Oracleデータベースの高速引継ぎを実現するクラスタシステム

Cluster System for High-Speed Database Takeover

飯沼　哲也　　田中　　雅

■IINUMA Tetsuya　　■TANAKA Satoshi

一般論文

クラスタシステムにおいて，Oracle(注1)データベースシステムの引継ぎ時間を短縮するために，プロセスチェックポイント＆レジューム技術を用いたDNCWARE™ ClusterPerfect™ for Oracle Quick Recovery（以下，Oracle Quick Recoveryと略記）を開発した。Oracle Quick Recoveryは，従来数分から数十分かかっていたOracleデータベースシステムの引継ぎ時間を，わずか数十秒に短縮する。これにより，クラスタ構成をとったOracleデータベースシステムで障害が発生しても高速なデータベース引継ぎを行い，より可用性の高いデータベースシステムの構築が可能になる。

In order to cut down the takeover time in Oracle database systems, we have developed DNCWARE™ ClusterPerfect™ for Oracle Quick Recovery. The conventional takeover time of a few or tens of minutes shrinks to tens of seconds by the mechanism of taking checkpoints and resuming the processes. The availability of an Oracle database system configured on a cluster computer system is enhanced by such quick takeover in the event of failure.

1　まえがき

今日，UNIX(注2)サーバやパソコン（PC）サーバに代表されるオープンシステムは，基幹システムの中心的役割を果たすようになった。これに伴い，メインフレームと同等の可用性を得るため，複数のサーバで構成されるクラスタシステムが広く利用されている。

当社では，サーバシステムの可用性を高めるミドルウェアとして，統合クラスタソフトウェアDNCWARE™ ClusterPerfect™（以下，ClusterPerfect™と略記）を開発し販売している。ClusterPerfect™は，システム構築の容易性，システム構成の柔軟性，マルチベンダ／マルチプラットフォーム対応などが市場で高く評価され，国内トップクラスの実績を誇っている。今回述べるOracle Quick Recoveryは，ClusterPerfect™のファミリー製品として開発し販売している製品である。

一般にクラスタシステムは，システムの可用性を大幅に高めることができ，サーバのダウンタイムを短縮できる。しかし，稼働中にダウンしたデータベースシステムは，ダウン前の整合性の取れた状態に復元させる必要がある。これは，ジャーナルリカバリ処理と呼ばれるデータベースの回復処理によって行われる。すなわち，稼働系サーバがダウンし，待機系サーバでデータベースシステムを引き継いだ場合，まずジャーナルリカバリ処理を実行する必要がある。

このジャーナルリカバリ処理に要する時間は，実行中であったトランザクションの状況にも依存するが，通常数分から数十分を要する。このため，サーバ自身は短時間で切り換えることができるが，データベースのジャーナルリカバリ処理のため，数分から数十分もの間，サービスが再開できない場合がある。

そのため，数分のデータベースサービスの停止も許されない重要なシステムにおいては，サーバ自身がダウンしないフォールトトレラント計算機がしばしば利用されてきた。しかし，ハードウェアの選択肢の幅が広く，ソフトウェア障害への対応に優れた汎用のクラスタシステムを利用したいという要求は多い。

このような背景から，障害発生時のOracleデータベースシステムの引継ぎ時間を大幅に短縮するクラスタソフトウェア，Oracle Quick Recoveryを開発した。

2　高速引継ぎの動作概要

データベースシステムのOracleでは，処理性能を高めるため，ディスク上のデータベースファイルを直接更新せず，主記憶上に置いたデータベースバッファを更新し，それとともに，実行したトランザクションの履歴をジャーナルファイルに書き込む方法を採る。万一，データベースシステムがサーバの障害によりダウンしてしまった場合，ジャーナルファイルを基に，ディスク上のデータベースファイルを復元する。これが，ジャーナルリカバリ処理に数分から数十分もの時間を要する理

