

ORACLE®
NOSQL DATABASE

時系列データ向けOracle NoSQL Database

Oracle ホワイト・ペーパー | 2017 年 12 月



ORACLE®

はじめに

膨大な量のデータが生成され、それらのデータを保存し、分析する必要性が増す中で、この洪水のような情報を処理できる効率的なシステムの導入が求められています。モノのインターネット（IoT）とビッグ・データ、双方の環境において、データベースに保存されるデータ量を単純化して低減し、他のソフトウェアを使用してさらに分析できるようにしようとする動機には説得力があります。たとえば、必要のない古いデータを削除できる機能や、もはや価値がないデータに有効期限を設けることなどで、それを実現できます。

すべてのデータがタイムスタンプと関連付けられているわけではありませんが、多くの種類のデータは、ある一定期間タイムスタンプと関連付けられている場合に限り意味を持ちます。データがタイムスタンプと関連付けられている場合、開発者はその情報を使用して、過去の行動の分析や未来の行動の予測を行うことができるアプリケーションを作成できます。

しかしながら、ストレージや問合せのほか、すでに意味や価値がなくなっている“古い”データの使用は、分析アプリケーションの円滑な実行に影響を及ぼす可能性があります。また、システム全体の非効率性のために、意思決定プロセスが遅れる可能性もあります。

時系列の例

情報のほとんどがタイムスタンプとともに生成される一般的な区域としてセンサーが挙げられます。たとえば、ビルの一角で温度や照明をモニタリングするセンサーは、温度や照明の量と併せて時間が記録されている場合は重要なデータ・ポイントです。データはタイムスタンプとデータから構成され、測定データだけを保存した場合のおよそ2倍になります。

組立ラインから成る製造オペレーションには、多数のデータ収集ポイントがあり、データは関連のタイムスタンプとともに取得され、保存されます。このデータは、単なるセンサーの読取り値ではなく、画像や動画を含む場合もあります（車のフロントガラスを枠に装着する場合など）。このデータは、障害モードが発生している可能性を判断する際は価値がありますが、そのうちのほとんどは、長期間データベースに保存する必要はありません。ロボットによるフロントガラスの装着を記録するこの例では、車が組立ラインの品質保証工程に合格したら、この大量のデータは破棄して構いません。

時系列データは、財務的イベント、工場の自動化、家庭用電気製品、公共料金、モバイル機器でも生成される可能性があります。時系列データは通常、膨大な量の“小規模”レコードです。このデータが長期間にわたり取得され、保存されると、データの総量がシステムのストレージ容量を容易に超えてしまう可能性があります。

タイムスタンプを使用したデータ管理ソリューション

迅速な意思決定を下すためのデータや、過去と比較するためのデータなど、さまざまな種類のデータを統合する総合的なソリューションには、データを素早く取り込み、信頼性のあるデータベースでデータを保護することができるキャパシティが必要です。データは、ネットワークの端（センサー）から、待機時間の短いデータベースへと流れ、その後、長期間にわたり取得される他の情報とともに、リレーショナル・データベースに長期間保存されます。

たとえば、Oracle NoSQL Database は、数ミリ秒の待機時間で膨大な量のデータを確実に保存できるため、短期的なデータベースとして使用できます。不要になった古いデータを削除した後は、残りの必要なデータを、さらなる処理のためにリレーショナル・データベースに保存できます。このアーキテクチャでは、データベース間を移動しなければならない情報の量が減少します。

Time-To-Live (TTL)

Oracle NoSQL Database には、データベースにデータを保持する期間を開発者が設定できる API 機能があります。Time-To-Live は、開発者がレコードの有効期限を指定できる仕組みです。この仕組みでは、期限切れになったレコードはシステムから自動的に削除されます。

開発者は、レコードを保存する期間を時間または日数で指定できます。その後、問合せによって期限切れのレコードがフィルタリングされ、NoSQL Database は領域を再利用されます。この機能は、ある限られた期間のみデータが必要なさまざまなアプリケーションで有用です。たとえば、長期間非アクティブなユーザーのプロファイル・データを消去したい Web アプリケーションなどで使用できます。また、チェーン店での顧客の購入パターンを追跡する際にも使用できます。顧客がある商品を購入する都度、そのデータを保持する期間を指定します。その顧客がその特定のチェーン店ではもう購入しないと判断されると、その顧客の購入プリファレンスがデータベースから削除されます。

TTL は、クリーンアップによりシステム・パフォーマンスが向上する点や、プログラマーが容易に実装できるという点からも、このような種類の操作を行う場合の効率的な方法です。

例

以下の例では、データの行に 5 日間の Time-To-Live を設定する方法を示します。これは、TTL を使用するアプリケーションが、データの行に有効期限を設定する方法と同様です。

```

package kvstore.basicExample;

import oracle.kv.KVStore;
import oracle.kv.table.Row;
import oracle.kv.table.Table;
import oracle.kv.table.Table.TimeToLive;
import oracle.kv.table.TableAPI;

...

// 簡略するために kvstore ハンドルの作成は省略します

...

TableAPI tableH = kvstore.getTableAPI();
Table myTable = tableH.getTable("myTable");

// 行インスタンスを取得
Row row = myTable.createRow();

// TTL 値を行に追加
row.setTTL(TimeToLive.ofDays(5));

// 行のすべてのセルを出力。
row.put("item", "Sensor");
row.put("description", "Temp and Humidity in Office");
row.put("temp", 75.3);
row.put("humidity", 0.49);

// 表をストアに書き込む。
tableH.put(row, null, null);

```

将来のある時点で（5 日間の有効期限が過ぎる前に）、アプリケーションが行に問合せを行うことができる日数を延長したいとします。これは簡単に行うことができます。行インスタンスを取得し、Time-To-Live の日数の値を新しい値に変更し、行をデータベースに再び保存するだけです。

