Veritas™ Volume Replicator 計画およびチューニングガイド

Solaris

5.0

N18951F

symantec™

Veritas Volume Replicator
計画およびチューニングガイド

Veritas Volume Replicator 5.0

サードパーティ（第三者）製ソフトウェアの権利に関する通知

本製品には、特定のサードパーティ製ソフトウェアが配布、組み込み、または同梱されている場合があります。また、本製品のインストールおよび使用にともない、サードパーティ製ソフトウェアの使用を推奨する場合があります。同サードパーティ製ソフトウェアのライセンスは、著作権の保有者により別途付与されます。サードパーティのソフトウェアの使用に必要なライセンスおよび著作権に関する情報については、本製品リリースノートのサードパーティに関する章を参照してください。

ライセンスと登録

Veritas Volume Replicator はライセンスが必要な製品です。ライセンスのインストールについては、『Veritas Volume Replicator インストールガイド』を参照してください。

テクニカルサポート

製品のサポートを受けるには、http://support.veritas.com ページへアクセスし「Phone Support」または「E-mail Support」をクリックします。このページから TechNote、Software Alerts、ソフトウェアのダウンロード、ハードウェア互換性リスト、VERITAS Email Notifications サービス などにアクセスすることもできます。「Knowledge Base Search」機能を使用し、製品ドキュメントのリリースなどの製品情報へアクセスすることができます。

目次

第 1 章 レプリケーションの計画と設定

第 2 章 レプリケーションパフォーマンスのチューニング

1

レプリケーションの計画と設定

効率的な Veritas Volume Replicator（VVR）設定を構築するには、VVR の各種コンポーネントがどのように相互作用するかを理解しておく必要があります。この章では、この相互作用について説明し、VVR 環境を構成するために必要な情報を示します。29 ページの「レプリケーションパフォーマンスのチューニング」の章では、パフォーマンスに関する特記事項を説明します。本書は VVR の概念を理解しているユーザーを対象としています。詳しくは、『Veritas Volume Replicator 管理者ガイド』の概念の説明を参照してください。

理想的な設定が組まれている場合は、レプリケーション対象のデータは、アプリケーションがローカルディスクに書き込むのと同じ速さでレプリケートされます。その結果、すべてのセカンダリホストが最新の状態に保たれます。プライマリホストのデータボリュームへの書き込みは、様々なコンポーネントによりネットワークに送信され、最終的にセカンダリホストのデータボリュームに到達します。セカンダリのデータを最新の状態に保つには、設定内の各コンポーネントが、プライマリで行われた書き込みに、セカンダリでも即時に対応できるようにする必要があります。また、プライマリでの書き込みやネットワークトラフィックの一時的な急増など、不定期にレプリケーションのボトルネックが発生したとしても、VVR がレプリケーションを行うことができるようなシステムを構成することが必要です。

VVR を構成するコンポーネントの 1 つに書き込み遅延が発生し、それが長期化した場合は、データボリュームへの書き込み遅延によるアプリケーションのスローダウン、セカンダリでのデータアップデートの遅延、または SRL のオーバーフローが発生する可能性があります。プライマリの書き込みプロセスで経由するコンポーネントが必要な速度を保てないと、データ書き込みに遅延が生じ、アプリケーションのパフォーマンスが低下します。また、プライマリシステムで、ローカルのデータボリュームへの書き込みプロセスに関与しないコンポーネントの速度が低下した場合、プライマリの書き込みが通常の速さで進行しても、たとえばネットワークが遅いために、SRL に書き込み履歴が蓄積される可能性があります。その結果、セカンダリは遅延し、最終的には SRL がオーバーフ

ローします。したがって、各コンポーネントを調査し、そのコンポーネントが、想定されるアプリケーションの書き込み速度に対応できるようにすることが重要です。

このマニュアルでアプリケーションという用語は、データボリュームに直接書き込みを行うプログラムを指しています。データベースが、データボリューム上のファイルシステムを使っている場合は、ファイルシステムがアプリケーションとなります。データベースが直接データボリュームに書き込みを行う場合は、データベースをアプリケーションであると見なします。

VVR のデータフロー

この項では、VVR でのデータの流れ、レプリケーション時に VVR でカーネルバッファがどのように使われるかを説明します。「セカンダリホストが複数存在する場合のデータフロー」に VVR のデータフローを示します。この設定では、2 つのセカンダリホストがあり、一方が非同期で、もう一方が同期レプリケーションを実行しています。