（注1）　Oracle及びその他のOracleを含む商標は，Oracle Corporationの商標又は登録商標。

（注2）　UNIXは，商標。

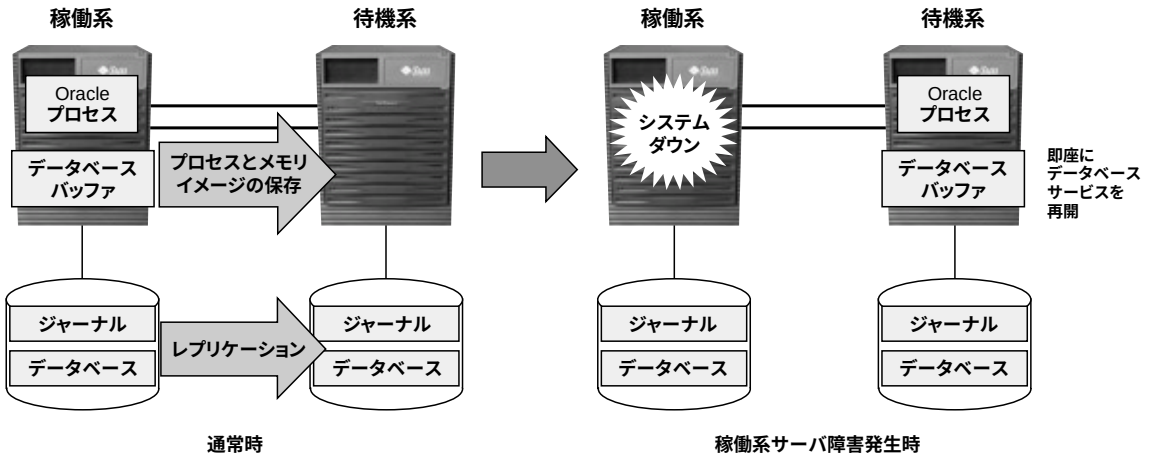

図1．Oracle Quick Recoveryによる Oracle データベースの引継ぎ ― 稼働系サーバで採取したチェックポイント情報により稼働系サーバにOracleプロセスを引き継ぐ。

Database takeover by Oracle Quick Recovery

由である。

Oracle Quick Recoveryでは，当社独自のプロセスチェックポイント＆レジューム技術[(1)−(3)]を用いることで，ジャーナルリカバリ処理を不要にしている。**図1**に示すように，稼働系サーバで動いているOracleのプロセスチェックポイント情報を採取し，待機系サーバに送る。そして稼働系サーバがダウンした場合には，待機系サーバで，Oracleをプロセスチェックポイント情報からレジュームする。Oracle自身は，その実行プロセスが稼働系サーバから待機系サーバにマイグレート(移行)されたかのように継続動作する。そのため，引継ぎ時にもジャーナルリカバリ処理が発生せず，引継ぎに要する時間をわずか数十秒に短縮し，オープンプラットフォーム環境で，より可用性の高いデータベースシステムの構築を可能にしている。

3 高速引継ぎを実現する主な機能

Oracle Quick Recoveryの特長的な機能として，次の3点を挙げる。

⑴ プロセスチェックポイント＆レジューム機能
⑵ ファイルレプリケーション機能
⑶ トランザクションのアトミック性維持機能

それぞれの機能について，以下に述べる。

3.1 プロセスチェックポイント＆レジューム機能

Oracle Quick Recoveryの高速引継ぎの基本は，汎用のプロセスチェックポイント＆レジューム機能にある。Oracle Quick Recoveryは，Oracleのシステムプロセスだけをプロセスチェックポイント＆レジューム機能の制御下で実行させ，1秒程度の間隔でチェックポイント情報を採取する。サーバに障害が発生した場合には，Oracleのすべてのシステムプロセスを，最後に採取したチェックポイント情報からレジュームする。このチェックポイント情報には，以下のデータが含まれる。

⑴ プロセスのアドレス空間とコンテキスト
⑵ プロセスが基本ソフトウェア(OS)から受けているサービス

プロセスのアドレス空間は，読込み／書込み可能なページに対しても，いったん書込み保護しておき，書込み時に発生するページフォールト[(注3)]をハンドリングすることにより更新ページを記録し，チェックポイント情報採取時に，更新ページのみを保存する。

プロセスがOSから受けているサービスに関しては，プロセスが発行するシステムコールをフックし，その情報を保存する(**図2**)。レジューム時には，保存していた情報を基にシステムコールを再発行することで，プロセスがOSから受けていたサービスの状態を復元する。例えばファイル操作では，チェックポイント情報採取時にオープンしていたファイルを，同じファイル記述子で再オープンする。通常のオープン操作では，ファイル記述子はオープンごとに異なる値が返るが，システムコールのフック内で，チェックポイン情報採取時のファイル記述子と同じ記述を返すための操作を行っている。同様にファイルポインタも，チェックポイント情報採取時の位置に合わせる。

セマフォ[(注4)]や共有メモリは，レジューム時に最初と同じID(IDentification)で再獲得することができない。そのため，

(注3) 物理メモリへのアクセス時に，存在しないメモリ空間，書込み禁止のメモリ空間にアクセスしたときに起こる割込み。
(注4) 複数のプログラム(プロセス)の間で同期をとるための仕組み。一種の共有フラグ。