また、有効期限を設けないようにデータを設定する必要がある場合は、以下を使用します。
`row.setTTL(TimeToLive.DO_NOT_EXPIRE);`

時間に基づきデータの有効期限を管理する利点

データベースにさらにデータが追加されるのに伴い、使用リソースという観点からデータベースのサイズが増大するだけでなく、検索時間も増加する場合があります。高いパフォーマンスを維持し、十分な空きストレージ容量を確保するためには、問合せで不要になったデータは、データベースから永続的に削除する必要があります。以下の図では、保存されているデータにさまざまな TTL 値が設定されています。TIME 軸を右側に移動するにつれ、TTL に基づきデータベースからレコードが削除されます。データベースのサイズは縮小するため、新たなレコードを挿入できるようになります。

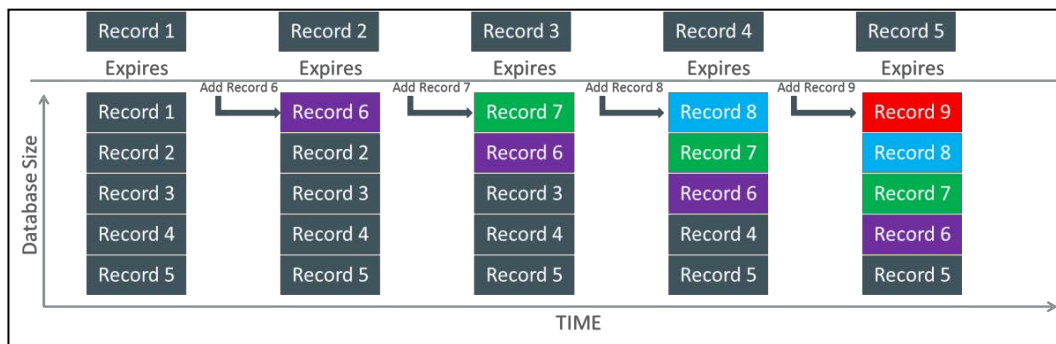


図1：古いレコードが期限切れになると、レコードがさらに挿入される

時系列データ向けに設計され、最適化されている Oracle NoSQL Database

膨大な量の時系列データを処理する場合、基盤となるデータベースが予測可能な待機時間で高速なレスポンスを実現できるように設計されていることが重要です。Oracle NoSQL Database の“スマートなクライアント・ドライバ”は、常に読取りリクエストをもっとも効率的なノードにルーティングします。つまり、どのようなレコード・フェッチも、“単一のリクエスト・メッセージ/単一のレスポンス・メッセージ”の方法で実行できるため、予測可能な短い待機時間が保証されます。

Oracle NoSQL Database では、セカンダリ・キーの索引付けがサポートされるため、セカンダリ・キー列で条件を指定することにより、レコードを効率的に取得することが可能です。Oracle NoSQL Database の 2 次索引は、パーティション化され、シャード固有です。シャードごとに 1 つの索引パーティションがあり、各索引パーティションは、そのシャードのレコードのみを参照します。つまり、2 次索引は、索引付けするレコードとともに、トランザクションによって更新されます。また、(すべてのシャードの) すべての索引パーティションに平行でアクセスすることで、2 次索引の検索を極めて効率的に実行できます。その結果、2 次索引へのアクセスが効率的に行われるだけでなく、常に正確かつ完全な結果セットが返されます。データベースによっては、2 次索引は非同期で更新されます。つまり、そのようなデータベースでは、2 次索引への問合せで不完全な、または古い結果が返される可能性があります。

まとめ



Oracle NoSQL Database では、レコードがデータベースに保存される期間を開発者が決定できる API が提供されます。この機能を使用すると、データベースに保存されるデータ量を低減できるとともに、データの検索時間やスキャン時間を短縮できます。一部のデータはある一定の期間のみ価値があるため、この機能は、アプリケーション開発者が保存の必要なデータ量を管理し、パフォーマンスを向上させるのに役立ちます。



Oracle Corporation, World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065, USA

海外からのお問い合わせ窓口
電話：+1.650.506.7000
ファクシミリ：+1.650.506.7200

CONNECT WITH US

-  blogs.oracle.com/oracle
-  facebook.com/oracle
-  twitter.com/oracle
-  oracle.com

Integrated Cloud Applications & Platform Services

Copyright © 2017, Oracle and/or its affiliates. All rights reserved. 本文書は情報提供のみを目的として提供されており、ここに記載される内容は予告なく変更されることがあります。本文書は一切間違いがないことを保証するものではなく、さらに、口述による明示または法律による黙示を問わず、特定の目的に対する商品性もしくは適合性についての黙示的な保証を含み、いかなる他の保証や条件も提供するものではありません。オラクルは本文書に関するいかなる法的責任も明確に否認し、本文書によって直接的または間接的に確立される契約義務はないものとします。本文書はオラクルの書面による許可を前もって得ることなく、いかなる目的のためにも、電子または印刷を含むいかなる形式や手段によっても再作成または送信することはできません。

Oracle および Java は Oracle およびその子会社、関連会社の登録商標です。その他の名称はそれぞれの会社の商標です。Intel および Intel Xeon は Intel Corporation の商標または登録商標です。すべての SPARC 商標はライセンスに基づいて使用される SPARC International, Inc. の商標または登録商標です。AMD、Opteron、AMD ロゴおよび AMD Opteron ロゴは、Advanced Micro Devices の商標または登録商標です。UNIX は、The Open Group の登録商標です。0116

時系列データに最適な Oracle NoSQL Database
2017 年 11 月
著者：Michael Schulman
共著者：Ashok Joshi



Oracle is committed to developing practices and products that help protect the environment