

メモリDBにおけるデータ保証種別に応じたリカバリ方式

7W-8

小林伸幸 森下慎次

NTT情報通信研究所

1 はじめに

近年のDBの適用分野の拡大に伴い、従来より高速なDBアクセスへの要求が高まっている。これを実現する方式にメモリDBがある。本方式では全データをメモリ上に展開するために、参照のためのI/Oが不要になり、高速な処理が実現できる。しかし、更新処理では、システム障害等に備え処理結果を2次記憶に保存する必要があり、そのためのI/Oが処理速度の劣化要因となっている。

本稿では、リカバリ時に、障害直前の状態に戻す必要がないデータに着目する。このようなデータに対して、どのような更新処理の場合に、2次記憶からリカバリする必要がないかを整理する。そして、必要な更新処理の結果のみ2次記憶に取得することにより、I/O回数を削減し、高速なDBアクセスを実現するリカバリ方式を提案する。

2 データ保証の種別について

メモリDB上のデータの中には、システム障害等の2次記憶からのリカバリでは、特定の更新処理の結果のみを保証すればよい場合がある。

そのような例を図1に示す。この例では放映開始時および終了時の変更処理結果はメモリ上のみ更新すれば良いが、新規登録時の更新結果については2次記憶からのリカバリで保証する必要がある。このように、2次記憶からのリカバリで保証する必要のない更新処理があるデータの種別を、本稿ではテンポラリデータと呼び、更新結果を常に保証する必要がある種別をパーシステントデータと呼ぶ。

A Recovery Method of Memory Resident DB  
 Considering Data Guarantee Level.  
 Nobuyuki KOBAYASHI, Shinji MORISHITA  
 NTT Information and Communication Systems  
 Laboratories.  
 1-2356 Take, Yokosuka, Kanagawa 238-03, Japan

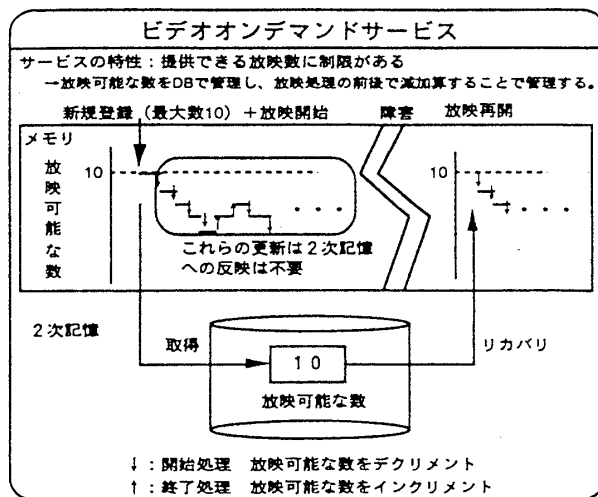


図1 テンポラリデータ

この様にテンポラリデータに対する更新処理には、処理結果を2次記憶からのリカバリで保証を必要とする場合と不要な場合がある。以下に整理する。

(1) 挿入処理 (INSERT) ・ 削除処理 (DELETE) ・

新規登録等のように対象となるデータ自体の存在の有無に関わるため、2次記憶上に処理結果を反映し、処理結果をリカバリで保証する必要がある。

(2) 更新処理 (UPDATE)

図1の例でも、放映可能な最大数を変更するような更新は、2次記憶からのリカバリで保証する必要がある。このように、更新処理では保証する場合としない場合があり、本稿では、前者を保証処理と呼び、後者を非保証処理と呼ぶこととする。

以上の2次記憶からのリカバリで保証する必要性を、表1に整理する。

表1 データ保証の種別

データ種別	テンポラリデータ	パーシステントデータ
更新種別		
挿入：INSERT	データ保証要	データ保証要
削除：DELETE		
更新：UPDATE	保証処理	
	非保証処理	

### 3 リカバリ方式

本章では、2章のデータ種別に基づいた、高速なアクセスを実現するリカバリ方式を検討する。

#### (1) パーシステントデータのリカバリ方式

パーシステントデータについては、従来と同様に、総ての更新結果を2次記憶からのリカバリで保証する必要がある。更新処理内におけるI/O回数を削減し、高速なアクセスを実現するためのリカバリ方式に、ファジーチェックポイント方式<sup>[1]</sup>（以下ではFCP方式と略す。）がある。本方式の概要を図2に示す。

