

インデックス I/O 多重制御方式

5P-4

黒岩 淳一*

中村 敏夫**, 村田 達彦**

* NTT情報システム本部

** NTT情報通信網研究所

1. はじめに

リレーショナル・データベース(以下RDBと記す)管理システム(同RDBMS)においては、インデックスを用いて二次記憶装置上に存在するRDBデータの検索範囲を限定する事で処理時間の短縮、スループットの向上を実現していることが多い。大規模RDBでは、検索処理実行時にこのインデックスを二次記憶装置より主記憶装置上に読み込むことが必要だが、その時RDBMSがインデックスの状態を管理するための共通制御表に対する更新も必要になる。ここで、制御表をインデックスの読み込み処理完了まで更新ロックしている場合、この区間で他プロセスによる制御表へのアクセスが制限されるため、スループットの向上に限界があった。

今回、インデックス・データの二次記憶装置とのデータI/O期間中制御表のロックを一次的に解除する制御表管理方式に変更した場合の効果を検証したところ、従来比3倍のスループット向上を確認できた。

2. 従来の制御表管理方式

RDBMSでは、検索処理の性能を向上するためインデックスを利用することが多い。しかし、近年の急速なRDB普及かつ大規模化に伴い、一層の高性能化が要望されている。

RDBMSでインデックスを使用する場合、二次記憶装置からインデックス・データをバッファメモリ上に読み込んだ後、該データ中の目的レコードの格納位置情報から二次記憶装置のアクセス箇所を限定し高速に検索処理を行う。

このインデックス・データをバッファメモリ上に読み込む過程で、RDBMSがバッファメモリの状態を管理する制御表に更新がかかるため、処理の矛盾を防ぐための更新ロックが必要であった。

ところが、二次記憶とのI/O期間を含めた読み込み処理が完了するまで、制御表を更新ロックしている場合、他プロセスによる該制御表へのアクセスが制限され、処理がシリアル化されてスループットが低下していた。これは大規模なRDBにとって大きな性能問題になり得る。この制御表管理方式を図1に示す。

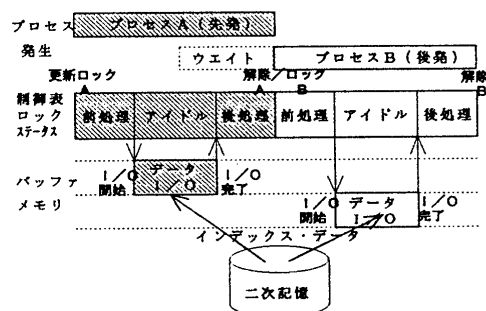


図1 従来の制御表管理方法

図1ではプロセスAが処理を開始後、ある時間が経過した時に別のプロセスBが発生した場合について例を示している。

プロセスBが制御表にアクセスしようとした時点では、該制御表は既にプロセスAによって更新ロックがされているため、I/O中であるプロセスAの制御表に対する処理が実質的にアイドル状態であっても、ロックが解除されるまでプロセスBは該制御表に対してアクセスできずウェイト状態となる。この様にバッファメモリを管理する制御表への各プロセスのアクセスがシリアル化されるため、更なる性能向上には限界があった。

3. インデックス I/O 多重制御方式

上記問題を解決するための方法として、二次記憶からのインデックス・データI/O区間のみ、制御表に対するロックを解除し、I/O完了後に再度制御表を更新ロックする方法が考えられる。

このため、ロックが解除されているI/O区間内に他プロセスによる該制御表へのアクセスが可能となるため、従来に比べてCPUのアイドル時間の減少、さらに発行可能なI/O数が増大することにより、システム全体としてのスループット的大幅向上が期待される。

図2に、このインデックスI/O多重制御の概要を示す。

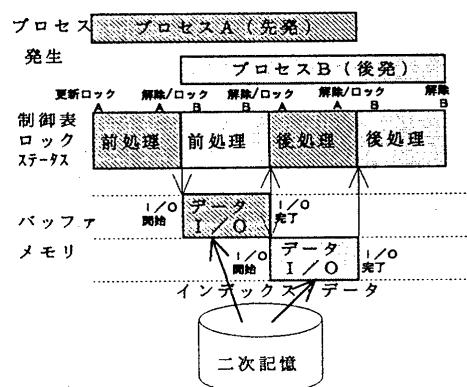


図2 インデックスI/O多重制御

図2は、図1と同様な処理をインデックスI/O多重制御により実現した場合の例を示している。

従来方法の図1とは異なり、データI/O前後の前処理/後処理の完了時点で制御表の更新ロックが解除されるため、プロセスBはプロセスAのデータI/O背景で制御表にアクセスし、データI/Oのための前処理を行うことが可能になる。

すなわち従来方式では、I/O背景でCPUがアイドル状態となっていたが、インデックスI/O多重制御ではI/O背景で別プロセスの処理が並走でき、CPUの効率的利用、さらには検索処理のスループット向上が期待できる。

Effect of Multiple Access Control for Index I/O

* Junichi KUROIWA,** Toshio NAKAMURA,** Tatsuhiko MURATA

* Information Systems Headquarters, NTT

**Network Information Systems Labs, NTT

しかし、本インデックス I/O 多重制御を実現するに当たり、考慮すべき問題が存在する。以下の 3 点である。

① I/O の衝突

処理可能な I/O 数が増大した結果、二次記憶装置へのアクセス時に I/O が競合・衝突する事によるボトルネックが顕在化する可能性

② 再ロックによるオーバーヘッド

ロック解除/再ロックの CPU 負荷増、再ロック時のウェイトによるオーバーヘッド

③ 複数プロセスで同一データ読み込み

複数プロセスで同一インデックス・データ（ページ）をバッファ・メモリ上に重複して読み込み、各々独自に更新を行うと処理に矛盾が生じる可能性があるため、チェック機構が必要であるが、そのためのオーバーヘッド

