

6F-3

UNIX用メモリDBMSのWWWサーバ用
高速インタフェースの検討^{*1}

加藤 謹詞 高橋 嘉裕 梅田 昌義^{*2}
NTT 情報通信研究所^{*3}

1. はじめに

インターネット、イントラネットの本格的なEC等への利用を控え、WWWサーバは、益々、高機能、高性能を要求されている。

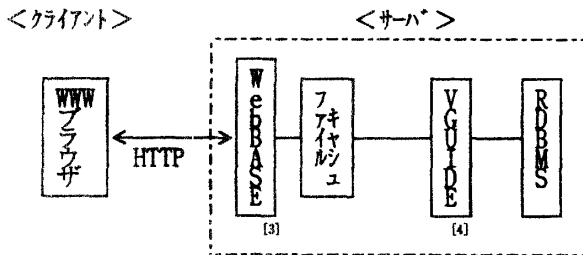
ECへの適用では、基幹システムとのデータの共用化のために、RDBとの連携が必須であるが、WWW-RDB連携^[1]では、トランザクション継続の問題や性能的な問題がある。

そこで、本稿ではWWW-RDB連携に関して、大幅な性能向上を目的に、UNIX用メモリDBMS^[2]をWWWサーバ用のキャッシュサーバとしての適用、及び、その高速性を生かすキャッシュ方式についての検討結果を報告する。

尚、具体的な評価のために、膨大なアクセス回数があるNTT DIRECTORYシステムへの利用を前提とする。

2. NTT DIRECTORYシステムの
現状と問題点

(1) NTT DIRECTORYシステムの概要
NTT DIRECTORYシステムの概要を図1に示す。



WebBASE : NTTが開発したWWWサーバ

VGUIDE : NTTが開発した簡易TPモジュール

図1 NTT DIRECTORYシステムの概要

NTT DIRECTORYのアクセス状況は、以下の通りである。

- ①更新契機は、1回/日である。
- ②アクセスパターンは参照が殆どである。
- ③同一のSQL文が繰り返し使用される。
- ④約百数十万ヒット/日のアクセスがある。

(2) NTT DIRECTORYシステムの性能向上

NTT DIRECTORYシステムでは、アクセス数の増大に対処するため、幾つかの性能向上対策を実施している。その一つとしてRDB検索へのキャッシュ適用^[5]が報告されている。

RDB検索キャッシュの概要を図2に示す。

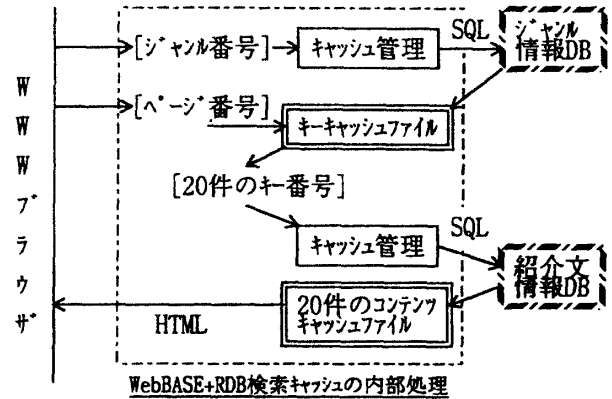


図2 RDB検索キャッシュの概要

【処理の流れ】

- ①WWWブラウザから、選択された「ジャンル番号」から、キャッシュ管理は、既に検索結果（キーキャッシュ）がキャッシュされているかを調べる。
- ②キーキャッシュが存在しない場合「ジャンル番号」を条件に検索文をDBMSに発行し検索結果を得る。
- ③「ページ番号」から20件分のキー番号をキーキャッシュより取得する。
- ④キャッシュ管理は、20件分のキー番号に対応する紹介文情報（コンテンツキャッシュ）がキャッシュされているかを調べる。
- ⑤コンテンツキャッシュが存在しない場合、20件分のキー番号に対応する紹介文情報を取得しWWWブラウザに返却する。