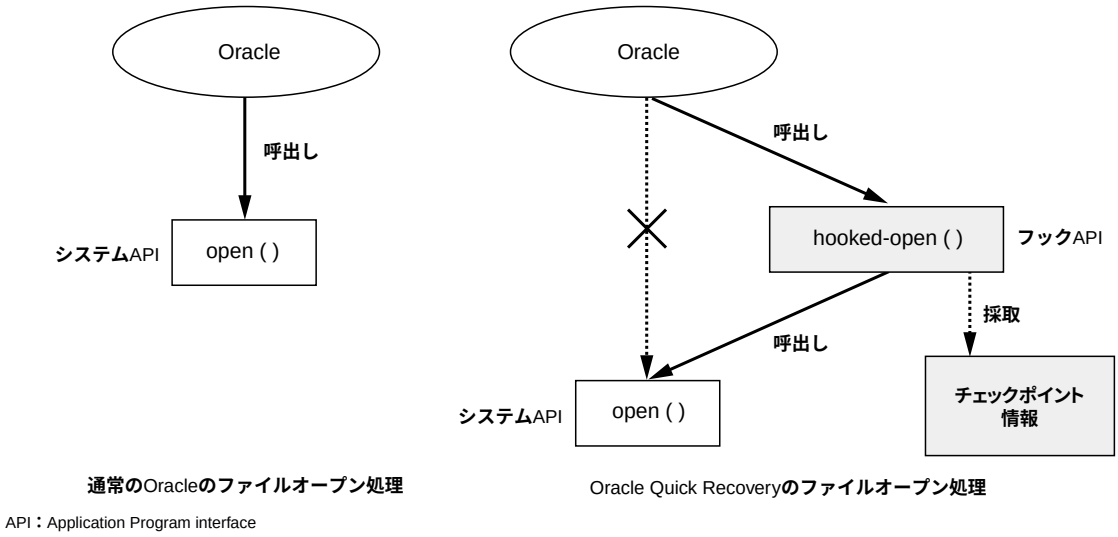

図2．Oracle が OS から受けるサービスのチェックポイント情報取得― システムコールのフックによりチェックポイント情報を採取する。
Acquisition of system service checkpoints by system call hooking

プログラムが扱っているセマフォや共有メモリのIDを仮想IDとみなし，それと一対一に対応する物理IDを定義する。最初は仮想IDと物理IDを一致させておき,レジューム後は，再獲得したIDを物理IDに設定する。プログラムは，レジューム後も最初のID（仮想ID）でセマフォや共有メモリにアクセスするが，プロセスが呼び出すシステムコールをフックしていて，仮想IDを物理IDに変換する。これにより，セマフォや共有メモリではレジューム時の再獲得でIDが変化してしまうが，プログラムは最初のIDをそのまま使用することができる。

3.2　ファイルレプリケーション機能

ファイルレプリケーション機能は，運用系サーバで実行されるOracleのシステムプロセスのファイル更新操作をフックし，そのファイル更新情報を収集する。例えば，書込み操作であれば，ファイル更新情報には以下の内容が含まれる。

⑴　書込みファイルの記述子
⑵　書込み位置（ファイルポインタ）
⑶　書込みデータの長さ
⑷　書込みデータ

このファイル更新情報は，運用系サーバでバッファリングし，バッファが満たされると待機系サーバに送る。待機系サーバに送られたファイル更新情報は，ただちに待機系サーバのファイルに反映するのではなく，いったん“未確定キュー”と呼ばれるキュー（待ち行列）にリンクする。チェックポイント情報採取時には，運用系サーバのバッファに残っているファイル更新情報を，待機系サーバの未確定キューにリンクする。

そして，待機系サーバでは，未確定キューにリンクされているファイル更新情報を，“確定キュー”と呼ばれるキューに移動する。確定キューに移動されたファイル更新情報は，チェックポイント情報採取後，待機系サーバのファイルに順次反映する。

レプリケーション機能の制御下では，運用系サーバで行われたファイルの更新は，待機系サーバではチェックポイント情報採取後にファイルに順次反映する。そのため，障害発生時にOracleのシステムプロセスをチェックポイント情報からレジュームする場合，待機系では直前のチェックポイント情報採取以後，未確定キューにリンクされたファイル更新情報を破棄することで，ファイルの状態をチェックポイント情報採取時の状態にすることができる。

3.3　トランザクションのアトミック性維持機能

データベースにとって，トランザクションのアトミック性(注5)を保証することは絶対条件である。しかし，一般的なプロセスチェックポイント＆レジューム方式では，稼働系でチェックポイント情報採取を行ってから障害が発生するまでのトランザクションに対して，トランザクションのアトミック性を保証することができなかった。

Oracle Quick Recoveryは，コミット(注6)要求が発生した際に，コミットの完了通知を次のチェックポイント情報採取完了後まで遅延させることで，データベースのトランザクション性を維持している[4]（**図3**）。

（注5）　一連の連続した処理の実行順序が保証されること。
（注6）　データベースのデータ変更結果を確定すること。一般の場合，メモリ上の変更データをディスクに書き出すことを行う。