①の対処のため、インデックスを格納する論理的に一つの領域を、物理的に複数の二次記憶装置に分割して格納することにより、アクセス時の I/O 衝突を軽減できる様に配慮した。

②、③については、インデックス読み込み処理に占める I/O の割合が大きいこと、本制御方法による再ロック、同一データ読み込みのチェックによるオーバーヘッドは、全体としては誤差レベルであり問題にならないと考えた。

従来の制御表管理方法では、制御表ロックによりインデックスのデータ I/O がシリアル化された結果として性能向上に限界が生じていた。しかし、本インデックス I/O 多重制御により、I/O 背景で CPU が効率的に動作可能となり、処理可能な I/O 数も増加することができることから、スループットの大幅向上が期待できる。

4. インデックス I/O 多重制御の効果

インデックス I/O 多重制御の効果を確認するため、表 1 に示す 1 レコード当たりの定義長 100 バイト、カラム数 10、70 万レコードの RDB (TPC-A/B ベンチマークで用いられる ACCOUNT テーブルに相当 [1] [2]) について従来方法との比較を行った。

表 1 試験に用いたテーブルの構成

テーブル名	ACCOUNT		
カラム名	属性	長さ	備考
ACCOUNT_ID	INTEGER	4	キーカラム
BRANCH	INTEGER	4	
A_BALANCE	DECIMAL(15,0)	8	
FIELD4	CHAR(24)	24	
FIELD5	CHAR(10)	10	
FIELD6	CHAR(10)	10	
FIELD7	CHAR(10)	10	
FIELD8	CHAR(10)	10	
FIELD9	CHAR(10)	10	
FIELD10	CHAR(10)	10	

定義レコード長 100 バイト

レコード件数 70 万レコード (70 M バイト)

なお、測定に当たりインデックス I/O 多重制御による効果を特定する目的から以下の点について考慮した。

・インデックス処理性能のみを測定

RDB データ自体へのアクセス処理は極力排除することを目的として、インデックス I/O のみに完結する様に SQL を発行

・二次記憶装置の I/O ネットの影響排除

インデックス格納二次記憶の分散化

測定環境の概略は以下の通り。

ホスト CPU DIPS-11/45EX
二次記憶 DK GEMMY (7 台に分散)

図 3 に、上記測定環境において従来の制御表管理と本インデックス I/O 多重制御の性能向上について示す。

なお、測定は表 1 の被測定テーブルに対し、キー値をランダムに変化させて繰り返しアクセスするバッチ AP を、多重度 1~6 まで変化させて実行した。

図 3 では、従来制御表管理と本インデックス I/O 多重制御に基づく処理ロジックで、ほぼ定常状態において単位時間当たりにインデックスに対して発行した I/O 数比で結果を示している。

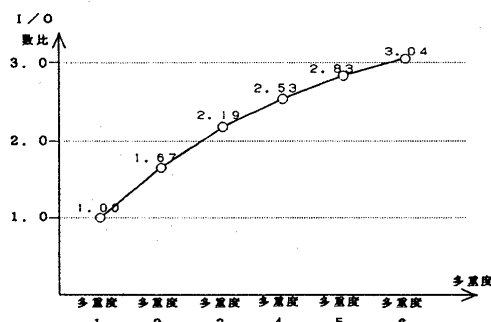


図 3 インデックス I/O 多重制御の効果

本インデックス I/O 多重制御により、スループットが大きく向上していることが、図 3 に示す I/O 数比の増加により明かである。

なお、多重度が上がるにつれて従来方式に対する性能向上の度合いが低下してきているが、インデックス I/O 多重制御による I/O 数増の結果、二次記憶装置への I/O が衝突してきたためである。また、今回測定したバッチ AP の多重度では、性能 (I/O 数比) 的に飽和領域に入っており、測定時の環境を変更 (多重度増、インデックスを格納する二次記憶装置の分割数増) することにより、最終的には 3 倍を超える効果を実現することも可能と思われる。

また、再ロック/同一データ読み込みチェック等のデメリット (オーバーヘッド) は、多重度 1 の結果より性能劣化が見られないことから、問題ないことも確認された。

5. おわりに

二次記憶装置からの I/O 背景において、RDBMS がインデックスの状態を管理するための共通制御表に対する更新ロックを一次的に解除し、その区間に他プロセスの制御表アクセスを可能とするインデックス I/O 多重制御の効果について検証を行った。

今回の検証で明らかになったように、本インデックス I/O 多重制御の方式自体は極めて単純なものであるが、効果としては極めて有効であることが確認できた。

ただし、二次記憶装置へのアクセス時における I/O の衝突が性能上のボトルネックとなるため、インデックスを格納する領域を物理的な複数の二次記憶装置に分割して格納できるかが性能向上のポイントとなる。

本インデックス I/O 多重制御により、測定環境によりさらに性能向上の余地はあるものの、従来比約 3 倍のスループット向上は達成できた。

【参考文献】

- [1] Anon. 他 "A Measure of Transaction Processing Power" DATAMATION, vol. 31, No. 7, pp. 112-118, Apr. 1, 1985.
- [2] 渡辺 編・訳 「OLTP システム」 pp. 233-277, マグロウヒル 1991 年 7 月。