

WINTERCORP レポート

INTERSYSTEMS  
スピード・テスト

オペレーショナル・クラウド・データベース・  
システムのパフォーマンスと  
データ遅延の比較

.....  
委託調査  
プログラム

大規模なデータ管理のエキスパート

W I N T E R C O R P O R A T I O N

# INTERSYSTEMS スピード・テスト

.....  
オペレーショナル・クラウド・データベース・  
システムのパフォーマンスと  
データ遅延の比較

RICHARD WINTER  
NORBERT KREMER  
ATIF MAJID  
TENNYSON YESUPATHAM



WinterCorp

[www.wintercorp.com](http://www.wintercorp.com)

42 DERBY LANE, TYNGSBORO, MA  
617-695-1800

大規模なデータ管理のエキスパート

**結論:** 独立した調査に基づき、WinterCorp は、低遅延、高パフォーマンスのトランザクショナル分析データ管理ソフトウェアを必要とする企業は InterSystems IRIS を詳しく検討してみることをお勧めします。InterSystems IRIS は、AWS 上にてシングル・ノードおよび 1 ~ 4 ノード・クラスタでテストしたあらゆる競合製品と比べて優れた性能を発揮します。競合製品と比較して、InterSystems IRIS はクエリのスループット、挿入のスループット、データ遅延、およびクエリ効率において大幅な優位性を示しており、特殊な調整や構成も不要です。

## S サマリー

重要なクラスのデータベース・アプリケーションは、以下のような重要なパフォーマンス要件を同時に複数満たす必要があります。

- 大量のトランザクション処理とデータ取り込み
- 大量のクエリ
- 高い一貫性（挿入後すぐにきわめて短い遅延でレコードを取得する能力を含む）

例えば、公開株の価格変動をモニタして瞬時に対応するために使用しているデータベースがあり、1 日あたり数十億件の取引があるとします。このような企業に必要なのは、ポートフォリオと市場データをモニタして、リスクの影響度を計算し、どの銘柄をどの程度売買するかを判断することです。こうした判断を他のトレーダーに先んじて下すことができれば、競争上強力な優位性が得られ、ビジネスに多大な影響をもたらします。証券市場で発生する新規取引はほとんどの場合、データベースに数ミリ秒以内に反映される必要があります。同様のニア・リアルタイムの要件を持つデータベース・アプリケーションは、そのほかに工業、商業、エンジニアリングといったアプリケーションにも多数あります。

このようなアプリケーションでは、トランザクションとレコードの構造は比較的シンプルですが、そこで求められる厳しい要件を満足するのはまったく容易ではなく、高スループットと低データ遅延を 1 日中一貫して提供しなければなりません。

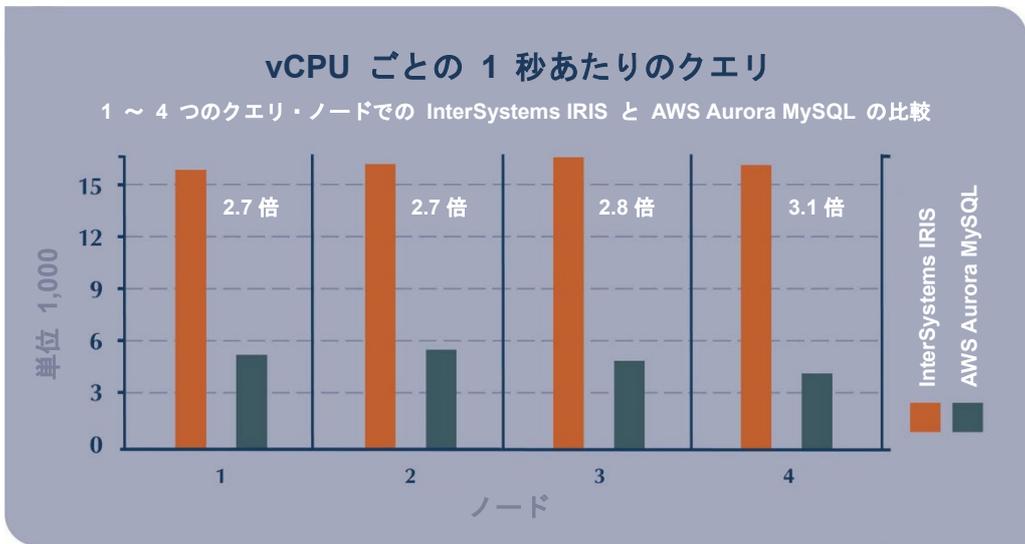


図 1: vCPU ごとの 1 秒あたりのクエリ数、1 ~ 4 つのクエリ・ノードでの InterSystems IRIS と AWS Aurora MySQL の比較

このような機能を顧客に実証したいという希望から、データ管理ソフトウェア・プロバイダである InterSystems 社は、InterSystems スピード・テストを考案しました。このテストでは、前述の基準でデータ管理ソフトウェアのパフォーマンスを簡単に測定できます。

