

# データベース・オンライン保守の一方式

4G-8

梅本 佳宏 中村 仁之輔 森下 慎次

NTT情報通信網研究所

## 1. はじめに

ネットワーク・サービス等、24時間連続無中断運転が必要なシステムに用いられるデータベース(DB)では、その保守もサービスを継続しながら行う機能(DBオンライン保守)の実現が要求されている。

現状、DBオンライン保守としては、主にデータのロード、アンロード、また、一部のDBMSでは再編成が実現されつつある。

しかし、一般に、DBの定義(スキーマ)を変更する再構成や、データ実体のファイル内移動を伴う再編成の処理は、サービスを停止したオフライン状態で行う必要があり、連続無中断運転を行うシステムの運用の妨げとなっている。

DBオンライン保守の実現の問題点としては、

- (1) アクセス中にスキーマ定義を変更したり、データ実体の操作を行う技術が確立されていない。
- (2) DB保守には多大なCPU資源が必要で、サービスと並行してDBオンライン保守を行うとスループットの低下が起こる。

等が考えられる。

本報告では、24時間連続運転が必要なシステムは、高信頼化のため、サービス系とスタンバイ系の2重化構成をとっていることが多いことに着目し、スタンバイ系で保守処理を行い、サービス系でサービスを継続することにより、オンライン状態でDB保守を行う方式を提案する。これにより、サービス系への負荷を軽減することができ、保守時間も短縮される。

## 2. 前提

### 2.1 マシン構成

本報告で用いるマシンのモデルを図1に示す。

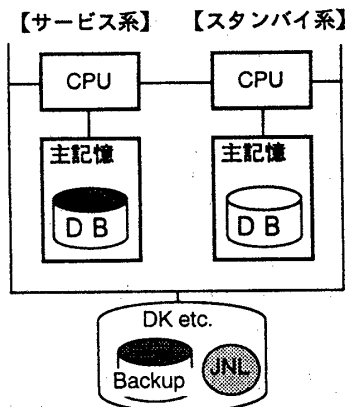


図1. マシン構成

A Method of Online Maintenance for Databases  
 Yoshihiro UMEMOTO, Jinnosuke NAKAMURA  
 and Shinji MORISHITA  
 NTT Network Information Systems Laboratories

- (1) サービス系とスタンバイ系が、それぞれ主記憶をもつ疎結合方式

- ・ CPU間は、直接通信/データ転送が可能
- ・ 各CPUは、バスを介してディスク装置に接続されている

- (2) DBは、アクセス高速化のため、主記憶にロードされている。(メモリ・データベース方式)

- ・ サービス系の主記憶に、DB実体がロードされている。
- ・ スタンバイ系の主記憶には、DBの実体はない。(スキーマのみ)
- ・ ディスク装置には、逐次バックアップ、及びジャーナルが蓄えられる。

### 2.2 データベース保守

DBオンライン保守として想定される項目を表1に示す。

表1. オンライン保守項目

項目	内容
1 再構成	DB定義の変更 ・ テーブル追加、カラム属性変更等 ・ 変更内容によってはアプリケーションの入替えが必要。
2 再編成	データ・ファイルのガベージコレクション ・ 領域利用効率向上/アクセス高速化が目的
3 バックアップ	トラブル対応等のため、DBの複製を取得する。

本報告では、再構成及び再編成を前提として方式を検討する。

## 3. DBオンライン保守方式

スタンバイ系を用いたDBオンライン保守として、以下の2方式を提案する。

【方式1】(図2. 参照)

- (1) スタンバイ系に、保守を行うDBを複写
- (2) スタンバイ系にて保守を実行
  - ・ 保守中にサービス系DBは更新される。
- (3) 保守中に出力されたジャーナルをスタンバイ系にて反映
- (4) サービス側トランザクションを予閉塞
- (5) 予閉塞前に走行していたトランザクションのジャーナルをスタンバイ系にて反映
  - ・ これで、サービス側とスタンバイ側のDB内容が一致
- (6) 系切替え

【方式2】(図3. 参照)

- (1) スタンバイ系に、保守を行うテーブルのみを複写
- (2) スタンバイ系にて保守を実行
  - ・ 保守中にサービス系DBは更新される。
- (3) 保守中に出力されたジャーナルをスタンバイ系に