(3) RDB検索キャッシュの課題

RDB検索キャッシュでは、性能向上が確認されているが以下の課題がある。

- ①コンテンツキャッシュのヒット率が約70%と低い。
 - ②コンテンツキャッシュは、20件単位で管理するためデータの重複が多い。
 - ③キャッシュファイルの増加に伴い、キャッシュファイル自身の検索やディスクアクセスの時間が増大する。
- 上記①～③の問題は、コンテンツの増加とともに一層、顕著に現れてくることが予想される。

3. メモリDBMSを使用した高速化方式

上記2. (3)の課題を解決し、更に、今後のアクセス回数やコンテンツの増大に対応するために、メモリDBMSを使用した図3に示す方式を提案する。

本方式では以下の特徴がある。

- ①メモリDBMSの利用
- ②データキャッシュ機能
- ③SQLキャッシュ機能
- ④カーソルキャッシュ機能
- ⑤検索結果の一括返却機能
- ⑥データ更新時期の指定、及び、更新データのタンキング機能
- ⑦DLLプロシージャインタフェースによるメモリDB領域の直接参照
- ⑧一括データ更新後のキャッシュの復元

*1:High-speed cache methods of memory resident DBMS for WWW server system.

*2:Kinji Kato, Yoshihiro Takahashi, Masayoshi Umeda

*3:NTT Information and Communication Systems Laboratories

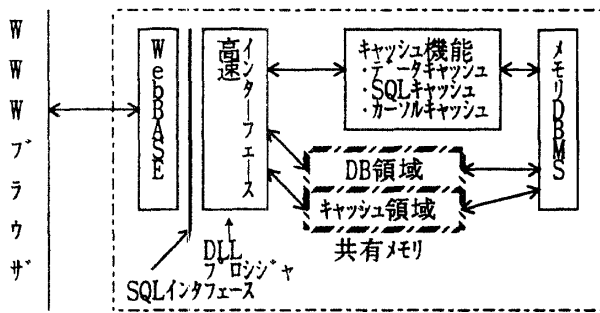


図3 メモリDBMSを使用した高速化方式

以下では高速キャッシュ機能に関して説明する。

4. 高速キャッシュ機能

WWWブラウザからの検索要求は、SQL文の形態で図4に示す処理を実行し、HTMLに変換し返却する。

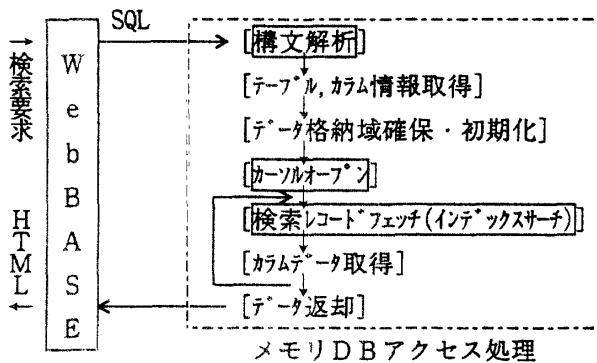


図4 メモリDBアクセス処理

処理コストが大きいのは、構文解析処理、カーソルオープン処理、そして、インデックスサーチ処理である。高速キャッシュ機能では、これらの処理を削減・軽減することにより高速化する。

(1) データキャッシュ機能

メモリDBMSでは、実データを基本的に共有メモリ上に、格納しておき、高速検索を実現している。

データキャッシュ機能は、検索時のインデックスサーチのコストを削減するために、検索結果をキャッシュ領域に格納する機能を実現する。更に、格納コストを評価し、以下の2種類のどちらかの形態でキャッシュ領域へ格納する。

