

**エンタープライズ・レプリケーション：
情報を配信および共有するための
ハイ・パフォーマンス・ソリューション**

目次

4	はじめに
5	エンタープライズ・レプリケーション: 概要
7	データ・レプリケーションについて
8	データ・レプリケーション: 目的に適合するタイプの選択
8	マスター/スレーブ所有権
9	データ・ディセミネーションによる集中的データ制御
10	データ統合で可能になるソースでのデータ更新
11	ワークロード・パーティショニングで可能になるデータのローカル制御とグローバル表示の両立
12	ワークフロー所有権による所有権の移動
13	随時更新レプリケーションで可能になるローカル更新とエンタープライズ全体の整合性の両立
14	変更のキャプチャー: トリガー・ベース対ログ・ベース
17	IDSエンタープライズ・レプリケーション: 多くの利点があるアーキテクチャー
17	ハイ・パフォーマンス
18	ハイ・アベイラビリティ
19	一貫した信頼できるトランザクションの整合性
19	フレキシビリティ
20	マネージビリティ
21	IDSエンタープライズ・レプリケーション: 技術的な概要
22	構成および制御
23	レプリケーション・ネットワーク・トポロジー
26	トランザクション・キャプチャー
27	トランザクションのキューイングおよび配信
28	ターゲット・サーバーでのトランザクション挿入
28	コンフリクトの検出および解決
29	既存のエンタープライズ・レプリケーション・ユーザー向け新機能
30	IDSエンタープライズ・レプリケーションを使用する理由



はじめに

この10年で、コーポレート・コンピューティングはメインフレームの役割を変えました。もはやメインフレームは企業の唯一の情報リソースではありません。こうした変化は、企業にITリソースの最大化を要請する、厳しい競争、コスト削減、および組織化によって引き続き進められています。このような脱中心化のトレンドの結果、ISマネージャーたちは、エンタープライズの隅々にまで企業の情報を届ける分散データベースの管理をするようになりました。

こうした情報を効果的に配信および共有するためには、データのレプリケーションが重要です。しかしレプリケーションは、データをサイトからサイトへ移動するだけの処理ではありません。レプリケーションは、きわめて高度なプロセスです。それゆえ、レプリケーション・システムの成功は、ビジネスおよびアプリケーション全般の必要を満たすテクノロジーにかかっています。

レプリケーション・システムは、ソース・データベースに負荷をかけず、かつアプリケーションの変更なしでハイ・パフォーマンスを提供しなければなりません。システムのアベイラビリティを極力高めて、データの安定した配信を確保する必要があります。そして、最終的に、データベース管理者はレプリケーション・システムのすべてのコンポーネントを構成する際に、エンタープライズ・コンピューティング・リソースを最も効率的に利用できるようにする必要があります。このペーパーでは、以上の要件を説明しつつ、IBM Informix® Dynamic Server (以下、IDS) がどのようにしてエンタープライズにこうした重要な機能をもたらすかを示します。最初にIDSのテクノロジーを概観してから、レプリケーションの重要な概念および事項を説明します。その次に、IDSレプリケーション・ソリューション — エンタープライズ・レプリケーション (以下、ER) — を詳しく説明します。

エンタープライズ・レプリケーション:概要

ハイ・パフォーマンス・レプリケーション・ソリューションであるIDS ERは、企業の情報を配信および共有するためのエンタープライズ全体にわたる基盤を提供します。エンタープライズ・レプリケーションは、ワークグループからハイエンドSMPシステムまでのレプリケーション要件をサポートします。多くの重要な要素がこのソリューションの際だった利点を特長づけ、多様なビジネスおよびアプリケーションの要件に適したものにしています。

●ハイ・パフォーマンス

先進的なIDSテクノロジーは、大規模なエンタープライズ・アプリケーション向けの優れたパフォーマンス、スケーラビリティ、およびマネジビリティを提供します。IDSは、スケーラブルな並行処理サーバー・アーキテクチャーを十分に活用する初のRDBMSレプリケーション・ソリューションを提供し、要求水準の高いコンピューティング環境のほとんどに適します。

●ハイ・アベイラビリティ

IDSレプリケーション製品は、多様なオプションを通じて高いアベイラビリティを保証します。ホット・スタンバイ・サーバーから、単一または複数サイトへのデータの非同期レプリケーションまでをそろえています。非同期レプリケーションでは、ネットワークおよびターゲット・ノードの機能停止が許容され、ノード間での障害の影響が最小に抑えられます。レプリケートが必要な更新は、IDS ERグローバル・カタログへの変更も含めて、レプリケーション処理が再開された時点でリモート・サイトへ自動的に伝達されます。

● **トランザクション・レベルの整合性**

一貫性を維持するために、データベースへの変更はソース・データベースからターゲット・データベースへ即座に伝達されます。トランザクションがレプリケートされる際には、トランザクションの順序も保持されます。また、IDS ERでは、更新のコンフリクトを検出および解決するための多数のオプションが用意されています。

● **フレキシブルなアーキテクチャー**

IDS ERが提供するレプリケーション・アーキテクチャーによって、組織に固有のビジネスおよびアプリケーション要件に基づいたレプリケーション環境を定義することができます。IDS ERでは、マスター/スレーブ、ワークフロー更新、および随時更新など多様なレプリケーション使用モデルがサポートされています。ネットワーク・トポロジーは、全接続および階層的接続の両方がサポートされています。IDS ERの実装はアプリケーションに対して透過的であり、このためコードの変更は不要です。

● **マネージビリティ**

構成および監視は、データベース・サーバー管理フレームワークと統合されている、ブラウザ・ベースの一連のツールを使用して実行可能です。また、スクリプトの使用を容易にするために、コマンド・ライン・インターフェースも用意されています。

IDS ERの詳細を説明する前に、レプリケーション・テクノロジーの導入を検討されている企業のために、レプリケーションの主要な概念および考慮事項を概説します。



データ・レプリケーションについて

レプリケーションを簡潔に定義すると、1つ以上のサイトでデータの複数のコピーを生成および複製するプロセスのことです。データベース・レプリケーションは、エンタープライズがユーザーに対して、必要なときに必要な場所で最新データへのアクセスを可能にするという理由で重要です。データ・レプリケーションには多くの利点があり、例えば、集中化されたリソースのオーバーロード時におけるパフォーマンスの改善、データ・アベイラビリティの向上、処理能力の余裕、および、意思決定支援を容易にするデータ・ウェアハウジングのサポートなどが含まれます。

レプリケーションには、同期または非同期があります。どちらのタイプも、特定の目的に適した機能および利点を備えています。同期レプリケーションでは、ソース・データが更新された時点で必ず、レプリケート・データも即座に更新されます。同期レプリケーションの間、データの整合性は2フェーズ・コミット・プロトコルによって通常保護されます。各サーバーは、データを更新するとソース・データベースに確認メッセージを送信し、応答のコマンドを待ってトランザクションを完了します。同期レプリケーションは、即座にデータを同期しておく必要があるアプリケーション（例えば金融トランザクション）に適した技術ですが、2フェーズ・コミットのオーバーヘッドが大きく、パフォーマンスを大幅に低下させる可能性があります。さらに、同期レプリケーションの実装に使用されるトランザクション・モニターは一般にシステムの障害個所となりうるため、アベイラビリティに影響します。

一方、非同期レプリケーションでは、ターゲット・データベースの更新はソース・データベースが変更された後にのみ実行されます。この処理による更新の遅れは、構成に応じて2、3秒から数時間になりますが、これは最終的にすべてのサイトでデータが同じ値に同期されるためです。特定のサイトに障害が発生したり、アクセスができなくなった場合も、非同期レプリケーションではローカル処理の要求を継続することができます。例えば、注文入力システムでは、発送サイトがダウンしていても発注の入力を継続することが可能です。障害を起こしていたサイトがオンラインに復帰した時点で、システムはこれらの入力の伝達を保証します。