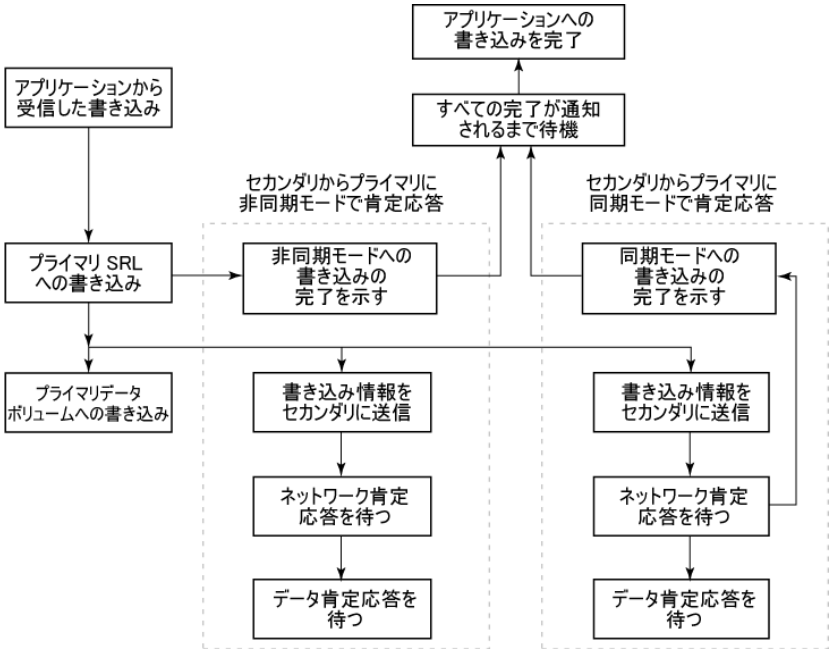


図 1-1 セカンダリホストが複数存在する場合のデータフロー

RVG（Replicated Volume Group）に属しているデータボリュームに書き込みが行われた場合、VVR がそのデータをプライマリ上のカーネルバッファにコピーします。VVR はバッファ上のデータにヘッダーを付けて、SRL に書き込みます。ヘッダーは、書き込みについての説明です。

VVR は、このカーネルバッファから、書き込み情報をすべてのセカンダリホストへ送信し、プライマリのデータボリュームに書き込みます。プライマリのデータボリュームへのデータの書き込みは非同期で実行されます。そのために、データボリュームへの書き込みに必要な時間は、プライマリのデータボリュームへの書き込み時間だけとなります。プライマリでデータボリュームへの書き込みが終了するまで、その書き込みに関するデータをカーネルバッファから解放することはできません。

同期モードでレプリケーションを行うすべてのセカンダリホストについて、VVR はまず書き込み情報をプライマリ SRL に送信します。次に、書き込み情報をセカンダリホストに送信して、情報を受信したことを示すネットワーク肯定応答を待機します。同期モードでレプリケーションを行うすべてのセカンダリホストから書き込み情報の確認通知を受信したら、書き込みの完了をアプリケーションに通知します。セカンダリの VVR カーネルメモリが書き込み情報を受信すると、セカンダリはただちにネットワーク肯定応答を送信します。アプリケーションはフルディスク書き込みを待機する必要がないので、パフォーマンスが向上します。データはその後、セカンダリデータボリュームに書き込まれます。セカンダリのデータボリュームに対する書き込みが完了すると、VVR はプライマリにデータ肯定応答を返信します。

非同期モードでレプリケーションを行うすべてのセカンダリホストについて、VVR はプライマリ SRL へ書き込まれた後に書き込みが完了したことをアプリケーションに通知します。したがって、書き込みの遅延時間とは、SRL に情報を書き込んでいる時間を意味します。次に、VVR は書き込み情報をセカンダリホストに送信します。セカンダリの VVR カーネルメモリが書き込み情報を受信すると、セカンダリはプライマリに対してただちにネットワーク肯定応答を送信します。セカンダリのデータボリュームに対する書き込みが完了すると、VVR はプライマリにデータ肯定応答を返信します。

