第5回 分散アルゴリズムを活用した連続可用KVSシステムSonicShare

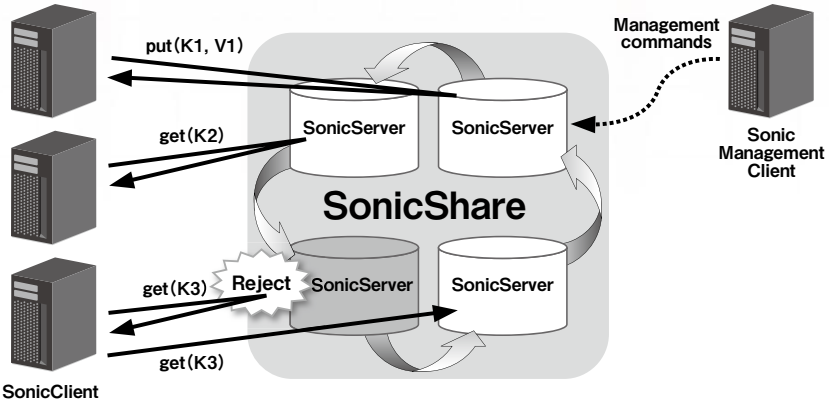
数秒の障害停止も許されないアプリケーションに、IAサーバーでも連続可用なデータ・ストアを提供

今回ご紹介するテクノロジーは、シンプルなデータ・モデルに基づくデータ・ストアとして昨今注目されているKVS（Key-Value Store）を、東京基礎研究所が有する分散システムに関する深い専門性を基に新たに開発した、高可用KVSシステムSonicShareです。高可用性を上回る連続可用性（Continuous Availability）を追求するためのアーキテクチャーやロジックを設計し、実装。約1秒間の猶予があればクライアントのリクエストに確実に応えることが可能です。このSonicShareを超高信頼のハブとしてシステムに組み込めば、システム全体の可用性やパフォーマンスを向上することができます。

技術概要

KVSは、「Key（キー）」と「Value（値）」のペアから成る非常にシンプルなデータベースの一種で、ある値（Key）に関連する値（Value）の照会・更新の処理を実行します。通常のデータベースと比べるとACID（Atomicity、Consistency、Isolation、Durability）を保証するトランザクション処理や、複雑な照会・更新はできませんが、シンプルであるがゆえにスケーラビリティーを高めることができ、膨大なデータも、複数のサーバーで分割してデータを保存するだけで性能を向上させることが可能です。また、Keyに関連するValueの挿入・更新・削除というシンプルなデータ更新のため、比較的容易に複数のサーバーに更新の複製ができ、高い可用性を実現することができます。SonicShareは、この特性を生かし、数秒の障害停止も許されない分野において極めて高い可用性を実現するために、GBS（Global Business Service）事業のアセット（資産）であるMCT/CIS※1の1コンポーネントとして、IBM独自のアーキテクチャーやロジックを設計、および、実装しています。

図は、4台のサーバー（SonicServer）を構成するシステム例で、クライアント（SonicClient）からは4台のサーバーが1つに見えています。SonicShareが稼働する各SonicServer、各SonicClientはTCP/IPで通信しており、クライアントがデータを照会（get）・更新（put）する際は、1台のサーバーが内部的に自動で選択され、リクエストが送信されます。リクエストを受信したサーバーは、照会の場合はそのサーバーで、更新の場合はすべてのサーバーで処理が行われます。すべてのデータがすべてのサーバーで保存されているため、リクエストを送信したサーバーのレスポンスが遅く、障害が疑われる場合は、クライアントはほかのサーバーに同じリクエストを送信することが可能です。障害が疑われる時間を調整することで、どんなサーバー障害・ネットワーク障害でも、その障害が単一障害ならば、すべてのデータ照会・更新を、例えば、1秒未満で処理することが可能になります。この仕組みにより、SonicShareは、CAP定理※2でいう、可用性（Availability）と分断耐性（Partition Tolerance）を保証し、連続可用性を追求しています。



