2.2.プラグイン機構

HiRDB US では、プラグイン機構を採用している。プラグイン機構とは、ユーザが定義したあるデータ型 (抽象データ型: Abstract Data Type (ADT) と呼ばれる) に対して特定の処理を行う関数 (プラグイン関数と呼ぶ) をネイティブコードとして DB に組み込むための機構である。

プラグイン機構の概要を図 2 に示す。プラグインに対する外部仕様は ADT 定義と呼ばれるデータ定義によって規定される。そして、その実装はプラグインモジュールにより行われる。また、プラグイン関数のインターフェースについては、一部 ADT 定義で行われ、一部プラグインのインターフェース定義を行う、プラグイン組み込み定義を用いて行われる。

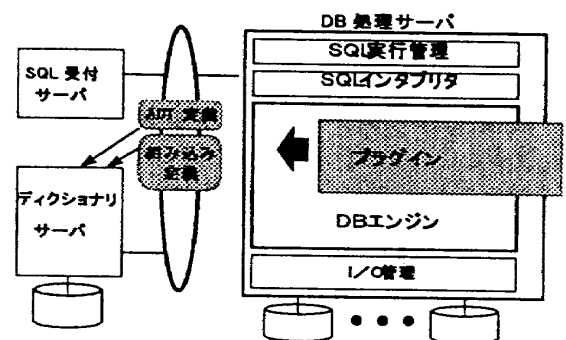


図 2 プラグイン機構概要図

Implementation and Evaluation of Parallel Executable
Plug-in Function on ORDB

Takanori Kondo, Takayoshi Shimokawa,
Yoshito Kamegi, Kazuo Masai
Software Development Center, Hitachi Ltd.

3. プラグイン機構の利点

3.1. 複雑な後処理の実現

図3の処理のように、SELECTを行った結果に対しての後処理を行う場合には、評価関数としてデータに依存した複雑な処理を行うのは困難であった。

単純な演算については、SQL文の中に演算処理を直接入れることで処理が可能であるが、その場合SQL受付サーバのみで演算が実行されるため処理速度の面において問題があった。

```
SELECT 評価関数 FROM T1
FROM T1 WHERE C1<X
```

図3 プロシジャの例1

HiRDB USにおいては、特定のデータに対してADTを宣言し、同時にプラグイン関数を定義、実装することで複雑な評価関数の並列実行を実現できる。

これにより、莫大なデータからの統計情報を得るといったデータマイニングなどに対する応用に好適であると考えられる。

3.2. 関数の再利用性

これまでのストアド・プロシジャにおいては、定義された関数が多くのDB上で実行されるということが少なかった。つまり、関数の再利用性は低かった。

プラグイン関数は、ある特定のデータ型(例えば、SGMLテキストデータ型)に対して実現されるものであり、そのデータ型を扱うニーズのあるところでは、共通してDBにインストールし、利用することができるという利点がある。

すなわち、プラグイン関数においては、ストアド・プロシジャでは実現されなかった高い再利用性が実現されている。

3.3. 関数実行の高速性

プラグイン関数は、コンパイルされたモジュールをDBMSに組み込み実行を行うため、高い実行速度を実現している。

図3における評価関数の例において、単純な演算などの処理については、ストアド・プロシジャでも評価関数の記述は可能であるが、プラグイン関数におけるような高速性は、望むことができない。

3.4. 並列実行

HiRDB USにおいては、プラグイン関数は、SQL受付サーバとDB処理サーバにインストール可能である。

実際には、主にDB処理サーバにインストールされる。これにより、HiRDB USの並列性が生かされ、高いスケラビリティが実現された。

以上により、Shared Nothing方式を採用したHiRDB

USが持つデータ操作の並列実行の高速性に、プラグインが持つ高速性が加えられることとなった。

3.5. データに応じた機能拡張

プラグインは、特定のデータ型に対して作成される。よって、DBに新たなデータを扱わせる場合には、そのデータ型のためのプラグインをDBMSにインストールすることでデータ操作のための機能を容易に拡張することができるという利点がある。

よって、最近の静止画像、動画、音声等、種々のマルチメディアデータについてのDBに対する、効率的なデータ格納と検索・更新処理というような新しいニーズに対応することが容易である。

多様化するマルチメディアデータの処理は複雑化しており、柔軟なアプリケーション・プログラムの迅速な開発が必要である点についても、再利用性の高いプラグイン機構が重要な役割を持つこととなる。

4. 評価

プラグイン機構により、ルーチンとしてコンパイルされたモジュールがDBMSに組み込まれ、実行されることとなり、これにより、ルーチンがSQL文により記述されインタプリタとして実行されるストアド・プロシジャに比べ、高速な実行を実現することができた。

5. まとめと今後の課題

本稿で述べたプラグイン機構を実現することにより、ルーチン処理の実行速度向上を実現した。

また、それと同時に、プラグイン関数の持つ再利用性などの利点も実現された。

DBMS全体としては、従来のHiRDBのオブジェクト拡張とプラグイン機構の導入を行ったORDB、HiRDB USが実現されたこととなる。

今後は、より多様なデータ型に対するプラグインの開発を行い、文書や画像などのマルチメディアデータの蓄積や操作、また、並列実行性を生かした大規模データの検索などのニーズへの対応していくことが課題である。

参考文献

- [1] 正井他:「更新処理を並列実行するUNIX向けDBMSを開発」, 日経エレクトロニクス, 1995.2.27(No.630)
- [2] 根岸他:「並列リレーショナルデータベースシステムHiRDBの概要と基本技術」, 電子情報通信学会 信学技報 DE95-79, 1995
- [3] 藤原他:「並列RDBシステムにおける通信機能の実現方式」, 情報処理学会台49回全国大会講演論文集, 7W-3 1995
- [4] 金丸他:「並列DBサーバHiRDBにおけるSQL最適化処理方式」, 情報処理学会第52回全国大会講演論文集, 3Q-5 1996
- [5] 八高他:「異種プラットフォーム混在環境下での並列DBMS(HiRDB)の実現」, 情報処理学会第54回全国大会講演論文集, (3) 1R-3 1998