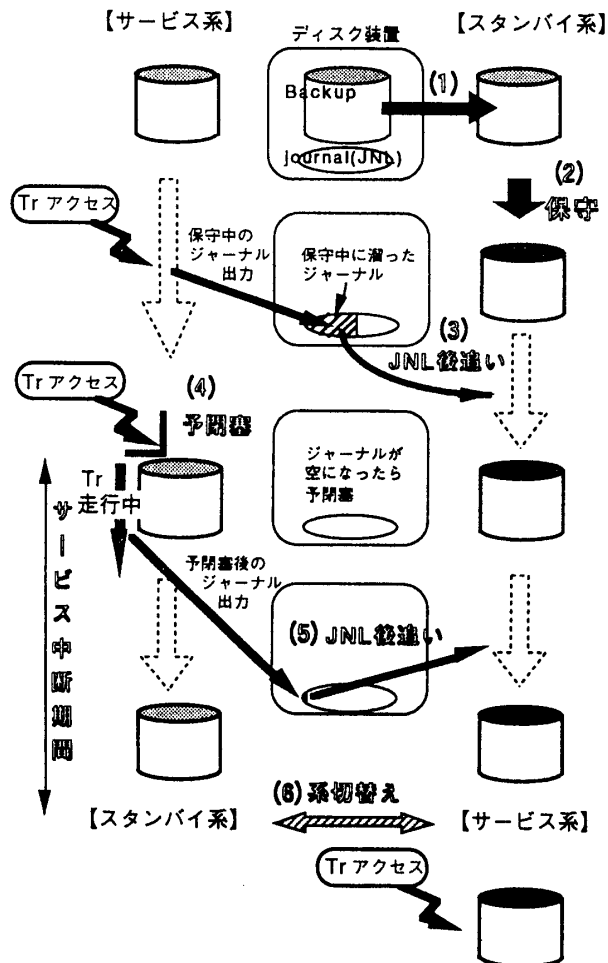


図2. オンライン保守の流れ (方式1)

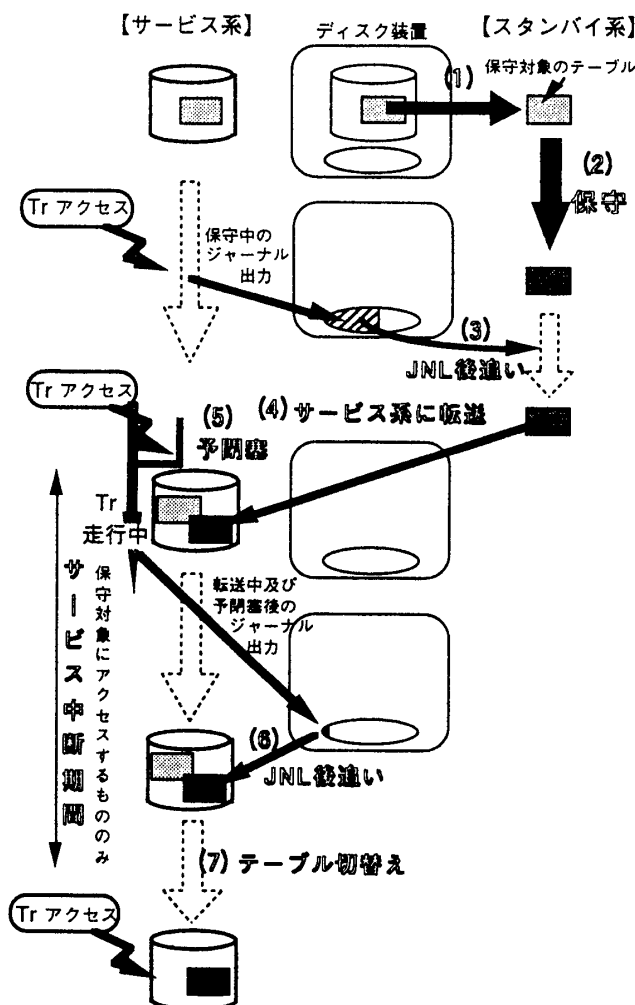


図3. オンライン保守の流れ (方式2)

- 保守対象のテーブルに該当するもののみ
- (4) 保守済のテーブルを、サービス系に転送
- (5) サービス側トランザクションを予閉塞
  - 保守対象のテーブルにアクセスするもののみ
- (6) 転送中及び予閉塞前に走行していたトランザクションのジャーナルをサービス系にて反映
  - これで保守前と保守後のテーブル内容が一致
- (7) 保守前後のテーブルを切替え、予閉塞を解除

方式1は、スタンバイ系⇒サービス系の系切替えのルートに従っているので実現及び運用が容易であり、システムの更改等の大規模な保守にも対応できる。

但し、スタンバイ系に全DBに対するジャーナルを反映させるので、保守時間が長くなる。また、予閉塞後に出るジャーナルも多いためその反映に時間がかかり、サービス中断時間が長くなる欠点がある。

方式2は、方式1の欠点を改善するため、スタンバイ系に複製するのは保守を行うデータのみ(例えばテーブル単位)とした方式である。

この時、後追いのためスタンバイ系では、ジャーナルから、保守対象のジャーナル・データのみを捜して反映すればよく、保守時間及びサービス中断時間が短縮される。このサービス中断は、サービス系でのテーブル切替え時のわずかの中断(わずかのジャーナル反映)ですみ、利用者からはトランザクションの待ち時間程度に見える。

一方、トランザクションの制御が複雑となり、運用上も系間のデータ転送があるなど比較的複雑である。

以上より、サービス無中断の要求が厳しい場合や保守範囲が限られる場合は方式2、システム更改や複数のテーブルの保守等、大規模な保守を伴うならば方式1が有用であると言える。

#### 4. おわりに

スタンバイ系で保守を行うことにより、サービス系のスループット低下を招くことなくサービス無中断のDBオンライン保守を行う方式を示した。

尚、スループットの低下が許容される、あるいは、CPU能力に余裕があるシステムであれば、本方式におけるスタンバイ系の処理をサービス系で行う方式の実現も可能である。

今後の課題としては、

- トランザクションの予閉塞制御(保守対象にアクセスするトランザクションのみを予閉塞)
- 高速なジャーナル反映方式
- アプリケーションの入替えを伴う場合の保守方式等があり、今後検討を進める予定である。