データ・レプリケーション:目的に適合するタイプの選択

分散コンピューティングに対応する上でレプリケーションが大きな役割を負うゆえに、組織はレプリケーションの使用法を理解して、このテクノロジーをどのように利用するのが最善かを検討する必要があります。基本的な問題は、データの整合性とアベイラビリティとの折り合いです。すでに説明したとおり、同期レプリケーションでは整合性を保証しつつ、アベイラビリティと応答時間のパフォーマンスが犠牲になります。他方のレプリケーション方式である非同期レプリケーションでは、アベイラビリティと応答時間のパフォーマンスは高くなりますが、計画および設計の時点でデータの整合性を保証し、更新のコンフリクトを解決する必要があります。最終的に、最適な運用戦略は、ビジネスおよびアプリケーションの要件によって決定されます。レプリケーション・システムは、これらの要件をサポートする必要があります。非同期レプリケーションでのもう一つの基本的な問題は、データの所有権です。所有権とは、サイトがデータを更新する権限のことをいいます。所有権のタイプによって、コンフリクトの検出および解決の必要性が決まります(それらを実行する場合)。例えば、マスター/スレーブ構成では、1つのサイトのみがデータを更新でき、通常それは特に指定のない限りマスターのみなので、コンフリクトの検出および解決の必要性が解消されます。その他の所有権モデルは、後で説明するとおり、複数のサイトでの情報更新を許容するためにコンフリクトの検出および解決を利用することが必須となります。

マスター/スレーブ所有権

マスター/スレーブ所有権では、複数の利用シナリオが可能で、データ・ディセミネーション、データ統合、およびワークロード・パーティショニングなどがあります。

データ・ディセミネーションによる集中的データ制御

データ・ディセミネーションでは、中央のロケーションでデータを更新し、それを各地の読み取り専用サイトにレプリケートします。この方式は、情報をブランチ・ロケーションへ配信する必要がある場合に便利です。例えば、ある書店チェーンの本部が、在庫書籍の価格表の更新版を夜間に各支店へ送信する必要があるとします。このデータの一貫性を確保するために、本部は情報に対して読み取りおよび書き込みができる一方で、各支店は読み取りのみのアクセスができるようにします。

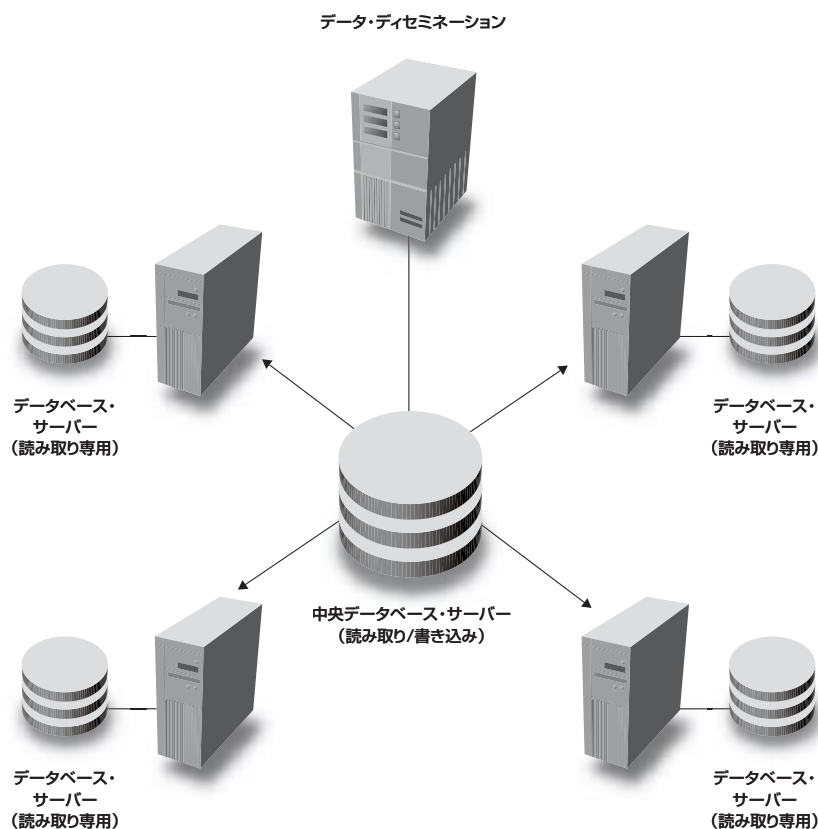


図1:データ・ディセミネーションでは、データは中央のロケーションで更新され、複数の、読み取り専用のサイトへレプリケートされます。

データ統合で可能になるソースでのデータ更新

データ統合では、地域でデータ・セットを更新して、それを中央データベース・サーバーにある読み取り専用のリポジトリにまとめます。この方式では、データ所有権およびロケーション自律性をブランチのレベルに付与します。こうしたデータ統合環境の例として、店頭売上情報を一日中収集する小売店チェーンがあります。営業日の終わりに、各店舗はデータを本部へ送信し、本部ではデータが中央データ・ウェアハウスに統合されて、トレンド分析など各種の業務情報処理に利用されます。

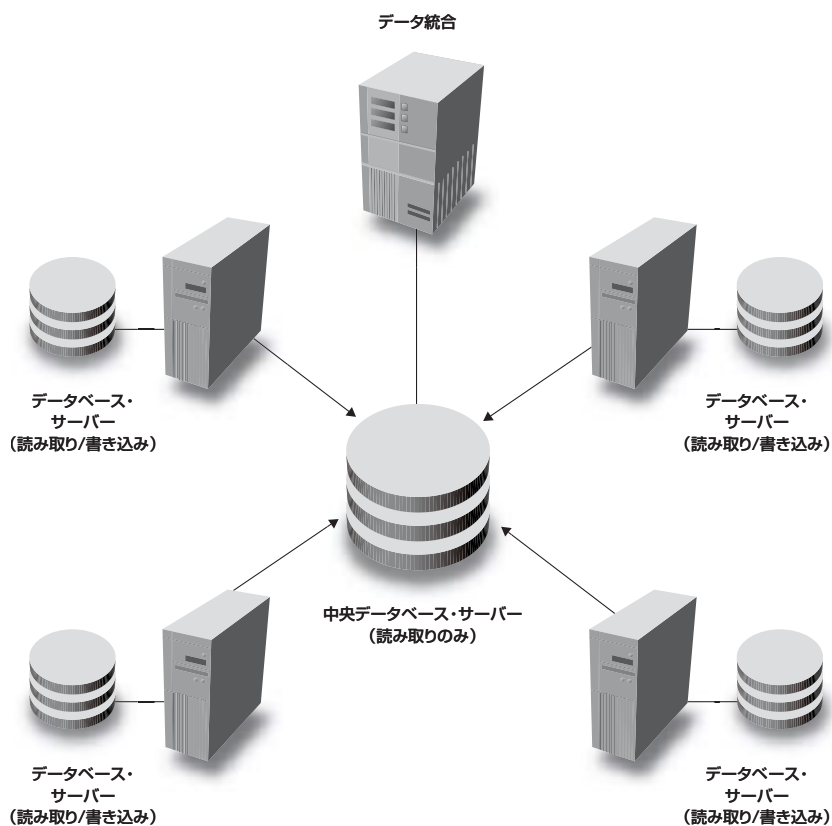


図2: データ統合では、データは複数のサイトで更新され、中央の、読み取り専用のサイトにレプリケートされます。

ワークロード・パーティショニングで可能になる データのローカル制御とグローバル表示の両立

ワークロード・パーティショニングでは、データベース管理者 (DBA) が柔軟にデータ所有権をテーブル・パーティション・レベルで割り当てることができます。次の図に例を示します。このレプリケーション・スキーマは、従業員テーブルのパーティショニング・スキーマにマップされています。アジア/太平洋サイトは自身のパーティションの所有権を持っていて、地域内の人員に関する従業員レコードを更新、挿入および削除することができます。データへのあらゆる変更は、米国および欧州のサイトへ伝達されます。アジア/太平洋サイトは、ほかのパーティションに対してクエリーを発したり、読み取ったりできますが、更新することはできません。同様に、米国および欧州のサイトは、自身のパーティション内のデータのみを変更でき、すべてのパーティションに対してクエリーを発したりデータを読み取ったりすることができます。