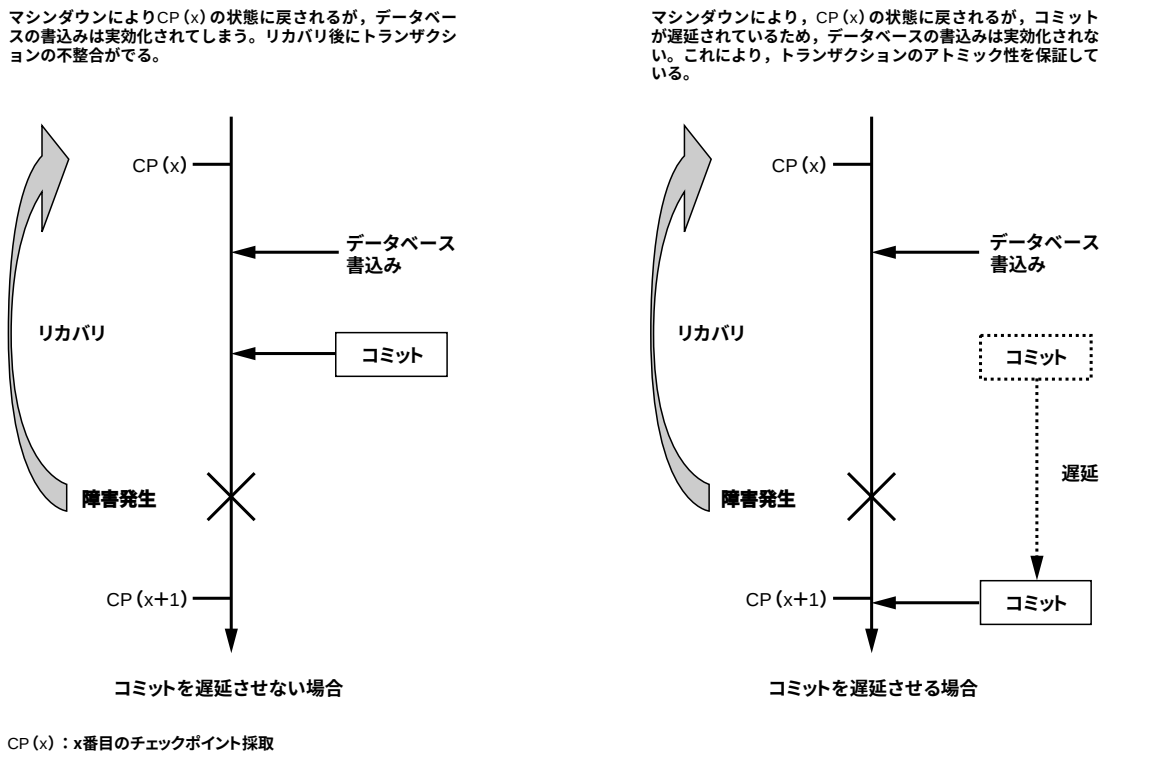

図3．トランザクションのアトミック性の維持 ― データベースのコミット処理を次のチェックポイント取得終了まで遅延させることにより，トランザクションのアトミック性を維持する。

Consistency control for atomicity of transactions

4 あとがき

Oracle Quick Recoveryは，汎用的なプロセスチェックポイント＆レジューム技術に当社独自のトランザクションのアトミック性維持機能を組み込み，Oracleデータベースシステムに適用することで，万一データベースシステムに障害が発生しても，サーバのみならずOracleデータベースも高速に復旧させ，数十秒のうちにOracleサービスを再開できる。これにより，オープンプラットフォーム環境で，より可用性の高いデータベースシステムの構築が可能であり，他社のクラスタソフトウェアとは一線を画すものとなっている。

文　献

(1) Shirakihara, T.,et al. ARTEMIS: Advanced reliable distributed environment middleware system. Proceedings of the International Conference of Parallel and Distributed Processing Techniques and Applications. **1**, 6, 1997, p.97－106.

(2) 白木原敏雄，ほか．高可用性の新技術（無停止分散システム構築ミドルウェア）．東芝レビュー．**52**，8，1997，p.40－42.

(3) 平山秀昭，ほか．分散チェックポイント方式との組合せによりフォールトトレラントシステムを実現する分散レプリケーション方式．電子情報通信学会論文誌 D-I，**J82-D-I**，3，1999，p.496－507.

(4) 村田明文，ほか．プロセスマイグレーション技術のリレーショナルデータベースへの応用．情報処理学会第60回全国大会．**1**，3，2000，p.27－28.

飯沼　哲也　IINUMA Tetsuya

e-ソリューション社 府中e-ソリューション工場 ミドルウェア部。ミドルウェアの開発に従事。

Fuchu Operations－e-Solutions

田中　　雅　TANAKA Satoshi

e-ソリューション社 府中e-ソリューション工場 ミドルウェア部。ミドルウェアの開発に従事。

Fuchu Operations－e-Solutions