データ更新のタイミングはサーバーごとに少しずつ遅れが生じるので、あるリクエストに対して、1つのサーバーからはAというデータが、ほかのサーバーからはBというデータが返ってくる状態が発生します。これは、一般的なデータベースでは許されないことですが、SonicShareは、このデータのずれが発生する時間を1秒間だけ許容します。その代わり、それぞれのサーバーは、ほかのサーバーからどのくらい遅れているかを常にチェックしており、最終的にはすべてのサーバーが同じデータを保存することを保証します。また、クライアントからリクエストを受け取った時に、ネットワーク障害などでほかのサーバーから1秒以上遅れてしまった場合は、いったんクライアントにリダイレクトの通知を返し（Reject）、クライアントが別のサーバーから整合性の取れたデータを取得するよう指示します。これらの仕組みにより、CAP定理において犠牲となる一貫性（Consistency）を、最大限配慮することを可能にしています。また、すべての処理はSonicShareのライブラリー内で完結しているので、クライアントがリダイレクトするプログラムは不要になります。さらに、長時間、同期が取れないサーバーは管理専用クライアントで確認することができ、管理者はSonicShareの状態を正確に把握することが可能です。

※1 IBM Multi Channel Transformation/Channel Integration Server：IBM WebSphere Application Serverで稼働し、SOA（サービス指向アーキテクチャー）化推進の基盤となるソリューション・アセット（担当：石田考朗 itis@jp.ibm.com）。

※2 分散システムにおいて、一貫性（Consistency）、可用性（Availability）、分断耐性（Partition）のすべてを保証できないことを示した定理。

ここに注目！

究極の可用性を実現するアーキテクチャー

すべてのサーバーにコピーするアクティブ・レプリケーションで、すべてのデータが各サーバーのメモリーで共有され、お互いに同期を取ります。万が一サーバーに障害が発生した場合でも、ほかのサーバーが稼働していれば、1秒未満で確実に処理を継続することができます。また、特定の機器に依存しないシンプルな構造であるため、汎用IAサーバーでも動作し、ACIDを保てない代わりに、究極の可用性を実現します。

データの鮮度に基づく新しい一貫性モデル

データの更新は、タイム・スタンプとともにすべてのサーバーに複製されます。ほかのサーバーと同期した時刻から現在時刻までの時間でデータの鮮度を判断し、決められた鮮度が保たれていない場合は、ほかのサーバーに再リクエストすることで、データの一貫性を保証します。SonicShareは、データの鮮度に基づいてデータの一貫性を保つ新しいモデルを確立しました。

既存のアルゴリズムを効果的に活用

シンプルなデータ・モデルのデータ・ストアであるKVSをベースに、データの不整合を排除するためのLamportの論理クロック※3や、柔軟なシステム構成を実現するためのGossipプロトコル※4といった既存分散アルゴリズムを効果的に利用しながら、独自のアーキテクチャーやロジックを設計、実装しています。

※3 各サーバーにおいて事象の前後関係を認識することで、分散システム上の時刻を実現するプロトコル。

※4 サーバー間のデータの伝播（でんぱ）を実現するプロトコル。新規に受信したデータのみを転送する。

ゲートウェイ・システムやセッション管理などで効果を発揮

ゲートウェイ・サーバーのように最終的なデータの一貫性がバックエンドのサーバーで保証されているシステムであれば、一貫性の緩和されたキャッシュとして利用可能で、金融機関のゲートウェイ・サーバーにおいても、SonicShareの効果が発揮されています。また、最終的に同じ値になることが保証されることを利用して、HTTPセッションの管理や、オークション・システムなどへの応用も期待されています。

日本アイ・ビー・エム株式会社
IBM東京基礎研究所
主任研究員

堀井 洋 Hiroshi Horii

IBMのWebSphereブランドには、IBM WebSphere eXtreme Scaleという製品がありますが、東京基礎研究所のSonicShare開発チームの部門は、そのプロトタイプの開発段階からかかわっていました。WebSphere eXtreme Scaleは、ハイパフォーマンスや高可用性、拡張性を重視したトランザクショナルなKVSとして製品化されていますが、同製品の拡張機能などを研究する中で、日本のお客様はより高い可用性への要求が非常に高いことから、可用性を極める研究が続けられました。

SonicShareでは、複数のサーバーにコピーを保存することで高可用性を実現していますが、従来はこの方法でアプリケーションに最適なデータの一貫性を提供することが困難でした。そこで指定された時間の遅れを許容する一貫性を表すデータの鮮度というアイデアを盛り込むことで、高可用性と共に、アプリケーションに適した一貫性を実現することができたのです。

現在、SonicShareは、すでに日本のお客様にご利用いただいていますが、IBMグローバルのアセットとして登録され、その国々に特化した要求を解決するようなアルゴリズムや考え方を追加していくことで、より広くお客様にご利用いただけるように発展させていきたいと考えています。

東京基礎研究所のSonicShare開発メンバー