本レポートでは、弊社において InterSystems スピード・テストを単一ノード構成とクラスタ構成で実行した経験を紹介し、その結果を示し、結論をまとめます。

弊社のテストで判明した重要な結果は次の 5 つです。

1. InterSystems スピード・テストは、さまざまなオペレーショナル・データベース・アプリケーションの重要な構成要素を測定する有効な方法です。

2. 弊社が独立して実施した単一ノード・テストの結果から、InterSystems IRIS には、AWS Aurora MySQL、MariaDB、Microsoft SQL Server、Oracle、および PostgreSQL と比較してパフォーマンス上大きな優位性があることが実証されました。

3. InterSystems IRIS の挿入速度は、他のシステムより 1.7 ~ 9 倍高速でした。また、データ・クエリ速度は、他のシステムより 1.1 ~ 600 倍高速でした。**挿入速度とクエリ速度の組み合わせにおいて、InterSystems IRIS に迫るシステムはありませんでした。**

4. 弊社は、上記のシステムの 1 つである AWS Aurora MySQL と InterSystems IRIS を比較するクラスタ・テストを実施しました。このテストでは、AWS Aurora MySQL の挿入速度が InterSystems IRIS にほぼ一致する構成を選びましたが、そのためには AWS Aurora MySQL を IRIS の 4 倍のプロセッサ能力と 4 倍のメモリを持つサーバで実行する必要がありました。その上で、仮想 CPU あたりのクエリ速度（AWS でのコンピューティング能力の尺度）を比較しました。この測定（3 ページの図 1 を参照）では、**InterSystems IRIS は AWS Aurora MySQL よりも 2.7 ~ 3.1 倍効率的で、効率面での優位性はクラスタのノード数が増えるにつれて大きくなりました。**つまり、InterSystems IRIS はリニアスケールを示しましたが、AWS Aurora MySQL はリニアスケールに達しませんでした。

5. InterSystems IRIS は、実行した 3~5 分間のクラスタ・テストの期間中、一貫して低いデータ遅延を示しました。つまり、InterSystems IRIS によって挿入されたレコードは、挿入後きわめて短時間でクエリに現れました。一方、図 2 に示すように、AWS Aurora MySQL でレコードを挿入してから以降のクエリで取得できるようになるまでにはかなりの遅延がありました。この遅延は 7 分間のテストの間中続きました。

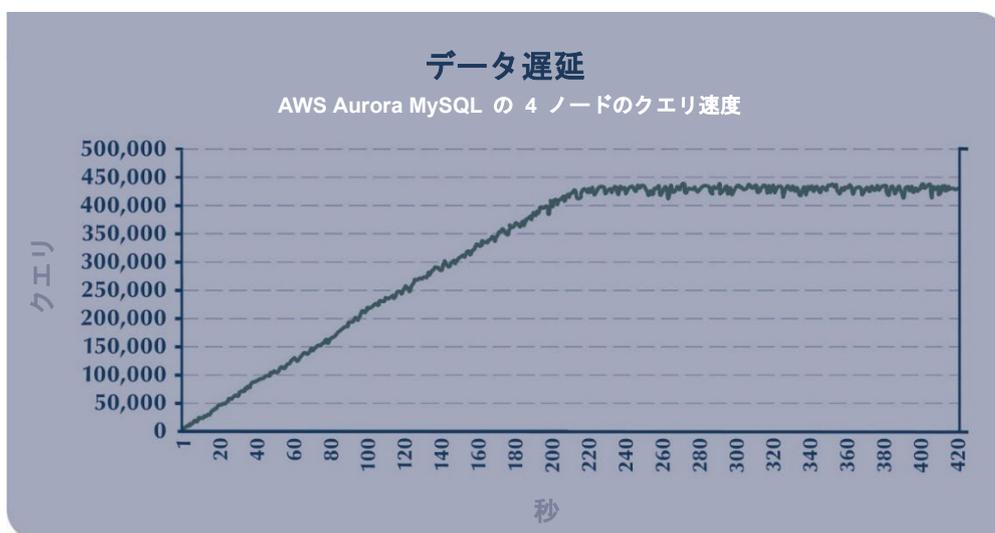


図 2: 7 分間のテスト中、データ遅延の持続が明白

低データ遅延は、すべてのニア・リアルタイム・データベース・アプリケーションで必須です。例えば、証券の価格変動をモニタするシステムは、取引時に瞬時に反応する必要があります。新たに挿入された

データを取得できるようになるまでに大きな遅延があると、多大な損失を招くおそれがあります。

6. AWS Aurora MySQL において 5 ノード・クラスタ構成で正常にテストを実施するのは難題でした。これを解消するには、テスト・クライアントとテスト・スレッドの数を通常ではあり得ないほど大きな数にまで意図的に増やすしかありませんでした。これが示唆するのは、AWS Aurora MySQL では、**弊社がテストしたビジネス・ソリューションと同じような要件を持つソリューションを使用しようとすると、複雑なアプリケーション・アーキテクチャが必要になる**ということです。弊社は前に AWS Aurora MySQL で 3 つ以上のリーダ・ノードを使用したテストを実施しようとして失敗していますが、AWS サポートでは解決策を特定できませんでした。対照的に、**InterSystems IRIS は 1 ~ 5 ノード構成ですぐに動作しました。**