アプリケーションは、VVR から通知を受信した後に完了した書き込みについては、データがプライマリ SRL に書き込まれたと解釈します。同期モードでレプリケーションを行うすべてのセカンダリホストについては、書き込みがカーネルバッファに受信されたと解釈します。ただし、VVR は、すべてのセカンダリホストからデータ肯定応答を受信するまで、書き込みをトレースします。セカンダリのデータボリュームへの書き込みが行われる前にセカンダリがクラッシュした場合、またはプライマリがデータ肯定応答を受信する前にクラッシュした場合は、SRL から書き込みを再生できます。

SRL からのリードバック時のデータフロー

非同期モードのセカンダリは、様々な原因（ネットワーク障害や、使用可能なネットワーク帯域幅を上回る書き込み処理の爆発的な増加など）によってレプリケーション処理が遅れ、プライマリのデータと比較した場合、データ更新が遅延する場合があります。セカンダリが遅延すると、セカンダリに送信するためのデータがプライマリの書き込みバッファ領域に保存されます。非同期モードのセカンダリがアプリケーションの書き込み速度に対応できない場合、VVR はプライマリカーネルバッファを解放して、受信した書き込みリクエストが遅延しないようにする必要があります。

このようにメモリから削除された書き込み情報については、プライマリの SRL から読み込み、セカンダリに送信されます。このような場合、書き込み情報は、前述のようにプライマリバッファから送信されるのではなく、リードバックバッファから送信されます。リードバック処理は、セカンダリのデータボリュームの状態がプライマリに追いつくまで続きます。追いついた時点で、セカンダリへの書き込み情報の送信処理は、SRL からのリードバックではなく、カーネルバッファからの送信に戻ります。

設定前の準備

VVR の設定を行う前に、レプリケーション対象のアプリケーションのデータ書き込み特性を理解しておく必要があります。また、VVR でシステムを構成しなければならない、ビジネス上の必要性も理解しておく必要があります。

ビジネスニーズの理解

ビジネスのニーズを満たすためには、次の事項を考慮する必要があります。

- 災害が発生した場合に損失する可能性があるデータの量と、そのような状況になったとしても、ビジネスを正常に継続するのに必要なデータ量
- ビジネスを正常に継続するのに、災害発生後のデータリカバリに許容できる時間

従来のテープによるバックアップ方式では、バックアップの頻度と分割されているテープの数によっては、災害によって失われるデータの量が膨大になる可能性があります。さらに、バックアップテープからのリカバリには、膨大な時間が必要になる場合もあります。VVR 環境の場合、リカバリに必要な時間はごくわずかであり、次の要因によって、失われるデータの量が決まります。

- レプリケーションのモード
- ネットワーク帯域幅
- プライマリとセカンダリ間のネットワーク遅延時間
- セカンダリデータボリュームの処理能力のために発生した書き込み遅延

セカンダリのデータを可能な限り最新の状態に維持する必要がある場合は、同期モードを使います。この場合は、アプリケーションがプライマリで実行する書き込みのピーク速度と同じ帯域を持つレプリケーション用ネットワークを用意することを推奨しています。一方、セカンダリのデータ更新が遅延することを許容できる場合は、非同期モードを使います。この場合は、アプリケーションがプライマリで実行する書き込みの平均速度と同じ帯域を持つレプリケーション用ネットワークを用意することを推奨しています。これらの事項は、ビジネスのニーズに合わせて決定します。

アプリケーションの特性の理解

RDS を設定する前に、レプリケーション対象のアプリケーションの書き込み速度など、運用時のデータスループットを事前に把握しておく必要があります。レプリケーションに関係するのは、書き込み操作のみです。読み込み操作は、レプリケーションに影響を与えません。後の項で説明する分析を実行するには、アプリケーションの書き込み速度のプロファイルが必要です。アプリケーションによっては一定頻度でデータの書き込みを行うものもあり、その場合比較的データの書き込み速度も一定であれば、プロファイルとして次のようなデータを算出することが可能です。

- アプリケーションの書き込みの平均速度
- アプリケーションの書き込みのピーク速度
- アプリケーションの書き込み速度がピークになる時間帯

アプリケーションによるデータ書き込みが一定の頻度で行われない場合は、特定の時間間隔でのデータの書き込み量を測定したデータが必要になります。アプリケーションのデータ書き込みの速度と使うディスクの容量の問題は、レプリケーション特有の問題でないため、ここでは扱いません。ここでは、すでにアプリケーションが導入済みであり、かつアプリケーションが必要とする書き込み速度を実現できるボリュームを Veritas Volume Manager（VxVM）で設定済みであることを前提としています。この場合、アプリケーションの書き込み速度の特性がすでに測定済みである場合もあります。