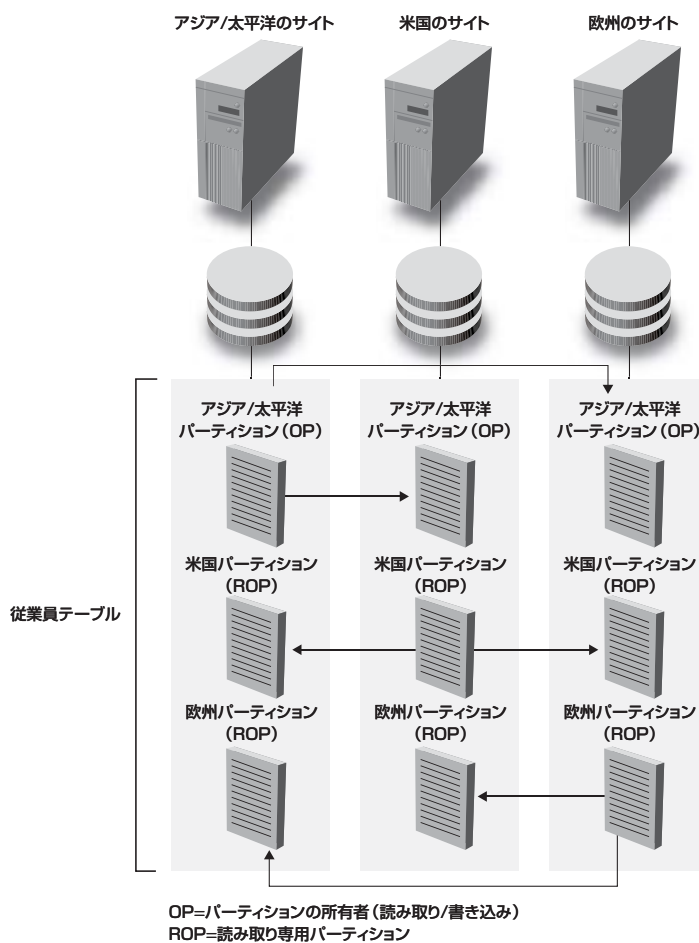


図3:ワークロード・パーティショニングでは、各サイトのユーザーは自身のパーティション内のデータのみを更新できる一方で、すべてのテーブルのデータを表示することができます。

ワークフロー所有権による所有権の移動

マスター/スレーブ所有権と同様に、ワークフロー所有権でもIDS更新のコンフリクトは回避されますが、マスター/スレーブの場合よりも動的な更新テクニックが用意されています。ワークフロー所有権では、データがサイトからサイトへ移動することができます。ただし、ある時点でデータを更新できるサイトは1つのみです。各サイトは1つ前のサイトからのデータに依存するため、それぞれが更新できるデータは自身の業務機能に関するもののみです。各ステップが完了した時点で、データは更新されて次のサイトへレプリケートされます。

一般的なワークフローの例は、注文処理システムです。まず、ある注文が受注入力部門で入力されます。次に、この注文はクレジット承認および請求のため会計部門へ送られ、それから発送許可のために在庫管理部門へ送られます。最後に、注文は発送部門へ回送されて、包装および配送が行われます。

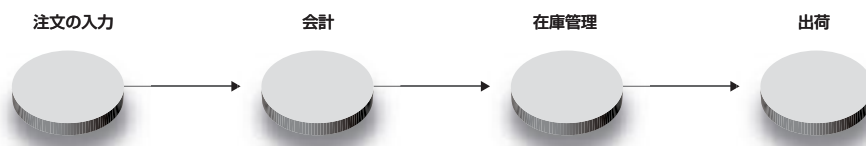


図4: 一般的なワークフローの例では、データを変更する権限がサイトからサイトへ進みます。

随時更新レプリケーションで可能になる ローカル更新とエンタープライズ全体の整合性の両立

組織では、複数のサイトでのリアルタイム更新をサポートする機能も求められます。レプリケートされたデータが読み取り専用となるマスター/スレーブ所有権とは異なり、随時更新レプリケーションは、複数のサイトがデータを更新する同等の権限を持つピアツーピア環境で発生します。これによりローカル・ユーザーは、分散環境でほかのシステムまたはネットワークを利用できないときでも、自律的に機能することができます。ただし、更新のコンフリクトを解決するためには、レプリケーションのアーキテクチャーが多様なコンフリクト検出および解決ルーチンによってこの環境をサポートする必要があります。次の図に随時更新の例を示します。この例では、通信販売業者が注文用のフリーダイヤルを用意しています。電話は、時刻および電話負荷バランス要件に応じて3つのコール・センターの1つへ転送されます。このため、各コール・センターには、レコードを更新する機能と、発注があった時点でほかのコール・センターにも即座にレプリケートする機能が必要になります。



図5:随時更新のシナリオでは、データが一部のサイトによって所有および更新される一方、他のサイトは読み取り専用のコピーを利用します。

変更のキャプチャー:トリガー・ベース対ログ・ベース

最も頻繁に使用されるレプリケーション変更キャプチャーのメカニズムは、トリガー・ベースおよびログ・ベースのキャプチャーです。トリガー・ベースのキャプチャーでは、データへの変更がデータベース内のトリガーを起動します。これによりソース・データベース内でレプリケーション固有のコードが作動し、今度は、レプリケーションのプロセスが開始されます。このプロセスは、ユーザーのトランザクションと同じ作業ユニット内で同時に発生するため、ソース・データベースでのトランザクション処理のパフォーマンスに直接影響します。

トリガー・ベースのシステムの別の重要な問題は、汎用のデータベース・トリガーからトリガー・ベースのシステムが生まれた経緯から派生します。当初データベース・トリガーはデータベースのベンダーによって導入され、その目的は企業データの参照整合性を保護することと、ビジネス・ルールを中央集権的に強制することでした。こうしたベンダーたちにとって、トリガー・ベースのレプリケーションは、自社の製品に対する合理的かつ明快な拡張でした。また、トリガー・ベースのレプリケーションによって、サード・パーティーのベンダーには異種間でのレプリケーションを実装する柔軟性が与えられました。これは、ほとんどのDBMSベンダーが内部のログ・フォーマットを非公開としているからです。しかし、その結果、トリガー・ベースのレプリケーションには不利な点が生まれました。なぜなら、各ベンダーはトランザクションの整合性を保証するために追加機能を導入する必要があり、こうした追加機能がパフォーマンスおよび管理の面で重荷になったからです。