## 1 高パフォーマンスのオペレーショナル・データベースの課題

データベースは、現代のほぼすべての企業で心臓部にあたります。アナリティクスと意思決定の支援のためだけに存在するデータベースもありますが、それ以外の多くのデータベースは企業の活動の瞬間瞬間を実際に記録、制御、管理します。このようなオペレーショナル・データベースでは、最も厳しいパフォーマンス要件が要求されることがほとんどです。つまり、瞬時にクエリに応答し、真のファイアホース<sup>(※注1)</sup>を通じて新しいデータを到着と同時に統合し、常に最新の情報を、多くの場合受信から数ミリ秒以内に返す必要があります。要求される挿入、クエリ、更新の量は膨大になる可能性があります。

このような要件の例は製造、流通、e コマース、設備管理、輸送分野で数多く見られますが、最たる例は金融サービスです。金融サービスでは多くの企業が有価証券のポートフォリオを管理しており、その価値はほんの一瞬で変わる可能性があります。多くの場合、価格が変化するとプログラム売買がトリガされ、有価証券ポートフォリオのバランス化を図ったり、公表指標などの基準との関係を維持したりといった処理が行われます。このようなデータベースでは、新しいトランザクションを瞬時に統合し、数ミリ秒以内に、最新の情報を必要とするクエリに応答する必要があります。

このような要件を持つ顧客は、どのデータベース製品が自社のニーズに最適であるかを検証する方法を必要としています。InterSystems スピード・テストが設計されたのも、この目的のためです。

(注1) インターネット上のソーシャルメディアやオンラインサービスなどによって生成される大量のデータストリーム

## 2 InterSystems スピード・テスト

InterSystems スピード・テストでは、テスト対象データベースに、格納すべき新しいレコードのストリームが絶えずフィードされます。同時にシステムには、データベースのレコードに対するクエリのストリームが絶えずフィードされます。新しいレコードにはそれぞれ一意のキーがあり、このキーを使用してレコードを取得できます。

このキーの値 10 万個を使用してクエリと挿入を繰り返しました。テスト・ハーネスでは、レコードの取得に成功した完了クエリの数をカウントしました。テストはデフォルトで 5 分間実行され、以下を示すレポートが生成されます。

1. 挿入されたレコードの数
2. レコードの取得に成功したクエリの数
3. テスト期間全体を対象とした、各秒の終了時点における上記の値のグラフ

テスト・ハーネスには 2 種類の「ユーザ」プロセスが含まれており、ライタは挿入をデータベースに送信し、リーダはクエリを送信して応答を処理します。ライタとリーダはどちらも複数の「スレッド」を持つことができます。各スレッドは単純にそのタスクを繰り返します。ライタ・スレッドはレコードをデータベースに挿入して応答を待機し、その後すぐに別のレコードを挿入します。挿入速度とクエリ速度は、テスト対象データベースが「飽和」するまで、つまり処理可能な作業をすべて実行している状態になるまで、徐々に上げることができます。スレッドを追加すると最初のうちは各速度が上がっていきませんが、ある時点で、クライアント・プロセスが実行されているサーバのリソースが枯渇します。その時点でクライアント（ワーカとも呼びます）を追加すると、さらに速度を上げることができます。

したがって、一部の広範な条件下では、データベースに適用するワークロードをきわめて高いレベルにまで（つまり、テスト対象データベースが実行可能な作業をすべて実行している状態になるまで）増やすことができます。

一定の期間は、上記の項目 1（挿入されたレコードの数）がデータベース取り込みスループットの尺度になります。上記の項目 2（レコードの取得に成功したクエリの数）は、クエリのスループットと遅延の尺度です。上記の項目 3（テスト・プログラムによって生成されるグラフ）は、テスト期間全体のスループットと遅延の推移を示します。これらのアイテムについては、各テストの結果の説明と併せて、後続のセクションで図を交えて少し詳しく説明します。

### 3 WinterCorp のテスト・プロセス

WinterCorp はこのテストを Amazon クラウドで実施しました。また、クラウドの経験を持つシニア・データベース・エンジニア 2 名をテストのために採用しました。弊社の実践的テストを行うエンジニア両名は、さまざまな商用データベース製品の実装と、そのパフォーマンス・テストの実施において何年もの経験を有しています。また、AWS でのデータベース実装経験も豊富です。

このテスト・アクティビティの監視には WinterCorp のシニア・アーキテクト 2 名があたりました。両名ともデータベースのパフォーマンスに関して豊富な経験を持っており、うち 1 名はクラウド・アーキテクト、実装者、トレーナーとして認定されています。