- a. 実データ
- b. 検索レコードへのアドレス

(2) SQLキャッシュ機能

SQLキャッシュ機能は、構文解析処理を削減することにより、高速化を実現する。NTT DIRECTORYでは、同一のSQL文が繰り返し発行されることから、一度、発行されたSQL文を検索結果レコードのキャッシュへのアドレスと共に、メモリDBのテーブルに格納し再利用する。図5に、SQLキャッシュとデータキャッシュの処理を示す。

発行されるSQL文とSQLキャッシュテーブル内のSQL文は、ユニークに対応するため、メモリDBMSのハッシュ検索を用い高速検索が可能である。

(3) カーソルキャッシュ機能

上記(1)では、キャッシュ領域を必要とするため、格納コストが大きくなる問題がある。そこで再利用率がある程度低いSQL文に関しては、キャッシュ領域を使用しないカーソルキ

ャッシュ機能を実現する。

カーソルキャッシュ機能では、カーソルのオープン情報を保持しておき同一のSQL文が発行された際に、メモリDBMSのトランザクションIDによる再接続機能を利用し処理を軽減する。

SELECT KEYNO FROM SERVICE

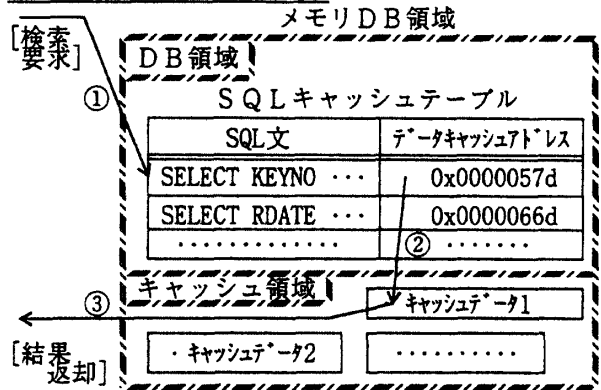


図5 SQLキャッシュとデータキャッシュの処理

SELECT KEYNO FROM SERVICE

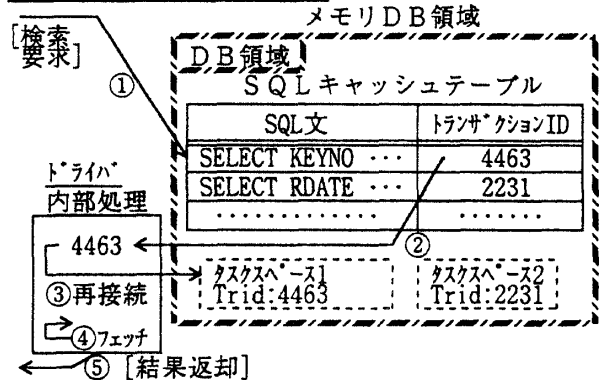


図6 SQLキャッシュとカーソルキャッシュの処理

5. まとめ

本稿では、メモリDBMSを高トラフィックなWWWサーバに適用するための高速なキャッシュ方式を検討した。検討の結果、①検索データを実データとアドレスを格納コストにより評価しキャッシュする機能、②検索要求のSQL文と検索データのキャッシュへのアドレスを組でキャッシュする機能、③トランザクションIDによるカーソル情報の再利用によるキャッシュ領域を使用しない機能、を提案した。

今後は、定量的な評価と共に、最適なキャッシュ機能の組み合わせのコスト評価を検討する予定である。

本検討を進めるにあたり、NTT ソフトウェア研究所 元田敏浩研究主任 のご協力に感謝致します。

参考文献

- [1]徳丸, 元田, 黒川: WWWによるデータベース検索方式の検討, pp. 105-112, ADBS95
- [2]黒岩, 板倉, 岡田: UNIX用メモリDBMSの開発コンセプト, 情報処理学会第55回全国大会, 1997-9
- [3], [4]Software Dreamers: "WebBASE, VGUIDE page", <http://robin.sl.cae.ntt.co.jp/>
- [5]元田, 川崎: WWW-RDB連携におけるキャッシュ制御について, 情報処理学会第54回全国大会, 1997-3