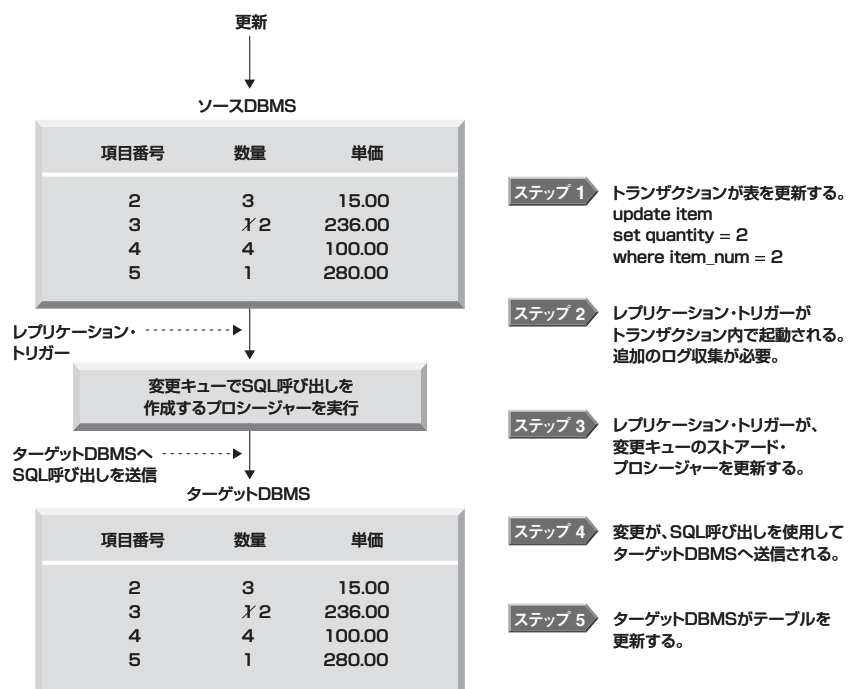


図6: トリガー・ベースのレプリケーションは、本来のトランザクションのパフォーマンスに悪影響を与えます。

他方、ログ・ベースのシステムは、通常のログ収集プロセスの一部として機能することができ、システムに追加するオーバーヘッドが最小で済みます。トランザクション・ログは、ソース・データへの変更をキャプチャーする最善のソースとみなされています。ソース・データベースへの変更の検出および伝達は、ソース・データベース・システムの通常の機能を中断せずに実行されます。

ただし、ログ・ベースのレプリケーション・システムがすべて同じというわけではありません。重要な相違点の1つは、データベース・システムとログ・キャプチャー・メカニズムの統合の度合いです。例えば、外部のサーバーおよびプロセスを利用するログ・ベースの製品では、データ転送のオーバーヘッドが加わるために、ログ・ベースのトランザクション・キャプチャーの利点がすべて相殺されてしまいます。ログ・ベースのシステムの成功を左右するもう一つの特徴は、ログのすべてのエントリーをトランザクション・キャプチャーのプロセスの一部として読み取る必要があるかどうかです。

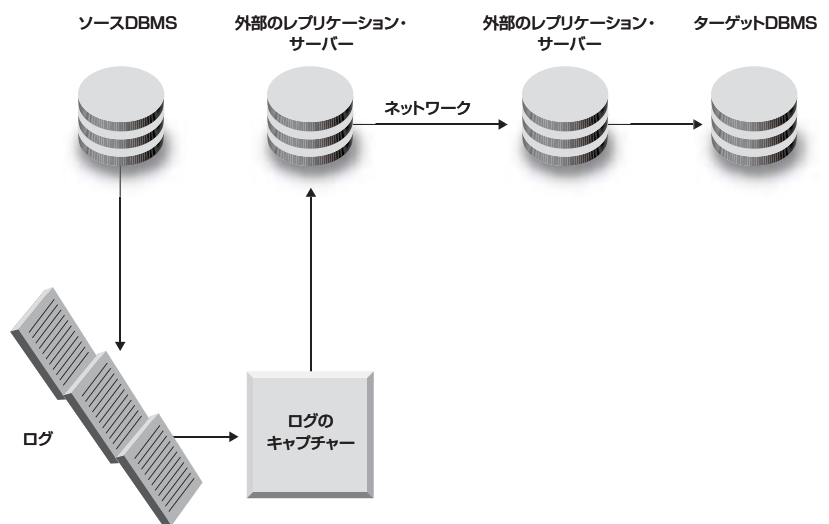


図7:ログ・ベースのキャプチャーに使用される外部レプリケーション・サーバーは、追加のデータ転送オーバーヘッドを招きます。

トリガー・ベースのアーキテクチャーおよび粗末なログ・ベースの実装に内在する弱点について、IDS ERソリューションを設計する際に事前調査が行われました。その結果、IDS ERは、データベース・システムに統合されたハイ・パフォーマンスなログ・ベースのトランザクション・キャプチャーおよび配信メカニズムを基に開発されています。

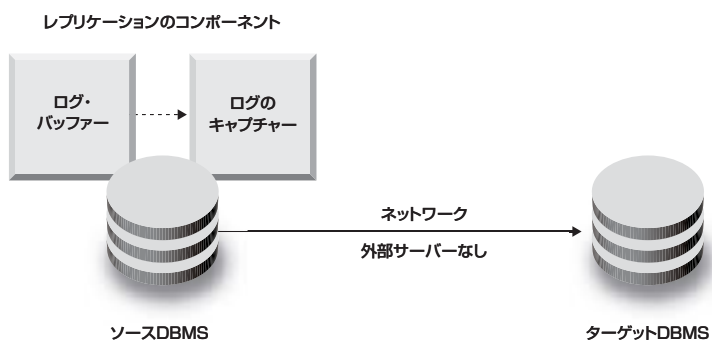


図8: DBMSと統合されたレプリケーションにより、外部サーバーが不要になり
ログ・キャプチャーのオーバーヘッドが最小になります。



IDSエンタープライズ・レプリケーション:多くの利点があるアーキテクチャー

IDS ERの導入により、非常に先進的なレプリケーション・ソリューションが市場に登場しました。業界をリードするDynamic Scalable Architecture(以下、DSA)に基づいて構築されたIDS ERは、ほかのレプリケーション製品の限界にとらわれず、クラス最高レベルのテクノロジーを提供します。IDS ERは、ワークグループ・サーバーからハイエンドSMPシステムまでの各システムをサポートし、以下のように広範な業務上の要件を解決するように設計されています。

- ハイ・パフォーマンス
- ハイ・アベイラビリティ
- 一貫した信頼性の高いトランザクションの整合性
- フレキシブルなアーキテクチャー
- マネージビリティ
- 異質なデータ・ソースのサポート

ハイ・パフォーマンス

DSAは、エンタープライズ規模のデータ・レプリケーション・システムの要求をサポートするにふさわしいハイ・パフォーマンスでスケーラブルなプラットフォームを提供します。早期のうちに、トリガー・ベースのレプリケーションの限界が認識されたことから、データベース・アーキテクチャーと統合された、効率の高いログ・ベースのトランザクション・キャプチャーおよび並行配信メカニズムを実装することになりました。このアプローチは、外部サーバーを利用するほかのログ・ベースのシステムにつきまとう、パフォーマンスの低下および管理の難しさを伴わずに実装されています。

さらに、DSAは、SMPアーキテクチャーのサーバーが使用される場合のために、そうしたシステム本来の並行処理能力を活用する「パラレル・エプリシング」アプローチを実装しています。その結果、すべてのアプリケーション処理が並行して実行され、システムのパフォーマンスをさらに高めつつボトルネックを最小にしています。

ハイ・アベイラビリティ

アベイラビリティとフォールト・トレランスを最大にするために、IDS ERは中央データベースを1台のセカンダリー・サーバーへレプリケートする構成に対応しています。このオプションは、プライマリー・サーバーに障害が発生した場合に処理を引き継ぐホット・スタンバイ・サーバーを作成するときに有用です。