本方式では、更新処理内からのI/Oは、更新後ログの出力のみで済む。本稿では、パーシステントデータに本方式を適用する。

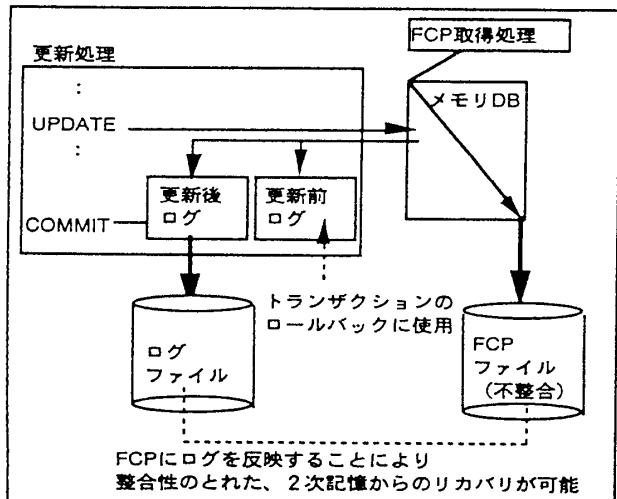


図2 パーシステントデータのリカバリ方式

#### (2) テンポラリデータのリカバリ方式

上記のFCP方式では、2次記憶からのリカバリで整合性を保証するために、ログ情報を必要とする。そのため、テンポラリデータには不要なログ情報の取得が必須になり、I/O回数を削減できない。

テンポラリデータの特徴を活かし、2次記憶を選択的に更新する方式にトランザクション内更新方式がある。本方式の概要を図3に示す。本方式では、データ保証の種別に応じて、更新処理内で2次記憶の更新を選択的に行えるために、テンポラリデータの特徴に対応できる。

FCP方式とトランザクション内更新方式で必要に

なる更新処理内でのI/Oを表2に整理する。

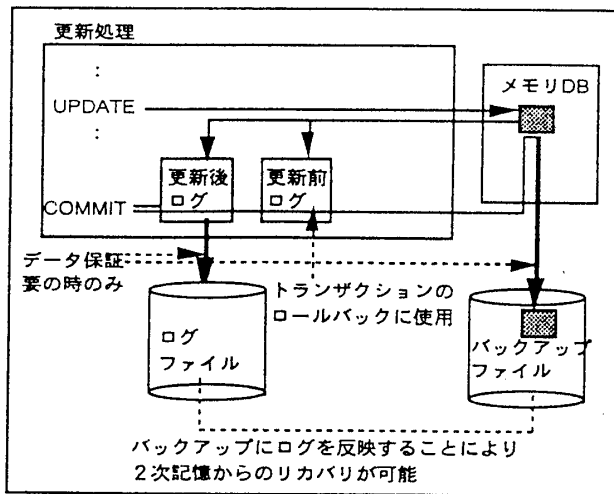


図3 テンポラリデータのリカバリ方式

表2 I/O処理

		更新前 ログ出力	更新後 ログ出力	バックア ップ更新
トランザク ション内 更新方式	保証処理	×	○	○
	非保証処理	×	×	×
FCP方式 (パーシステントデータ)		×	○	×

×：不要 ○：必要

表2より、テンポラリデータにトランザクション内更新方式を適用すると、テンポラリデータの非保証処理については、パーシステントデータより高速なアクセスが実現できる。しかし、保証処理については、パーシステントデータよりもI/Oが増大する。ただし、このような保証処理は、アプリケーションの提供条件の変更等に伴うものであり、発生頻度は低い。そのため、テンポラリデータには、トランザクション内更新方式が有効であると考えられる。

### 4 おわりに

本稿では、メモリDBにおけるデータ保証の種別を設定し、更新処理内におけるI/O回数を削減するリカバリ方式を提案した。今後は、本方式の性能評価を行う予定である。

#### 【参考文献】

[1]小林他：「高速なオンライン処理のためのDBバックアップ方式」、第46情処全国大会,4G-9(1993)