

# データベース同期ミドルウェアを用いた 遠隔バックアップの実現

細谷 柚子<sup>†</sup>三島 健<sup>‡</sup>小口 正人<sup>†</sup><sup>†</sup>お茶の水女子大学<sup>‡</sup>NTT ソフトウェアイノベーションセンタ

## 1 はじめに

近年ビジネスフィールドにおけるバックアップの需要が高まっている。TechTarget ジャパンによるデータ保護・管理対策に関するアンケート調査 [1] によると、約 90% がバックアップを実施していると答えたが、3分の1以上が現状に不満を抱いていた。その理由として、維持にかかるコスト以外に、技術面ではバックアップ処理におけるサービス性能への影響が挙げられており、この点に対する技術的な解決が望まれている。また、2011年3月の東日本大震災以降から遠隔バックアップの必要性が急速に求められており、その理由としてローカルのリソースだけでシステムを運用していれば、大規模災害時にはバックアップごとになってしまうおそれがある。そのため、日本のような地震国では、遠方にバックアップを置いておくことがますます重要になっている。本研究は上記バックアップの対象として、企業でのサービス提供において大規模災害に対する耐障害性保証が特に求められるデータベース・バックアップにフォーカスする。その中でも Web サービスへの適用性が高いクラスタ DB(Pangea[2]) を前提として、サービスのためのトランザクション処理に影響を及ぼさないリモートバックアップのためのモデルについて検討した。そのモデルを検証した結果、クライアントからのトランザクション処理においては、スループット性能にほぼ影響を及ぼさない方式であることを確認した。一方、Pangea を構成するサーバ台数を増加させ、トランザクション処理能力を向上させた場合に、バックアップ構成のためのトランザクション要求が積待する課題があることが分かった。

## 2 先行研究

LAN 環境を前提として、データベースの同期を行うミドルウェアの Pangea がある。Pangea は、サーバの 1 台を Leader、その他は Follower としており、クライアントからミドルウェアを介してサーバにアクセスすることで同期をとる。照会処理は 1 台のサーバで、更新処理は全てのサーバで実行される。更新処理の場合は Leader に対して更新をした後に、Follower に対しても同様に処理を行う。データベースを修正することなく

サーバを増やすことで性能を向上させることが可能である。本研究では、Pangea を拡張させた、新たなバックアップ確立手法 Pangea++ を提案する。

## 3 提案手法

クライアントからの処理を分担するローカルのサーバを用意する。既存研究と同様に、1 台を Leader、その他を Follower とする。それらとは別に、処理を分担しない遠隔バックアップ用のサーバを用意する。バックアップに関する処理は、Pangea++ を介して更新処理のみが行われるように拡張した。

図 1 のように、ローカルサーバに処理する複数の worker スレッドとバックアップに処理をする synchronizer スレッドを 1 つ用意する。worker スレッドはクライアントからクエリが来ると、ローカルサーバに対して処理を行う。また、送られてきたクエリに関して、更新クエリのみをトランザクションごとに保管し、commit された順に並べておく。これを writeset と呼ぶ。commit を契機にして、synchronizer スレッドを呼び起こす。synchronizer スレッドは、worker が保管しておいたクエリを先頭から順に取り出し、バックアップに対して処理を実行する。提案手法により、更新のみを処理させることでバックアップにかかる負荷を少なくし、より効率的にバックアップを確立させることができると推測する。

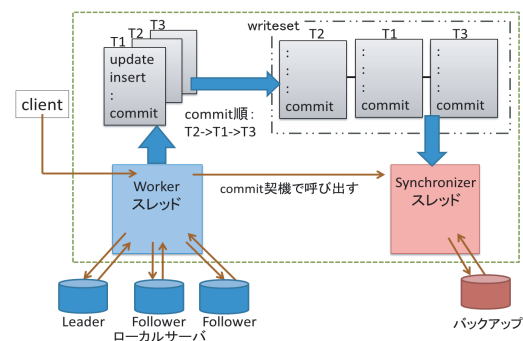


図 1: 提案手法

## 4 評価実験

### 4.1 実験環境

提案手法の性能評価実験を行った。ローカルサーバ用にマシン 3 台、バックアップサーバ用にマシン 1 台を用いた。サーバ PC は CPU が Intel(R)Xeon(R)CPU E5310 1.6GHz、OS が Ubuntu14.04、メモリが 2GB と 4GB のスペックのものを使用した。バックアップは海外にあることを想定し、dummy net を使用して RTT256ms

Realization of remote backup using a database replication middleware

<sup>†</sup> Yuzuko Hosoya, Masato Oguchi

<sup>‡</sup> Takeshi Mishima

Ochanomizu University (<sup>†</sup>)