また、IDS ERは1つまたは複数のセカンダリー・サイトへデータを非同期にレプリケーションすることで、プライマリー・システムの障害に対する保護も用意しています。プライマリー・サイトでの更新はすべて、IDS ERグローバル・カタログへの変更も含め、セカンダリー・サイトへ自動的に伝達され、すべてのサイトにデータの一貫したレプリカがあるようにします。更新の転送は即時または時刻指定が可能で、時刻指定の場合はDBAで更新の間隔を指定します。また、更新はイベント駆動も可能で、例えばトランザクションのコミット後、あるいはユーザーによって指定された時点などに行うこともできます。

さらに、IDS ERは、信頼性の高いメッセージ配信メカニズムも備えており、データをローカルに格納してから別個のトランザクションとしてリモート・サーバーへ伝達することができます。サーバーまたはネットワークに障害が発生しても、ローカル・サーバーはローカル・ユーザーへのサービスを継続でき、高度なフォールト・トレランスを提供します。障害が発生したサーバーまたはネットワークが稼働を再開すると、ソース・データベースへのすべての変更がリモート・データベースへ伝達されます。

一貫した信頼できるトランザクションの整合性

IDS ERは、分散環境での異なるサイトにデータの一貫したコピーを複数維持します。これは、トランザクションによるデータベース変更を、ソース・サーバーでコミットされた直後からターゲット・サーバーへ非同期に伝達することによって実現されます。さらに、トランザクションは保存され、トランザクションの一貫性および順序が維持されます。これにより、同一のレコードが複数回更新された場合も、変更はプライマリー・データベースに適用されたときと同じ順序でリモート・データベースに適用されます。更新のコンフリクトを回避するために、IDS ERは、さまざまな定義済みルーチンや、カスタマイズされた解決方法を通じて、組み込みの自動コンフリクト検出および解決を提供します。

フレキシビリティ

IDS ERは、ビジネスおよびアプリケーションの広範な要件を満たします。まず、主要な所有権モデルをサポートしています。すなわち、マスター/スレーブ、ワークフロー、および随時更新です。これらのために、全接続または階層的なレプリケーション・ネットワークトポロジーを利用します。

2番目に、IDS ERは、テーブルのサブセットおよびパーティションのレプリケーションをサポートしています。また、IDS ERはレプリケートされるサーバーごとに複数のスキーマを許容し、DBAに対してレプリケートされるデータベースのセクションの定義に際して、行または列レベルまでの細かい区分を提供します。この柔軟性により、DBAは固有の業務要件に合わせてデータベースをカスタマイズすることができます。例えば、トランザクション指向のデータベースで使用する作業データの整理構成は、データ・ウェアハウスで使用する、クエリーおよび分析に対応するよう設計されるデータとは異なります。

マネージビリティ

IDS ERの開発にあたっての大規模な投資は、レプリケーション・テクノロジーに対してのみでなく、同様に重要なマネージビリティについても行われました。DBAは、レプリケーション管理のタスクを、クロスプラットフォームかつブラウザ・ベースのツールであるIBM Informix Server Administrator(以下、ISA)を使用して実行できます。また、各コマンドはcdrコマンド行ユーティリティを使用して実行することができるので、動的に生成されるスクリプトの使用も容易になっています。管理コマンドを使用して、DBAは各種のレプリケーション方式を選択し、ターゲット・サーバーを指定し、読み取り/書き込みの機能を定義してからレプリケーション頻度を指定することができます。さらに、ほかのレプリケーション製品とは異なり、IDS ERを使用するDBAはレプリケーション構成を動的に変更することができ、システム全体をシャットダウンしなくとも新しいサーバーをオンラインで追加および構成することができます。トランザクションは接続が再確立されるまでキューにあるため、各サーバーの接続および切断はレプリケーションのプロセスを中断せずに実行できます。こうした機能により、DBAは環境をセットアップする際に、ネットワークのアイドル時または通信コストが低いときにのみデータを送信するようにしたり、1日の各時間に応じて、ロードのバランスを取るサーバーの組み合わせを変えたりすることができます。

もうひとつの重要な機能はグローバル・レプリケーション・カタログで、これを使用してIDS ERの構成情報を追跡記録します。これにより、レプリケーション定義、IDS ERの各サーバー、サブスライバーのリスト、デフォルトのキューイング・スレッショルドおよびコンフリクト検出と解決のルール、それらと関連づけられているストアド・プロシージャーなどの情報を維持することができます。DBAにとってのグローバル・レプリケーション・カタログの主な利点は、レプリケーション定義に対するあらゆる変更を、ネットワーク上のほかのサーバーへ自動的に伝達できることです。これにより、DBAはデータを手動でリモート・サーバーにコピーする作業から解放され、同時に、グローバル・カタログ内で不整合が発生する可能性も最小になります。モニター機能には、複製先ごとのレプリケーション・キューのサイズの監視、リモート・サーバーの可用性の表示、およびレプリケーション・サーバー間のコンフリクトの即時通知といった機能が含まれます。

IDSエンタープライズ・レプリケーション:技術的な概要

IDS ERには以下の技術があり、このセクションで詳しく説明します。

- 構成および制御
- レプリケーション・ネットワーク・トポロジー
- トランザクション・キャプチャー
- トランザクションのキューイングおよび配信
- ターゲット・サーバーでのトランザクション挿入
- コンフリクトの検出および解決

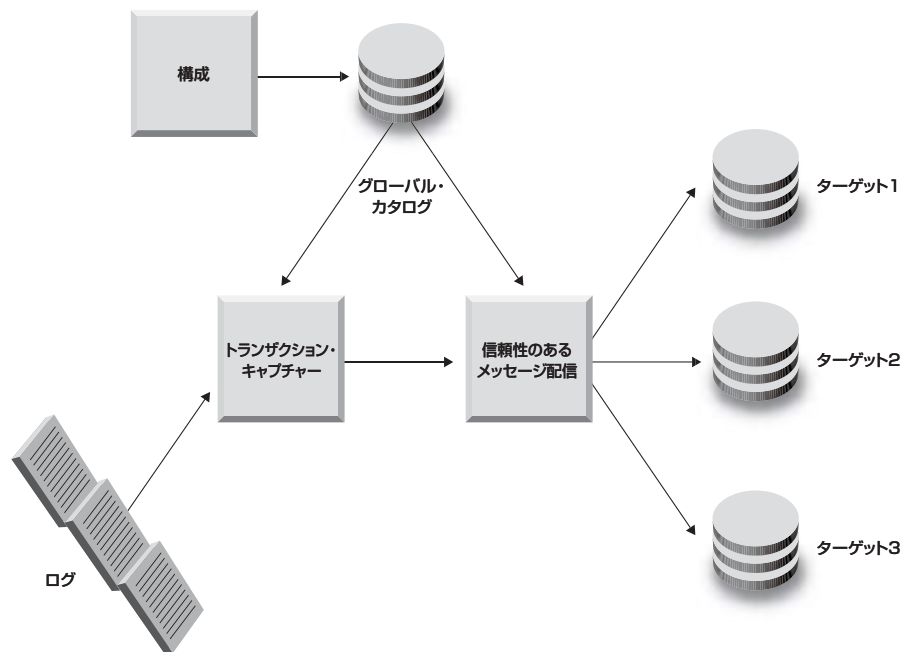


図9:IDSエンタープライズ・レプリケーションのコンポーネント

構成および制御

IDS ERは、レプリケーション定義の構成のために柔軟な方法を提供し、一貫性のあるトランザクション・キャプチャーおよび配信を実現しています。レプリケーションは、テーブルのレベルで、またはパーティションやビューといったテーブルのサブセットを使用して定義することができます。レプリケーション・グループ機能を使用すれば、レプリケートされた定義をグループにまとめることができます。レプリケーション・グループは、テーブル・レベルのすべてのレプリケーション・コマンドをグループ・レベルで適用可能にすることで、管理を単純かつ容易にします。また、この機能によってIDS ERは、異なるレプリケーション・グループにかかわる並行トランザクションを適用し、ターゲット・データベースをより高速に処理することができます。