WinterCorp は独立系技術コンサルタントとして、データベース・パフォーマンス・テスト、データベース・アーキテクチャ、データベースの実装、データベースのパフォーマンスとスケーラビリティの分析に 20 年以上にわたって携わってきました。弊社は大手企業各社、ユーザ、ベンダ、システム・インテグレータにご利用いただいております。前述のようなテスト目的で 50 超の別個の契約を結んでいます。

弊社は InterSystems 社が開発したテスト・ハーネスを使用する手法を採りました。ただし、構成とデプロイについては、InterSystems IRIS と比較対象製品を最も正当にテストできると考える方法で独自に行いました。作業は 2 段階で進められました。

第 1 段階では、InterSystems 社がテスト対象データベースの単一サーバ（単一ノードとも呼びます）構成で実施したテストを再現しました。この段階の目的は 2 つありました。1 つ目はテスト・ハーネス、テスト・プロセス、および出力について実際に経験することであり、2 つ目は最もシンプルな形式のテストで独自の検証結果を生成することでした。この第 1 段階では、InterSystems IRIS、AWS Aurora MySQL、Microsoft SQL Server、MariaDB、Oracle、および PostgreSQL をテストしました。

第 2 段階では、InterSystems IRIS と AWS Aurora MySQL のクラスタ構成を、1 つのライター・ノードと最大 4 つのリーダー・ノードでテストしました。

すべてのケースにおいて、テスト対象データベース製品にはデフォルトの設定を使用しました。テストの実行に問題が生じた際のサポートへの問い合わせを除き、製品ベンダは関与していません。サポートの利用時にも、ベンダにパフォーマンスに関するアドバイスを求めることはなく、テストを実行するための支援のみを求めました。

それでは以下に、個々のテストと、テストに関する問題について詳しく説明します。

### 4 単一ノード・テストの結果

単一ノード・テストでは、InterSystems 社が作成したデフォルトのハードウェア構成をそのまま使用しました。テスト・ハーネスは AWS *c5.xlarge* インスタンス<sup>1</sup> 上で実行しました。テスト対象のデータベース製品の実行には、AWS *m5.xlarge* インスタンス<sup>2</sup> か、該当製品で使用できる最も近いものを使用しました。

<sup>1</sup> Amazon AWS *c5.xlarge* インスタンスは 4 つの vCPU、8 GiB RAM、最大 4.75 Gbps EBS 帯域幅、および最大 10 Gbps のネットワーク帯域幅で構成されています。

<sup>2</sup> Amazon AWS *m5.xlarge* インスタンスは 4 つの vCPU、16 GiB RAM、最大 4.75 Gbps EBS 帯域幅、および最大 10 Gbps のネットワーク帯域幅で構成されています。

InterSystems IRIS には、クエリ・ワーカを 1 つと取り込みワーカを 1 つ使用しました。取り込みワーカは 15 スレッド、クエリ・ワーカは 10 スレッドで構成しました。キーの数は 8 です。つまり、8 個の別個のキー値を使用して挿入を繰り返し、クエリはすべてこの 8 個のキーに対して実行しました。取り込みバッチ・サイズは 1,000 レコードで、テストの実行時間は 20 分でした。

InterSystems IRIS、AWS Aurora MySQL、および Microsoft SQL Server を同様の構成で実行した結果は、次のとおりです。

製品	挿入の数 (1 秒あたり)	完了クエリの数 (1 秒あたり)	AWS サーバの種類
<i>AWS Aurora MySQL</i>	12,985	15,640	db.r5.xlarge
<i>MS SQL Server</i>	15,569	3	db.m5.xlarge
<i>InterSystems IRIS</i>	128,322	19,379	m5.xlarge

表 1: AWS Aurora MySQL、SQLServer、InterSystems IRIS の単一ノード・テスト

InterSystems IRIS の挿入速度は、AWS Aurora MySQL のほぼ 10 倍、MS SQL Server の 8 倍超でした。InterSystems IRIS の完了クエリ速度は AWS Aurora MySQL と比べて約 24%、SQL Server と比べて 6,000 倍超高速でした。

テストしたその他 3 つのシステムでは、AWS Marketplace を使用して、より大規模なサーバを実行する必要がありました。したがって、これらのシステムについては、InterSystems IRIS を *m5.2xlarge* サーバ（最も同等な条件になる選択肢）で実行しました。これらのテスト結果を表 2 に示します。

製品	挿入の数 (1 秒あたり)	完了クエリの数 (1 秒あたり)	AWS サーバの種類
<i>Maria DB</i>	32,908	11,362	db.m5.2xlarge
<i>Oracle</i>	50,360	42,378	db.r5.xlarge
<i>PostgreSQL</i>	110,299	12,872	db.m5.2xlarge
<i>InterSystems IRIS</i>	193,013	47,537	m5.2xlarge