NTT Software Innovation Center (<sup>‡</sup>)

の遅延を挿入した。データベースは PostgreSQL9.2.6 を使用，すべてのマシンに配置させた。Web サーバとアプリケーションサーバには Tomcat を用いた。性能評価は TPC-W ベンチマークを使用した。TPC-W は仮想的なブラウザ (以下 EB とする) が，それぞれ照会処理と更新処理の割合が異なる browsing mix, shopping mix, orderingmix の 3 種のワークロードでデータベースにトランザクションを発行する。性能評価指標は，スループット (1 秒あたりの WEB 画面表示:WIPS) とした。実験環境を図 2 に示す。

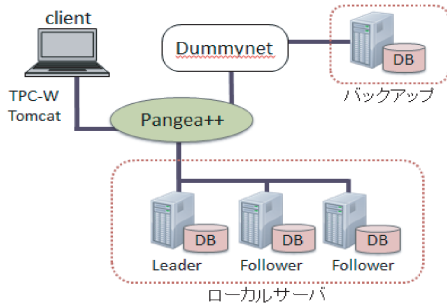


図 2: 実験環境

#### 4.2 基本性能

TPC-W の 3 種のワークロードのうち，shopping mix で実験を行った。ローカルサーバは 1 台，2 台，3 台と増やして測定し，サーバを増やした場合の性能を比較した。実験結果を図 3 に示す。



図 3: shopping mix における最大スループット

最大スループット値は，サーバが 1 台の場合は，EB が 120 のとき 14.31WIPS，サーバが 2 台の場合は，EB が 200 のとき 23.73WIPS，サーバが 3 台の場合は，EB が 300 のとき 33.65WIPS であった。ローカルサーバを増やすことで負荷分散され，性能の向上がみられた。

#### 4.3 バックアップの処理性能

次に，writeseet のキューの長さを測定し，バックアップの処理性能を調査した。低負荷時と高負荷時での処理の違いを比較するために，EB 数が 10 の場合と，最大スループット時の EB 数の場合で測定した。実験結果を図 4，5 に示す。

EB 数が 10 の低負荷の場合では，writeseet のキューの長さが頻繁に 0 になっていることから，バックアップはキューを溜めることなく，スムーズに処理ができているが，最大スループットの負荷をかけた場合では，

右上がりのグラフになっていることから，writeseet が詰まっていて効率よく処理を行えていないことがわかる。バックアップは低負荷時のようにキューを溜めずに処理をこなすことが望ましく，今後，改善すべき点であると考えられる。

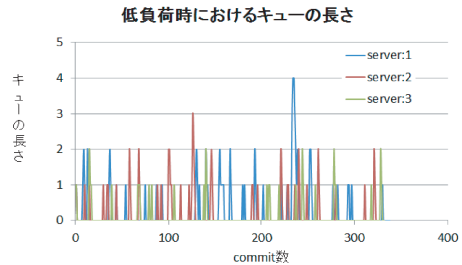


図 4: 低負荷時の writeseet のキュー長

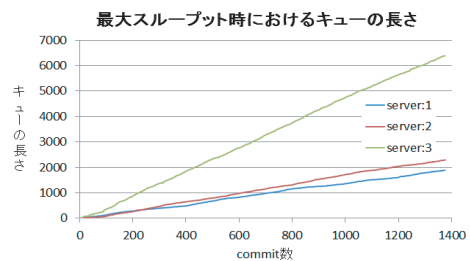


図 5: 高負荷時の writeseet のキュー長

### 5 まとめと今後の課題

データベース同期ミドルウェア Pangea を拡張したバックアップ確立手法 Pangea++の提案と評価を行った。shopping mix における最大スループット値の測定を行い，処理するサーバが増えれば，負荷が分散され，性能の向上ができることが確認できた。また，バックアップの処理性能を調査するため，writeseet のキューの長さを測定した。低負荷時にはバックアップはキューを溜めずに処理ができているが，高負荷時にはキューが溜まってしまい，バックアップのための負荷が高くなってしまったことがわかった。

今後の課題としては，高負荷時においても，キューを溜めずに処理をこなすように，提案手法の改善を行いたい。また，バックアップサーバにも負荷分散させる手法についても検討し，バックアップ確立の更なる性能向上を目指したい。

#### 参考文献

- [1] データ保護・管理対策に関するアンケートリポート <http://techtarget.itmedia.co.jp/tt/news/1109/27/news03.html>
- [2] T.Mishima, and H.Nakamura, "Pangea: An Eager Database Replication Middleware Guaranteeing Snapshot Isolation without Modification of Database Servers", PVLDB2009,424-435.
- [3] TPC-W <http://www.tpc.org/tpcw>