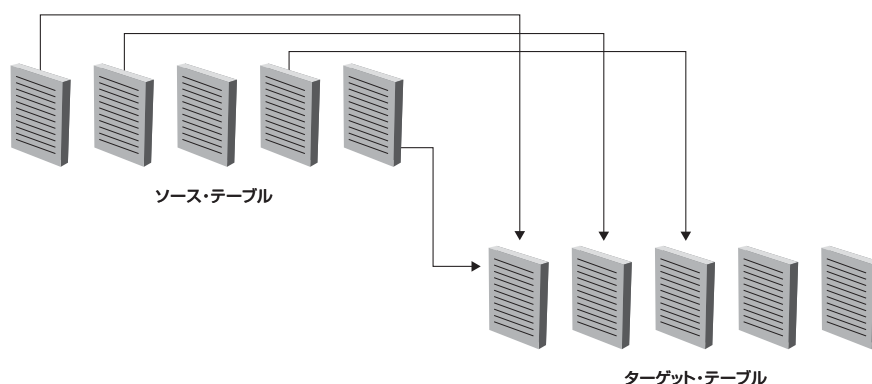


図 10:レプリケーションは、テーブルのサブセットを含むように定義できます。

レプリケーション・ネットワーク・トポロジー

エンタープライズ・レプリケーション・トポロジーは、レプリケーション・サーバーが相互に通信するために確立する接続を示します。IDS ERは、2種類のネットワーク・トポロジーをサポートします。

- 全接続
- 階層的レプリケーション

全接続トポロジーとは、すべてのデータベース・サーバーが相互に接続するトポロジーのことで、IDS ERが各接続を管理します。全接続トポロジーでは、レプリケーション・メッセージがサーバー間で直接送信されるので、それ以上のルーティングは不要です。

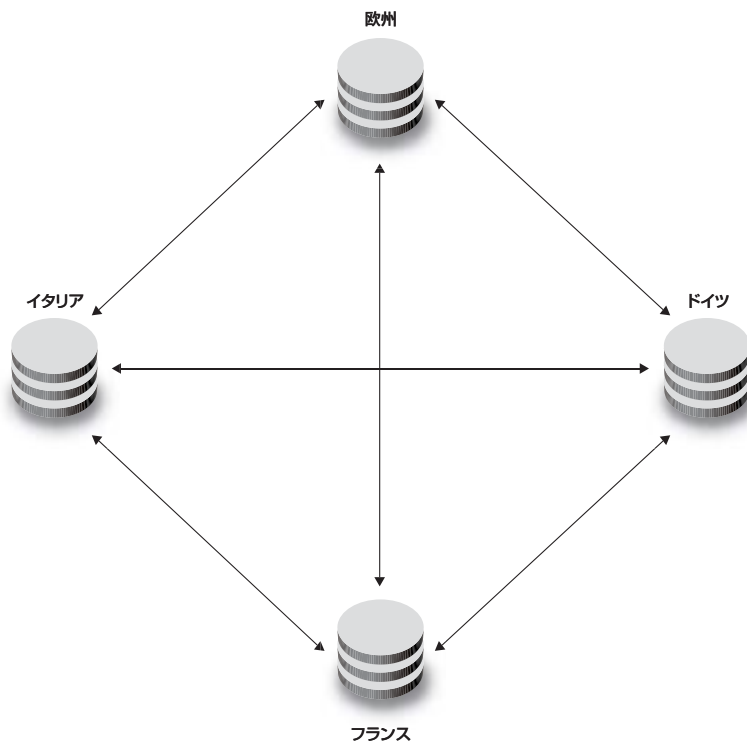


図11:全接続トポロジー

階層的ルーティング・トポロジーには、次の2種類があります。

- 階層的ツリー
- ツリーの森

階層的ツリーでは、ルートのデータベース・サーバーと、1つまたは複数のデータベースがツリー状に組織されています。ツリーにある各サーバーはその親を参照しますが、親のないルート・サーバーは例外です。

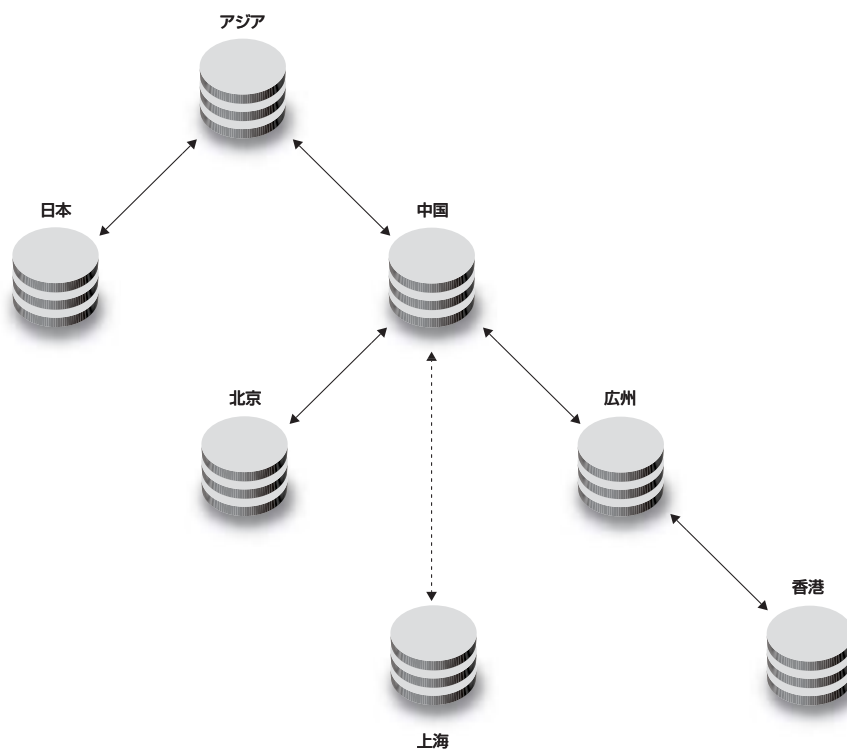


図 12: 階層的ツリー・トポロジー

「ツリーの森」トポロジーは、複数の階層的ツリーが、お互いに接続された複数のルート・データベース・サーバーを持つ形式です。図13では、北米、欧州、およびアジアがルート・データベース・サーバーであり、これらの中でレプリケーション・メッセージを転送し、各自の子ノードへ配信します。