アプリケーションの特性が不明な場合は、アプリケーションを実際に実行して、レプリケーション対象となるすべてのボリュームへのデータ書き込みをツールを使って測定してください。アプリケーションによる書き込みが raw データボリュームではなくファイルシステムに対して行われる場合は、ファイルシステムによって書き込まれるメタデータも同様に測定する必要があります。これにより、レプリケーション対象のデータ総量が大幅に増加する可能性があります。たとえば、データベースがレプリケーション対象のボリュームに作成されたファイルシステムを使っている場合、`vxstat`（`vxstat(1M)` を参照）のようなツールを使うとボリュームに書き込まれたすべてのデータの量を正確に測定することが可能ですが、データベースを監視してそのリクエストを測定するツールでは、ファイルシステムによって書き込まれたデータは測定できません。

また、アプリケーションの書き込みのピーク速度と平均速度の両方を考慮することも重要です。これらの数値は、レプリケーション用ネットワークの種類を決定するのに必要です。同期モードでセカンダリホストとレプリケーションを行う場合は、アプリケーションの書き込みピーク速度と同じ帯域のネットワークをレプリケーションに使う必要があります。非同期モードでレプリケーションを行う場合には、アプリケーションの書き込み速度に合わせる必要がないため、アプリケーションの書き込みの平均速度と同じ帯域のネットワークをレプリケーションに使う必要があります。

最終的に測定が終了したら、書き込みのピーク速度と平均速度の値は測定期間内に得られた最高値に近い値とし、平均値や中間値は採用しないでください。たとえば、1 日のピーク速度と平均速度を 30 日間測定した場合、本来なら 30 日間で最高値のピーク速度と平均速度を採用するのですが、30 日分のデータを基に平均値を算出し、その値をピーク速度、平均速度としたとします。これらの値を基にネットワークの帯域を決定した場合は、その半分の期間でネットワーク帯域不足により、アプリケーションに対応できなくなります。したがって、特別な理由があって、測定された値が極端に高くなった場合を除いて、期間中に取得された最高値をピーク速度、平均速度に採用します。

レプリケーションのモードの選択

非同期モードと同期モードのどちらを使うかについては、各モードがアプリケーションとレプリケーションのパフォーマンスに与える影響を理解したうえで決定する必要があります。レプリケーションの基本的なプロセスを理解していれば、非同期モードまたは同期モードを使う場合の相対的な利点は明確になります。

非同期モードに関する特記事項

非同期モードでレプリケーションを行うと、セカンダリへのデータ送信をアプリケーションの書き込み完了後に行うことによって、各書き込みごとに発生するネットワークによる遅延の加算を回避します。このモードでの明らかな短所は、アプリケーションの書き込みが完了しても、実際にそのデータのレプリケーションが完了している保証がない点です。非同期モードを使うことによって内在する問題は、アプリケーションのスループットはほとんど影響を受けないが、レプリケーション全体のパフォーマンスは低下する可能性があるという点です。

非同期モードでは、ネットワーク帯域幅またはセカンダリの書き込みの遅延により、プライマリのカーネルメモリバッファが一杯になってしまいます。VVR のためのバッファにスペースを作り、レプリケーション処理を継続するためには、プライマリデータボリュームに書き込まれてはいるがセカンダリには送信されていない書き込み情報が格納されているメモリスペースを解放する必要があります。そのため、まだセカンダリには送信していないがプライマリのメモリから削除された書き込み情報を、VVR がセカンダリに送信するときには、プライマリの SRL から書き込み情報をリードバックすることになります。したがって、同

期モードの場合はデータが常にメモリ内にあるのに対して、非同期モードの場合は VVR が頻繁に SRL からデータのリードバックを行う必要がある場合もあります。その結果、リードバック操作が追加されるために、レプリケーション全体のパフォーマンスが低下する可能性があります。ネットワークやセカンダリの書き込みの遅延が発生しない場合、または書き込み情報が VVR のカーネルバッファに収まるぐらい少ない場合は、VVR は SRL からのリードバックを行いません。共有環境では、非同期モードでレプリケーションを行う場合、VVR は常に SRL からのリードバックを行います。