表 2: MariaDB、Oracle、PostgreSQL、InterSystems IRIS の単一ノード・テスト

これらのテストでは、InterSystems IRIS の挿入速度は、他のシステムと比べて 1.7 倍 (PostgreSQL) ~ 5.9 倍 (MariaDB) 高速でした。InterSystems IRIS のクエリ速度は、他のシステムと比べて 1.1 倍 (Oracle) ~ 4.2 倍 (MariaDB) 高速でした。

どのテストにおいても挿入速度とクエリ速度の両方で InterSystems IRIS に匹敵するシステムはなく、両方の指標で高いパフォーマンスを必要とするアプリケーションに適しているのは InterSystems IRIS だけであることを示しています。有価証券の価格/ポートフォリオをモニタするアプリケーションにはこの両方が必要なのは明らかです。同様に、大容量かつニア・リアルタイムの要件を持つその他多くのトランザクショナル分析データベース・アプリケーションでもこの両方が必要です。

## 5 クラスタ・テストのプロセスと結果

単一ノード・テストでは InterSystems IRIS に大きな優位性があることが実証されていますが、一部のオペレーショナル・データベース製品はクラスタ構成でも動作可能であり、クラスタ構成であればスループットが向上するはずだと考えました。

そこで InterSystems 社に対し、テストを拡大してクラスタ構成をテストすることを提案し、InterSystems 社の同意を得ました。

この方法で InterSystems IRIS と比較する最初のシステムとして、AWS Aurora MySQL を選びました。AWS Aurora MySQL はテストを実行する AWS にネイティブであるため、最も簡単だろうと考えたからです。

最初に、InterSystems IRIS で一連のクラスタ・テストを実行してベースラインを確立しました。

ノード	挿入の数 (1 秒あたり)	完了クエリの数 (1 秒あたり)	完了クエリの数 (各ノード1秒あたり)
1	99,515	60,189	60,189
2	100,356	123,819	61,910
3	94,076	186,387	62,129
4	101,876	248,910	62,228

表 3: InterSystems IRIS のクラスタ・テスト、1 ~ 4 つのクエリ・ノード

これらのテストでは、挿入はすべて1つのノードで処理し、クエリは1~4つのノードで処理するように InterSystems IRIS を設定しました。挿入速度は4つのテストすべてで100,000/秒の範囲内でした(表に示すように概ね94,000 ~ 102,000/秒の範囲内でした)。ただし、クエリを処理するノードの数を増やすと、クエリ速度がノード数よりもわずかに増加しました。したがって、InterSystems IRIS は全般的に高い挿入速度を維持しながら、クエリ速度ではリニアスケールを上回り、ほぼ理想的なパターンを示しました。

AWS Aurora MySQL では、「レプリカ」を使用してクラスタ・アーキテクチャに最も近い構成を実装しています。Aurora は挿入に1つのノード、クエリに1~4つのレプリカ・ノードを使用するように設定しました。

弊社は、顧客が所定の挿入速度をサポートする必要がある場合に使うであろうテスト方法でテストを実施することにしました。そこで、InterSystems IRIS の挿入速度にほぼ匹敵するまで、AWS Aurora MySQL ノードのサイズを徐々に増大させました。この段階に達したのは、AWS Aurora MySQL を、16個の仮想CPUと64 GiB RAMをそれぞれ搭載した db.r5.4xlarge ノードで実行したときでした(これを、InterSystems IRIS を実行した m5.xlarge ノードと比較してみてください。こちらのノードは4個の仮想CPUと16 GiB RAMです。したがって、**AWS Aurora MySQL のノードは、InterSystems IRIS のほぼ4倍の計算リソースとメモリ・リソースを使用したことになります**)。これらの大きなノードを使用することで、AWS Aurora MySQL の挿入速度を約94,000/秒、つまり InterSystems IRIS の挿入速度の5%以内にまで引き上げました。その上で、リーダー・ノードを1つから4つまで増やしながら、AWS Aurora MySQL クエリ速度をテストするというのが弊社の計画でした。

ところがその後数週間、問題に直面することになります。リーダー・ノードが1つの場合と2つの場合は、比較的すぐにテスト結果が得られました。しかし、3ノード構成と4ノード構成では AWS Aurora MySQL でクエリを適切に分散できませんでした。テスト・ハーネスとテスト対象の AWS Aurora MySQL システムのさまざまな構成を試した後、AWS サポートに連絡を取りましたが、有効な回答は得られませんでした。結局、サポート契約をアップグレードして AWS Aurora MySQL の専門家とビデオ通話できるようにしました。この通話は数時間に及びましたが、やはり解決策は見つかりませんでした。

最終的には、弊社のプロジェクト担当最高シニア・エンジニアの判断でテスト・ハーネスの同時セッションの数を大幅に増やし、結局 28 のクライアント・プロセス（8 つの取り込みワーカと 20 のクエリ・ワーカ、それぞれに多数のスレッド）でテストを実行しました。その結果、表 4 に示すように、AWS Aurora MySQL クラスタは明らかに正しく動作するようになりました。