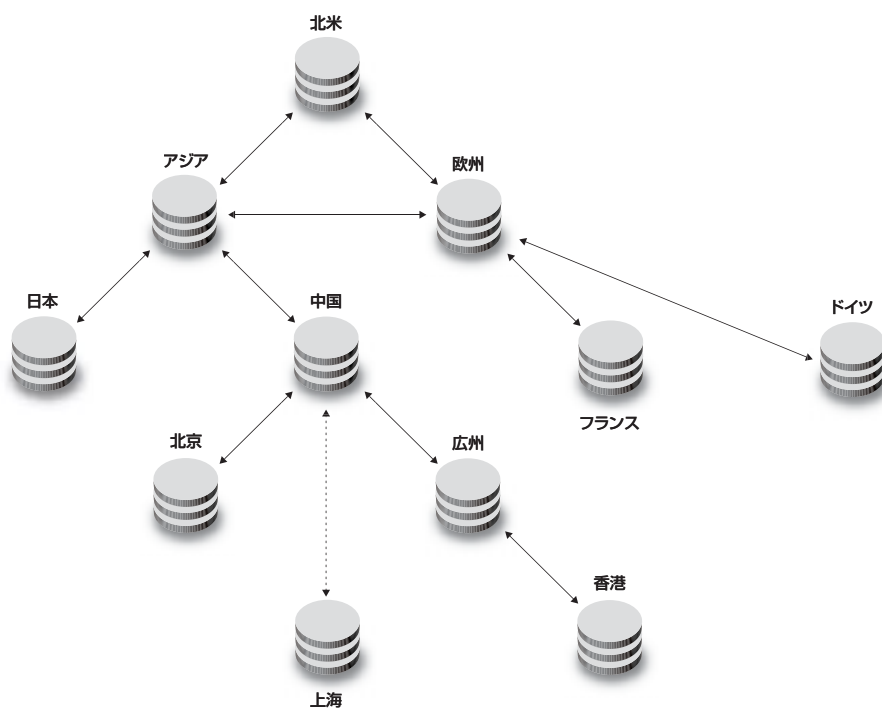


図13: ツリーの森トポロジー

データベース・サーバーを階層的ツリーまたはツリーの森に組織化することで、レプリケーション・システムを構成するために必要な物理的接続の数が大幅に削減されます。

トランザクション・キャプチャー

レプリケートされたトランザクションをキャプチャーするために、IDS ERはログの後処理という方法を実行します。後処理により、トランザクション処理への影響を最小にすることができ、独自のパフォーマンス向上を果たしています。最初に、IDS ERはログ・バッファを読み取りますが、ログの全体ではなく、レプリケーションのマークが付与されているレコードのみを対象にします。該当するレコードはフィルターにかけられ、レプリケーション定義のいずれかに適合するかどうかを判別されます。選別を通過したレコードのみが、トランザクション・グループに収集されます。トランザクションによって同一の行が複数回更新された場合でも、最後のバージョンのみが収集されます。高いパフォーマンスを確保するために、IDS ERではレコードのフィルターに並行処理を用いています。また、このプロセスはデータベース・エンジンと統合されているので、追加のI/Oが生じることもありません。

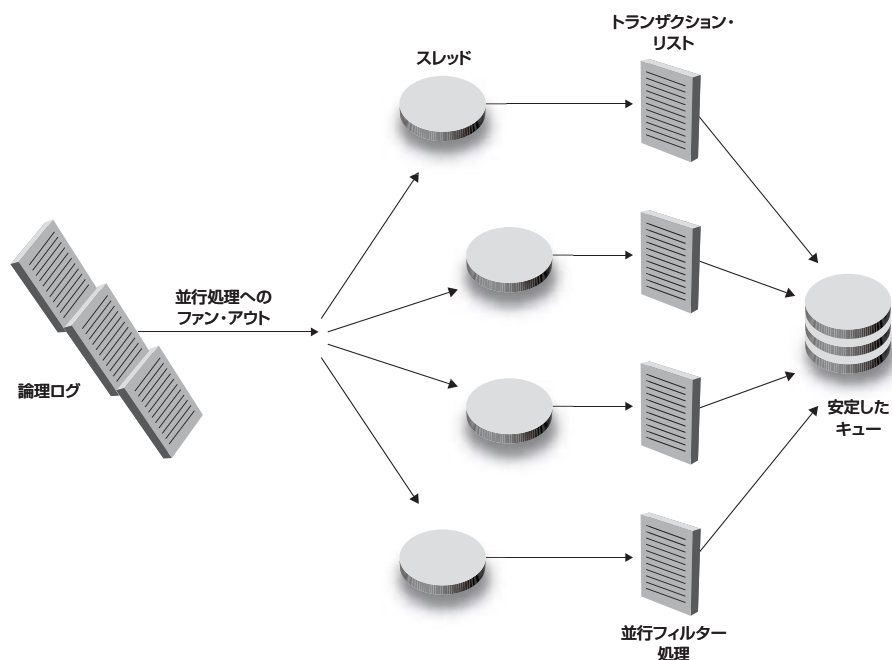


図14:エンタープライズ・レプリケーションでは、並行処理を使用してレコードがフィルタリングされます。

トランザクションのキューイングおよび配信

レプリケートされたトランザクションが実際にコミットするとき、トランザクションは配信のために「安定した」キューに配置され、中断されたトランザクションは破棄されます。DBAでは、ネットワークを通じて送信されるデータについて、標準的なデータ圧縮アルゴリズムを使用して圧縮するかどうかを選択することができます。圧縮によってデータが大幅に圧縮されれば（例えばテキストなど）、低速な接続を介したパフォーマンスを改善することができます。ネットワーク障害が発生した場合は、ネットワークが復旧するまで、安定したキューによってトランザクションが保持されます。整合性を確保するために、コミットの順序は配信中でも維持されます。ターゲット・サイトが変更を受信すると、ソース・サーバーへ通知が送信されて配信が確認されます。

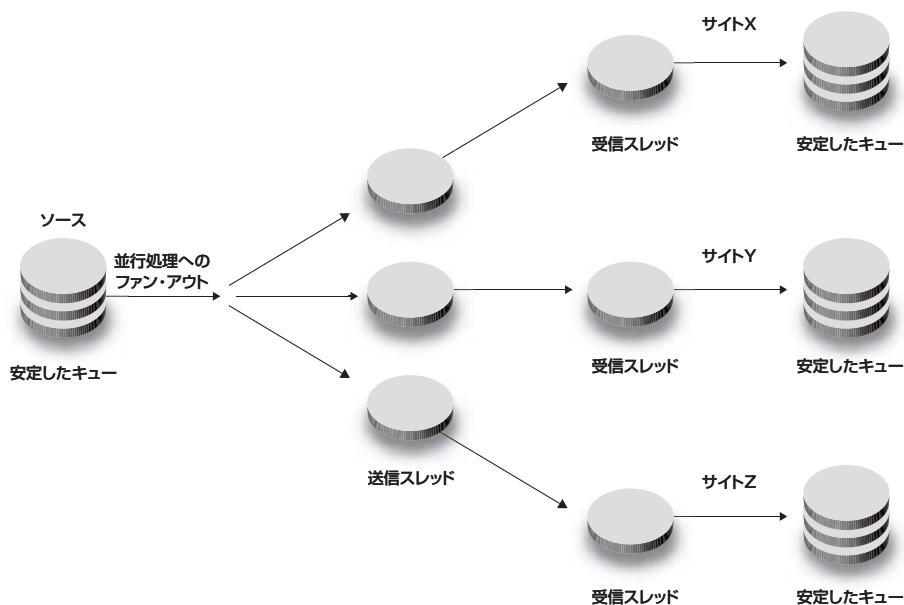


図15:トランザクションのキューイングと配信

ターゲット・サーバーでのトランザクション挿入

トランザクション・グループは、安定したキューから移されて、適切なテーブルにデータベース・トランザクションとして挿入されます。図16にこれを示します。パフォーマンスを高めるために、この処理では並行して挿入するオプションがサポートされています。レプリケートされたレコードがターゲット・テーブルの既存のレコードと同じ主キーを持っている場合、あらかじめ構成されている更新検出および解決のルールが適用されて、どちらのレコードが残されるかが決定されます。必要な構成データのすべてが、グローバル・カタログに格納されます。

コンフリクトの検出および解決

エンタープライズ・レプリケーションでは、統合された更新コンフリクト検出および解決機能が用意されており、データの一貫性を回復するためにアプリケーション固有のルーチンが起動されます。更新のコンフリクトが検出されたときは、レコードの優先度を判別するために、テーブルごとに以下のルーチンを適用することができます。