VVR と VxVM のカーネルバッファのサイズを調整する方法については、31 ページの「VVR バッファ領域」を参照してください。VVR による SRL のリードバックが頻繁に発生する場合は、複数のディスクで形成した（たとえば、平均書き込み速度の 10 倍の書き込み速度がある）ストライプボリューム上に SRL を作成すると、パフォーマンスが向上します。VVR が SRL からリードバックを行っているかどうかを判断するには、`vxstat` コマンドを使います。出力として、SRL の読み取り回数が表示されます。

同期モードに関する特記事項

同期モードは、アプリケーションによる書き込み処理が完了する前に、すべての書き込みに関する情報が必ずセカンダリに届いていることが保証されているという利点があります。ビジネス必要条件によっては、この特長が必須条件である場合もあります。その場合は、アプリケーションのパフォーマンスが、同期モードの選択の決定要因にはなりません。ただし、この項では、同期モードを選択した場合にパフォーマンスへ与える影響を説明します。

2 ページの「セカンダリホストが複数存在する場合のデータフロー」に示すように、まずすべての書き込みリクエストが SRL に書き込まれます。この書き込みが完了して初めて、データがセカンダリへ送信されます。ただし、同期モードでは、プライマリデータの SRL 書き込み完了前に、データをセカンダリに送信し、セカンダリとの応答確認を行うために、書き込みによる遅延時間は次のようになります。

`SRL` の書き込み遅延時間 + ネットワークを往復する伝達遅延時間

このように、同期モードでは各書き込みリクエストの遅延にネットワークを往復する伝達遅延時間が加算されるため、アプリケーションのパフォーマンスが大幅に低下します。

同期モードを選択する場合は、ネットワークに割り込みが発生した場合の VVR の動作を決定しておく必要があります。同期モードでは、`synchronous` 属性によって、プライマリとセカンダリのリンクが切れた場合に実行する処理を指定できます。`synchronous` 属性は、`override` または `fail` に設定できます。`synchronous` 属性を `override` に設定すると、ネットワークが一時的な機能不全になった場合、同期モードは非同期モードに切り替わります。この場合は、ネットワークの機能不全から回復し、セカンダリのデータボリュームがプライマリと同じ状態になったときに、レプリケーションは同期状態に戻ります。

synchronous 属性を fail に設定した場合には、セカンダリとのネットワーク接続が切断されている間に実行された、データボリュームへの書き込みに関するエラーが返されます。この結果、アプリケーションを使えなくなる可能性が高くなります。したがって、この設定を選択するのは、ネットワークが切断されることによってアプリケーションが使えなくなるよりも、プライマリとセカンダリが常に同じデータを保持していることが重要な場合に限られます。

synchronous 属性を override に設定するのは、大半のアプリケーションに適しているので、この設定をお勧めします。synchronous 属性を fail に設定するのは、プライマリとセカンダリのデータボリューム間で書き込みの差異がまったく許容されない特殊なアプリケーションにのみ適します。つまり、書き込み情報のレプリケーションがすぐにできない場合は、アプリケーションの書き込みが失敗するようにする場合にのみ、ハード同期を使ってください。ネットワークの停止がアプリケーションの停止につながる可能性があるため、アプリケーションの不要なダウンタイムの発生を回避する意味で、ハード同期を使っているホスト間のネットワークには高い信頼性が求められます。

synchronous 属性を fail に設定する場合のその他の特記事項

synchronous 属性を fail に設定した場合、VVR では、書き込みはセカンダリに到達しない限り正常に終了しません。RLINK が切断した場合には、書き込みは失敗し、SRL にもデータボリュームにも書き込まれません。ただし、セカンダリへの書き込み情報の送信中に RLINK が切断された場合は、アプリケーションに書き込みエラーが正しく返されたとしても、その書き込みが SRL に記録されてデータボリュームに適用される可能性があります。このような現象は、データボリュームの書き込みがレプリケーションのモードに関係なく非同期であるために発生します。操作順序について詳しくは、2 ページの「VVR のデータフロー」を参照してください。

この時点でプライマリで稼動しているアプリケーションの状態は、セカンダリをプライマリに昇格させてアプリケーションを起動した場合と同じです。ただし、プライマリデータボリュームとセカンダリデータボリュームの実際の内容は異なり、RLINK の切断後に発生した書き込みの分だけプライマリデータボリュームが新しくなっています。