ノード	挿入の数 (1 秒あたり)	完了クエリの数 (1 秒あたり)	完了クエリの数 (各ノード1秒あたり)
1	93,976	88,586	88,586
2	93,668	181,276	90,638
3	93,185	262,294	87,431
4	92,304	324,673	81,168

表 4: AWS Aurora MySQL のクラスタ・テスト、1 ~ 4 つのクエリ・ノード

表 5 からわかるように、AWS Aurora MySQL には効率とスケーリングの両方に問題があることを示しています。

ノード	AWS Aurora		InterSystems IRIS		IRIS の優位性
	v CPU	vCPU ごとの 1 秒あたりのクエリ	v CPU	vCPU ごとの 1 秒あたりのクエリ	
1	16	5,537	4	15,047	2.7x
2	32	5,665	8	15,477	2.7x
3	48	5,464	12	15,532	2.8x
4	64	5,073	16	15,557	3.1x

表 5: InterSystems IRIS は AWS Aurora MySQL よりも vCPU あたり  
2.7 ~ 3.1 倍のクエリを実現し、効率的にスケーリング

InterSystems IRIS にはハードウェアを有効利用する優れた能力があることに加え、スケーラビリティにも優位性があることから、4 ノードでの仮想 CPU（vCPU）あたりの完了クエリ数で測定すると、3.1 倍のクエリ処理効率という並外れた優位性が生じます。さらに、IRIS の優位性はノード数の増加に合わせて拡大します。ほぼすべての顧客は処理要件の経時的増加に直面するため、これは重大な問題です。

図 6（次のページを参照）にこれをグラフで示します。オレンジ色の棒は InterSystems IRIS のスループットを、暗緑色の棒は AWS Aurora MySQL のスループットを表します。さらに、2.7 ~ 3.1 倍にわたる大きな数値は、ノードの数を 1 つから 4 つへと増やすのに応じて InterSystems IRIS がもたらす優位性の大きさを示しています。

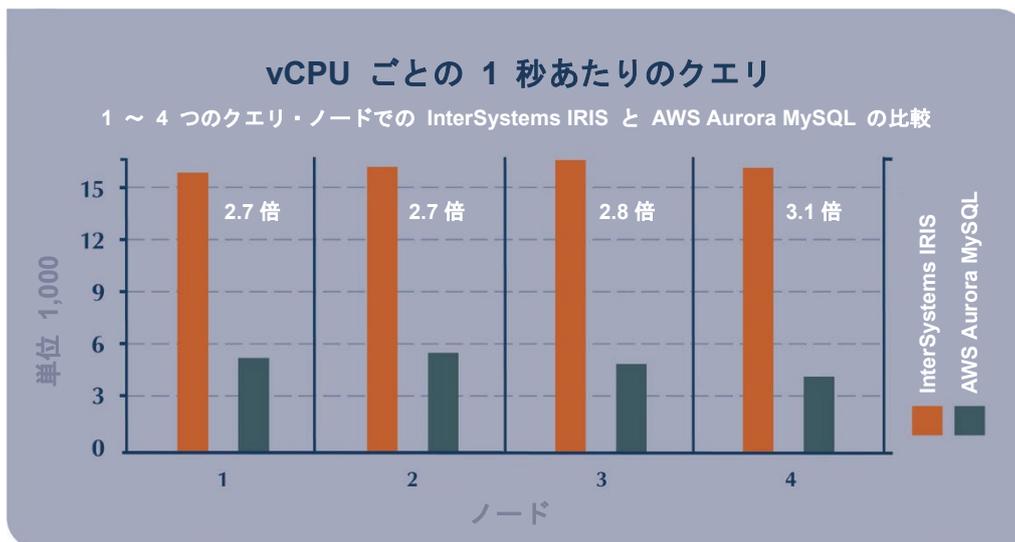


図 6: InterSystems IRIS は AWS Aurora MySQL よりも vCPU あたり 2.7 ~ 3.1 倍多くのクエリを実現し、ノード数の増加に応じて優位性が拡大

## 6 データ遅延

弊社のテストから InterSystems IRIS と AWS Aurora MySQL 間の違いがもう 1 つ明らかになりました。それは多くの顧客にとって最も重要な違いです。つまり、**InterSystems IRIS のデータ遅延がはるかに小さい**ことです。これが最も顕著なのは、図 7 と 8 に示すように 4 ノード・テストの場合です。「完了クエリ」という用語は、データベース内で探しているレコードが見つかったクエリのことを指します。レコードが見つからないクエリは「未完了」と見なされます。

クラスタ・テストでは、挿入するレコードの固有キーの数を 100,000 に増やし、公開有価証券を扱う巨大な環境にさらに近づけました。挿入速度から、テストで生成された 100,000 個のキーはテスト開始後 2 秒でデータベースに反映されることがわかっています。

