

第 10 章

高可用化と負荷分散

高可用化とはサーバが故障しても別のサーバへと速やかに引き継ぎ運用を継続できることです。

本章では PostgreSQL のレプリケーション機能（ストリーミングレプリケーション）とホットスタンバイ構成による高可用化と負荷分散の実現方法や仕組み、注意点などを解説します。

10.1 サーバの役割と呼び名

まず、各サーバの役割に対する呼び方を整理します(表 10-1)。

データの更新ができるサーバを、「読み書きサーバ」、または「マスタ」「プライマリ」と呼びます。プライマリ側のデータ変更を追跡するサーバを「スタンバイ」、または「スレーブ」と呼びます。また、プライマリに昇格するまでクライアントから接続できないスタンバイを「ウォームスタンバイ」、クライアントから接続できて読み取り処理のみできるスタンバイを「ホットスタンバイ」と呼びます(図 10-1)。

また、高可用化と負荷分散を実現する方式はさまざまなものがあります(表 10-2)。

表 10-1 各サーバの呼称と役割

呼称	役割
読み書きサーバ	データの読み書きを行う
マスタ	
プライマリ	
スレーブ	プライマリ側の変更を追跡する
スタンバイ	
ウォームスタンバイ	昇格するまでデータを読み書きできない
ホットスタンバイ	昇格しなくてもデータの読み込みができる

それぞれの方式で、同期のタイミングや同期される範囲(データベースクラスタ単位、データベース単位、テーブル単位など)が異なるため、要件に応じて選択することが可能です。

なお、本章では一般的に利用されるようになっているため、表 10-2 のログ SHIPPING に分類されるストリーミングレプリケーションとホットスタンバイを用いた方法を説明します。

図 10-1 ウォームスタンバイとホットスタンバイ

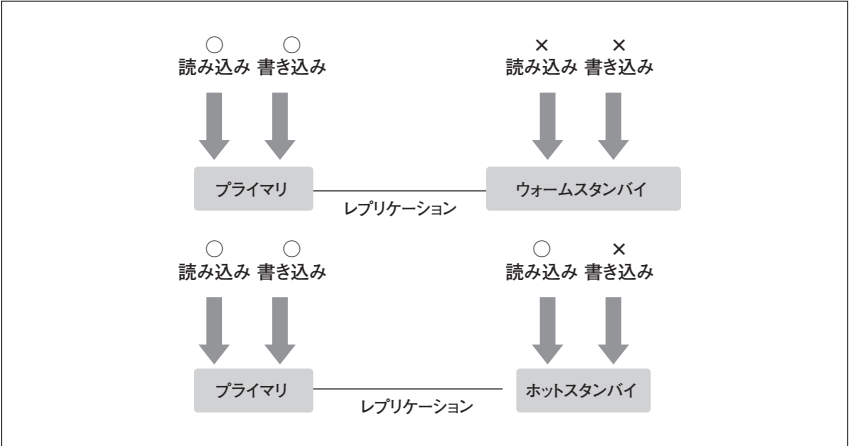


表 10-2 レプリケーションの方式 (例)

方式	特徴
共有ディスク	プライマリとスタンバイでデータベースクラスタを共有する。プライマリが稼働中はスタンバイは停止している必要がある
ファイルシステムレプリケーション	共有ディスク相当の機能をソフトウェアレベルで行い、プライマリとスタンバイでデータベースクラスタを共有する
ログ SHIPPING	WAL (Write Ahead Logging) ベースのレプリケーション
トリガベースレプリケーション	プライマリへの更新をトリガとして、スタンバイに同じ更新を伝搬する
SQL ベースレプリケーション	プライマリとスタンバイに同じ SQL を送るミドルウェアを仲介させて実現する

10.2 ストリーミングレプリケーション

ストリーミングレプリケーションはPostgreSQL本体に備わったレプリケーション機能です。WALをファイル単位ではなく、変更内容(WALレコード)単位で送り、粒度の細かいレプリケーションが可能なおことから、「流れ」を意味する「ストリーミング」が付けられて呼ばれています。PostgreSQL 9.0で非同期レプリケーションが導入され、PostgreSQL 9.1で同期レプリケーション、PostgreSQL 9.2でカスケードやwriteモードレプリケーションが導入されました。

PostgreSQL 9.3以降も利便性向上のための機能などが追加されています。歴史は浅いものの、他ツールや特別な装置を用意しなくてすむため、手軽にレプリケーションを行え、広く利用されています。

10.2.1 : ストリーミングレプリケーションの仕組み

PostgreSQLのストリーミングレプリケーションは、WAL(Write Ahead Logging)をベースに実現しています。スタンバイは、プライマリで生成されたWALを再実行することでプライマリと同じ状態を保つようになっています。

これらの仕組みを理解するために、WALや実際のプロセスについて見ていきます。

WALの特性

WALは、データベースの性能を担保しつつ、データの永続性を保証するための仕組みです。永続性を保証するということは、更新トランザクションの変更内容をどんなことが起きても必ず復元できる状態にするということです。

更新トランザクションがコミットされた際に、テーブルやインデックスといったデータファイルに直接同期書き込みを行っていると、大幅に性能が低下します。一方、WALを用いた場合、更新トランザクションのコミット

時にデータファイルへの書き込みは行わず、WALレコードのみを同期書き込みします。

リカバリ時にWALレコードを再適用(ロールフォワード)することで、性能と永続性を同時に保証します。

WALレコードにはLSN(Log Sequence Number)と呼ばれる一意の値が払い出されており、プロセスのクラッシュやオンラインバックアップ(ベースバックアップ)からのリカバリで、必要なLSN位置から順に再実行するようになっています。

ストリーミングレプリケーションは、WALのこの特性を利用してスタンバイをプライマリと同じ状態に保ちます。WALレコードを出力しないHashインデックスやunloggedテーブルなどは、ストリーミングレプリケーションではレプリケーションできません。



pg_resetxlog ツール

WALが壊れてしまった場合、PostgreSQLは読み書きすべきWALがわからなくなり、起動すらできなくなってしまいます。このような場合の復旧手段として「pg_resetxlog」というツールが用意されています。このツールを使うと、正常な位置でWALを読み書きできるようにPostgreSQL内部の制御情報を修正します。

詳しくはPostgreSQL文書「VI. リファレンス pg_resetxlog」を参照してください。

walsender/walreceiver プロセスの設定方法

プライマリとスタンバイはどのような仕組みでWALのやりとりを行っているのでしょうか。実際には、プライマリ側の「walsender プロセス」とスタンバイ側の「walreceiver プロセス」でWALのやりとりを行います。

これらのプロセスを起動するには、プライマリ側／スタンバイ側でそれぞれ設定します(表 10-3、10-4)。ファイルや設定項目が異なるので注意してください。