同期 RLINK が接続されると、これらの書き込みはただちにセカンダリに到達し、プライマリとセカンダリのデータボリュームの内容は同じになります。また、データの一貫性が損なわれることは一切ありません。

この時点でアプリケーションが停止またはクラッシュして再起動が発生した場合は、更新済みのデータボリュームの内容に基づいてリカバリが実行されます。プライマリ上のアプリケーションの動作は、RLINK が切断されたままである間は、セカンダリをプライマリに昇格させてアプリケーションを起動したときの動作と異なる場合があります。

データベースアプリケーションの場合、これらの書き込みがトランザクションをコミットするための書き込みである可能性もあります。プライマリのデータボリュームを使ってアプリケーションがリカバリを試みる場合、トランザクション

のコミットがすでにデータボリュームに存在するため、アプリケーションはトランザクションをロールフォワードします。ただし、アプリケーションのリカバリが、セカンダリからプライマリに昇格した RDS 上で実行される場合には、トランザクションはロールバックされます。

このケースは、アプリケーションがディスクに対して直接書き込みを行い、書き込みの一部が完了した時点でディスクに障害が発生した場合と同じです。書き込みの一部は物理的にはディスクに到達しますが、アプリケーションには書き込み全体について書き込みエラーが返されます。ディスクに到達した書き込みが、アプリケーションでトランザクションのロールバックとロールフォワードのどちらを行うかを決定するのに役立つ部分である場合は、失敗したトランザクションもリカバリ時に正常に実行されます。

また、アプリケーションによって書き込みが開始された後に RLINK が切断され、次の書き込みが開始される前に再度接続されるケースも考えられます。この場合には、アプリケーションは最初の書き込みについてはエラーが返されますが、2 番目の書き込みは成功します。

ファイルシステムやデータベースなどの他のアプリケーションでは、異なる方法でこうした切断のエラーに対処します。Veritas File System では、ファイルまたはファイルシステムを無効にすることなくエラー処理を行います。

synchronous 属性を fail に設定した場合、RLINK が切断した場合には、アプリケーションの書き込みは失敗します。自動同期または再同期は、SRL を完全に排出するために RLINK を切断する必要があります。アプリケーションエラーを回避するには次の点に注意してください。

- テイクオーバー後フェールバックを行うときには、DCM 再生が完了するまで、プライマリ上でアプリケーションを起動しないでください。または、DCM 再生が完了するまで、レプリケーションモードを一時的に非同期モードに変更してください。
- 自動同期または DCM 再生を使ってセカンダリを同期するときには、同期が完了するまで、レプリケーションモードを一時的に非同期モードに変更してください。

非同期レプリケーションと同期レプリケーション

同期または非同期のどちらのレプリケーションを使うかは、ビジネス上の必要条件と、ネットワークの機能に応じて決まります。主な考慮事項を次の表にまとめます。

メモ : 複数のセカンダリが存在する場合は、レプリケーションに非同期モードを使うホストと同期モードを使うホストを混在させることができます。詳しくは、『Veritas Volume Replicator 管理者ガイド』の「複数のセカンダリホストを含む RDS でのデータフロー」を参照してください。

考慮事項	
同期モード	**非同期モード**
セカンダリが最新の状態である必要性	
セカンダリが常に最新の状態になります。 synchronous 属性を override に設定した場合、ネットワークの機能不全が発生しない限り、セカンダリは最新の状態になります。	セカンダリには、ある時点でのプライマリの状態が反映されます。ただし、セカンダリは最新の状態とは限りません。プライマリには、セカンダリに書き込まれていない、コミットされたトランザクションが存在する場合があります。
データ遅延の管理の必要条件	
少量の書き込みに最も適しています。 遅延保護は必要ありません（セカンダリが常に最新の状態であるため）。	セカンダリでデータ遅延が発生することがあります。プライマリに災害が発生した場合に、コミットされたトランザクションが失われてもよいかどうか、また、失われてもよい場合はどの程度まで許容できるのかを考慮する必要があります。 VVR では、未送信の書き込み情報をいくつまで許可し、その制限を超えた場合にどのような処理を実行するかを指定することにより、遅延保護を管理することができます。