InterSystems IRIS の場合（図 7 のオレンジ色の曲線）、完了クエリ速度は早い段階で最大値に達し、テスト期間中、最大値またはほぼ最大値で一定しています。これはデータ遅延が非常に小さいことを示しています。つまり、挿入後すぐにレコードを取得できるため、ほぼすべてのクエリで、探しているレコードが見つかります。

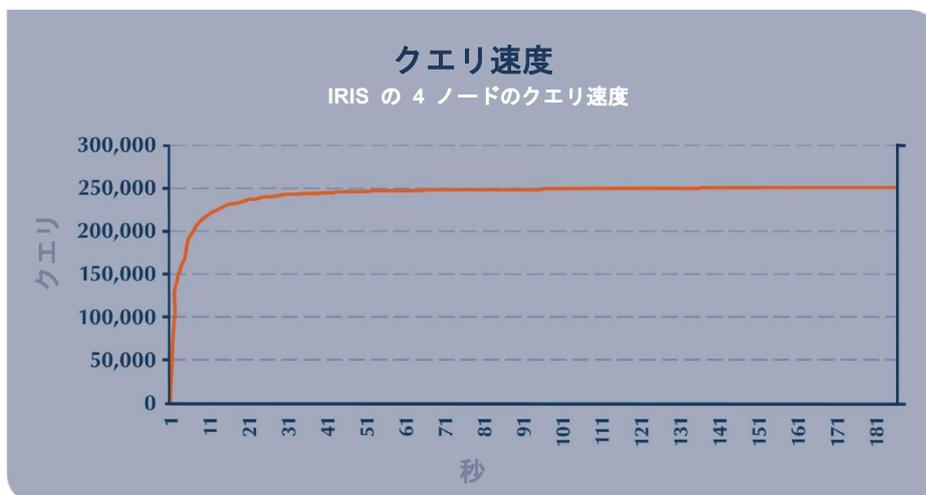


図 7: 4 ノードの InterSystems IRIS クラスタのクエリ速度

AWS Aurora MySQL の完了クエリ速度（下記の図 8 の曲線）は、InterSystems IRIS のクエリ速度より一貫して遅れています。これは 2 つの問題を示しています。まず、AWS Aurora MySQL は、レコードを十分な速さで挿入しながら同時にクエリを実行することができません。これはデータベース・アーキテクチャの根本的な問題です。2 つ目に、AWS Aurora MySQL に挿入されたレコードは、挿入後しばらく経つまで、または新しい値が挿入された後でしか取得できません。弊社の考えでは、この原因は AWS Aurora MySQL が挿入後にレコードをデータ・ストレージに実際に複製する必要があるからです。AWS Aurora MySQL の挿入速度が上がるほど、レプリケーションの遅れが大きくなります。

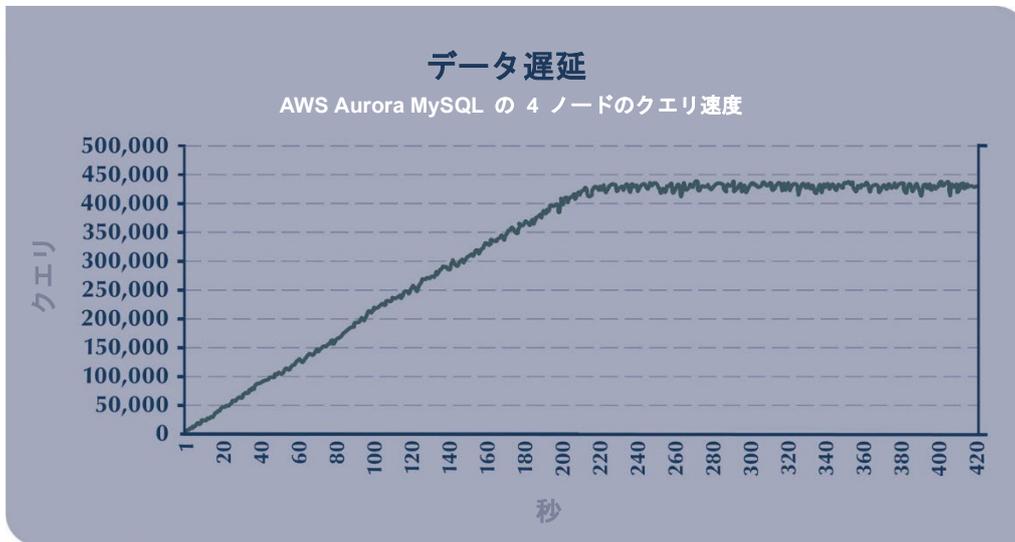


図 8: 4 ノードの AWS Aurora MySQL クラスターのクエリ速度

InterSystems IRIS は AWS Aurora MySQL より大幅に低い遅延を実現できます。つまり、新しいレコードや新しい値がデータベースに保存されてからクエリで IRIS 内に確認できるまでの時間は、InterSystems IRIS の方がはるかに短くなります。

図 9 に示すように、この理由は、InterSystems IRIS には独自の Enterprise Cache Protocol (ECP) を使用する分散インメモリ・キャッシュが実装されているためです。このため、ライブ・データはインメモリ・キャッシュ内に保持され、キャッシュにないデータはオンデマンドで取得されます。