- **最新タイムスタンプ**

タイムスタンプが最新のレコードに優先度を付与します。

- **ストアード・プロシージャ**

DBAは、IBM Informix Stored Procedure Language (SPL)を使用して、コンフリクトを解決するためにアプリケーション固有のプロシージャを作成することができます。たとえば、DBAは特定のサーバーがほかのサーバーよりも高い優先度を持つように指定することができます。処理されないトランザクションを、後で調査および照合できるようにキャッシュしておくこともできます。

- **無視**

コンフリクト検出および解決を無効にするために使用します。マスター/スレーブ構成の場合に便利です。

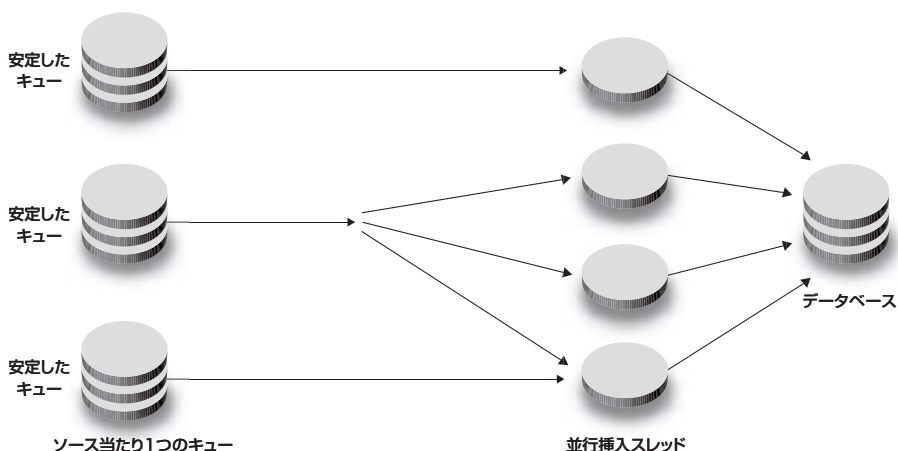


図16: トランザクション・グループは、安定したキューから移動されて、適切なテーブルにデータベース・トランザクションとして挿入されます。

既存のエンタープライズ・レプリケーション・ユーザー向け新機能

IDS ERには、最初に導入されて以来多数の改良が加えられてきました。以下に、それらをまとめます。

- 包括的なcdrユーティリティの使用により、DBAはMicrosoft® Windows®またはUNIX®システムのコマンド行からエンタープライズ・レプリケーションを構成および制御することができます。
- エンタープライズ・レプリケーションのプロセスを中断せずに、サーバーの接続および切断が可能。トランザクションは、接続が再確立されるまでキューにとどまります。この機能は、データベース・サーバーとネットワークとの負荷を均衡させるために使用できます。
- 標準的なデータ圧縮アルゴリズムを用いた、ネットワーク・メッセージの圧縮。テキストのように圧縮率が高いメッセージの場合は、低速なネットワーク環境でのパフォーマンスが向上する可能性があります。
- 大規模なネットワークでのサーバー間接続の数を大幅に削減する、階層的なルーティング・トポロジー。
- 行への最後の変更のみを取り込むことによる、トランザクションの圧縮。
- 空間オブジェクトを含む、ユーザー定義の独自の型およびスマート・ラージ・オブジェクトのレプリケーションのサポート。
- トランザクション適用時の並行処理の改善。
- シリアル主キーのサポート。
- 既存のテーブルで、再構築をしなくともレプリケーションを定義することができ、すぐに使用可能なALTER TABLE。
- 改善された、安定したキューでのスプーリング。パフォーマンスを向上させ、ブロッキング状態の確率を低減します。

IDSエンタープライズ・レプリケーションを使用する理由

データがますます散在するようになる今日の状況では、分散データベース機能は付加機能ではなく必須の機能になりました。レプリケーションは、組織がより細かな、脱中心化したビジネス・モデルへ移行するにつれて、分散データベース・システムを有効に構築する上で重要な役目を負うようになっていきます。レプリケーションを検討している組織は、各自の要件に合わせてレプリケーションをどのように運用するのが最善かを事前に計画する必要があります。レプリケーションの利点には、コンピューティング資源のパフォーマンス改善、データの可用性の向上、処理能力の余裕、および、意思決定支援を容易にするデータ・ウェアハウジングのサポートなどが含まれます。IDS ERは、エンタープライズ全体を通じて情報を配信および共有する要望を満たすために求められる必須の要素、すなわち、高いパフォーマンス、可用性、一貫したデータ配信および管理性を提供します。



エンタープライズ・レプリケーションへの第一歩

ご注文またはお問い合わせは、区域のIBM Data Management販売担当またはダイヤルIBM(0120-04-1992)へご連絡ください。IBM Data Managementのみが、レプリケーション・テクノロジーを最大限に活用し、分散コンピューティング・ソリューションを成功させるための経験、技術および人員を揃えております。IBM Informix Dynamic Serverについて、あるいはそのほかのIBM Data Management製品またはサービスについてのお問い合わせは、区域の販売店にご連絡ください。

その他の情報源

関連情報は、最寄りの販売店や、Web サイト <http://www-6.ibm.com/jp/software/data/informix/> でも得ることができます。

IBM, e(ロゴ) business software, Informixは、IBM Corporationの商標。
Microsoft, Windowsは、Microsoft Corporationの米国およびその他の国における商標。
UNIXは、The Open Groupがライセンスしている米国およびその他の国における登録商標。
他の会社名、製品名、サービス名等は、それぞれ各社の商標または登録商標。



日本アイビーエム株式会社

〒106-8711 東京都港区六本木3-2-12
11-02 Printed in Japan

IBM製品・サービスの詳細情報については、
IBMホームページ <http://www.ibm.com/jp/>
をご利用ください。

お問い合わせは、IBMビジネス・パートナー、製品販売店、弊社営業担当員、
または、ダイヤルIBM (☎ 0120-04-1992)へ、
受付時間：月～金 9:00～18:00 (祝日、12/30～1/3を除く)
携帯電話等でおかけのお客様は下記の電話番号をご利用ください。
ダイヤル IBM 044-221-1522 (この場合、通話料金はおお客様のご負担となります。)

●仕様は事前の予告なしに変更することがあります。●IBMがここに発表する情報は、日本において発表または提供されていないIBMの製品、プログラム、サービスについて参照されている場合があります。●このようなIBMの製品、プログラム、サービスへの参照は、国内での製品化、サービス化の予定を表明するものではありません。また、IBMの製品、プログラム、サービスのみが使用されていることを意味するものではありません。●IBMは、例示として説明した結果を保証するものではありません。また、顧客またはIBMビジネス・パートナーが提供する情報の正確性を証明または保証するものではありません。●ここに記載する情報は、お客様に参考としてお伝えする一般的なご案内を目的としています。●予定についての正式な対応は、IBMの今後の技術、および経営上の判断により最終決定されるものであることをご了承ください。●このホワイトペーパーの情報は、2001年12月現在のものです。●このカタログでIBMによって提供されている情報は「その時点のもの」であり、IBMはいかなる種類の誓約、担保、あるいは保証するものではありません。●IBMは、このカタログに記載されている情報にいかなる誤りまたは省略もないことについて、一切の誓約、担保、あるいは保証するものではありません。●IBMは、このカタログに含まれている情報を更新する義務を一切負いません。●製品、サービスなどの詳細については、弊社もしくはIBMビジネス・パートナーの営業担当員にご相談ください。