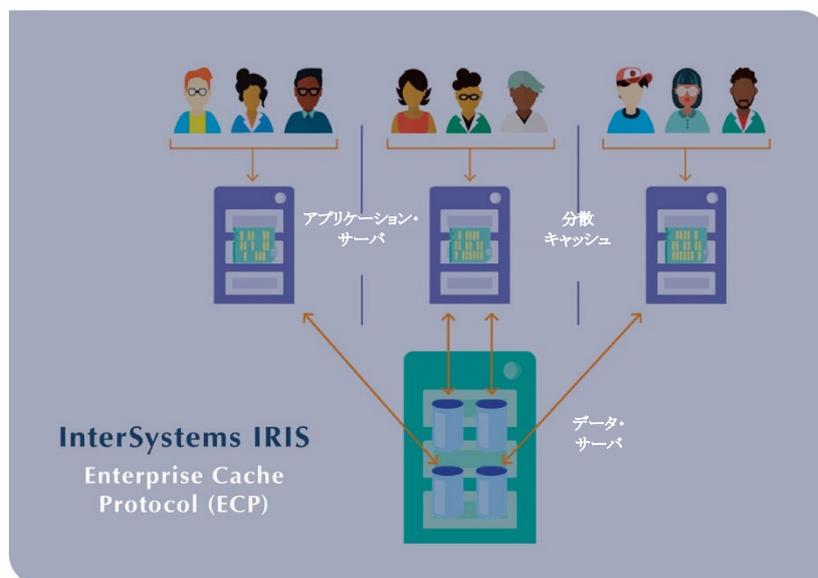


図 9: InterSystems IRIS の分散キャッシュ・アーキテクチャ

データがデータベースにアップロードされると、古いバージョンは、トランザクションの一部としてキャッシュされていたすべてのノードから削除されます。このキャッシュとオンデマンドでのデータ取得により、取り込みとクエリを高速に実行しても一貫性が保証されます。対照的に、AWS Aurora MySQL が使われているのは従来のディスクベースのレプリケーション・スキームであり、挿入時にデータ・ストレージに対してより多くの読み取りと書き込みを行う必要があります。

## 7 結論と推奨事項

弊社のテストからわかることとして、InterSystems IRIS は唯一、単一のデータベースに対してクエリと挿入を両方同時に実行しても高スループットと低遅延を実現できます。AWS Aurora MySQL、MariaDB、Microsoft SQL Server、Oracle、および PostgreSQL と比較した場合、InterSystems IRIS は AWS 上の単一ノード・テストで際立った優位性を示します。

さらに、前述の領域のほかに、1 ~ 4 ノードのクラスタ・テストでのデータ遅延とアプリケーションのシンプルさにおいても、InterSystems IRIS は AWS Aurora MySQL に対して際立った優位性を示します。

独立した調査に基づき、WinterCorp では、クラウド向けの高パフォーマンスなトランザクショナル分析データ管理ソフトウェアをお探しの企業は InterSystems IRIS を詳しく検討してみることをお勧めします。\*

**WinterCorp は、最新のデータベース管理エコシステムのアーキテクチャと戦略を  
専門とする独立系コンサルティング会社です。**

1992 年の設立以来、弊社は、データベース管理において最も困難かつ  
要求の厳しい課題に対応するソリューションの設計・構築に世界中で携わってきました。

弊社はお客様のデータ関連ビジネスの対象を定義するところから、  
データ戦略とアーキテクチャの開発、データ・プラットフォームの選定、  
ビジネス価値を最適化するためのソリューションの設計まで  
幅広くお客様を支援しています。

お客様は弊社の支援の下、クラウド、マルチクラウド、およびハイブリッド・クラウドの  
各アーキテクチャの作成・実装や、データ・サイエンス、人工知能、機械学習に必要な  
データ基盤の構築を進めています。

お客様はアナリティクスによってビジネス上の成果を実現し、  
そのリターンは多くの場合、投資の 10 倍以上になります。

必要であれば、弊社はベンチマーク、概念実証、パイロット・プログラム、  
システム・エンジニアリング調査の作成・実施も手がけ、重大な技術リスクの管理、  
コスト管理、ビジネス目標の達成においてクライアントを支援いたします。

弊社は、構造化データ、非構造化データ、半構造化データのエキスパートであるだけでなく、  
あらゆる主要形式のデータ・アナリティクス向けデータ管理製品、ツール、  
技術にも精通しています。

その深い知識と経験により、**WinterCorp** はスケーラビリティの支障となっている問題や、  
ビジネスを成功に導く技術と手法について比類ない洞察をお届けします。



**WinterCorp**

[www.wintercorp.com](http://www.wintercorp.com)

42 DERBY LANE, TYNGSBORO, MA

617-695-1800

©2022 Winter Corporation, Tyngsboro, MA. All rights reserved.