考慮事項	
同期モード	**非同期モード**
ネットワークの特性 : 帯域幅、遅延、信頼性	
帯域幅が広く遅延が小さい状況に最も適しています。ネットワークが遅延すると、アプリケーションがその影響を受ける可能性があります。 ネットワーク帯域幅は、常にアプリケーションの書き込み速度と同等か、それを上回る必要があります。	SRL を使って、ネットワークの IO の急増または混雑に対処します。これにより、ネットワーク帯域幅の変動によるアプリケーションのパフォーマンスへの影響を最小限に抑えることができます。 ネットワークには、アプリケーションの平均的な書き込み速度に常に対応できる帯域幅が必要です。ネットワークの帯域不足を非同期レプリケーションで補うことはできません。
アプリケーションのパフォーマンスのための必要条件（遅延など）	
セカンダリからのネットワーク肯定応答を受信するまでは I/O が完了しないため、アプリケーションのパフォーマンスに多大な影響を与える可能性があります。	セカンダリからのネットワーク肯定応答を待たずに I/O が完了するため、アプリケーションのパフォーマンスへの影響を最小限に抑えることができます。

遅延保護および SRL 保護の選択

レプリケーションパラメータ latencyprot および srlprot を使うと、同期と非同期の特性を融合させることができます。これらのパラメータにより、セカンダリに対して、容認できる遅延の範囲を指定することができます。

latencyprot を有効にした場合、セカンダリはあらかじめ定義されたリクエスト数（最高遅延水準点）までのみ遅延が容認されます。このユーザー定義の最高遅延水準点に到達すると、プライマリでの書き込みにウェイトが付加されます（スロットル）。この処理によって、セカンダリへの未送信書き込みリクエスト数が、あらかじめ定義されたもう 1 つのリクエスト数（最低遅延水準点）以下に減少するまで、プライマリでの書き込みは、ウェイトがかかったまま処理することになります。したがって、アプリケーション側から見た書き込みの平均遅延は増大します。最高遅延水準点と最低遅延水準点の数値差が大きいと、書き込みが抑えられているために、プライマリでの書き込み処理が遅延し、最低水準点まで減少するまでの間アプリケーションが停止したように見えることがあります。最高水位点と最低水位点の数値差が小さい場合、プライマリの書き込み遅延は各リクエストに均等に分散されるため、数値差が大きい場合に比べ短時間の遅延が頻繁に発生します。頻繁に書き込みにウェイトが追加されます。しかし、数値差を大きくした場合よりも、各書き込みにおけるウェイトが追加されている時間が短くなります。そのため多くのケースでは、その数値差を小さくして、短時間の書き込み抑制を多く発生させた方が良好なレプリケーション環境となります。

パラメータ latencyprot を効果的に使って、必要な RPO（Recovery Point Objective）を達成することができます。パラメータ latencyprot の設定前に、遅延高水準点や遅延低水準点の値に影響を与える要因を考えておくと便利です。

- 書き込み RPO
- 平均書き込み速度
- 使用可能な平均ネットワーク帯域幅
- 平均書き込みサイズ
- 遅延高水準点から遅延低水準点への情報の排出に、SRL が必要とする最大時間。この時間は、最も繊細なアプリケーション、つまり RVG のすべての使用ボリュームの中で「最小の」タイムアウト値を持つアプリケーションのタイムアウト値です。
- SRL に記録済みの書き込み回数

許容 RPO 値には、特定の要件に基づき、書き込み回数に関するユーザー定義の遅延高水準点を設定します。このとき、遅延高水準点には、書き込み RPO を平均書き込みサイズで割った値を設定します。

遅延低水準点には、書き込みが保留となってもアプリケーションに影響のない値を設定します。平均ネットワーク速度が平均書き込み速度以上とすると、SRL 排出速度は次の式で求められます。

平均ネットワーク速度 - 平均書き込み速度 この値を算出したら、遅延低水準点を以下の計算式で求めます。

遅延高水準点 – （効果的な SRL 排出速度 × 最小タイムアウト）/ 平均書き込みサイズ

レプリケーションパラメータ srlprot は SRL のオーバーフローの回避に利用でき、latencyprot と同様の機能があります。ただし、srlprot 属性はデフォルトで autodcm に設定されるため、SRL はオーバーフローする可能性もあり、オーバーフローした場合には、dcm_logging モードへ変わります。

dcm_logging モードになった場合、プライマリの書き込みパフォーマンスには影響がありませんが、プライマリとセカンダリのデータボリュームの整合性が失われるために、データボリュームの再同期を行う必要があります。詳しくは、『Veritas Volume Replicator 管理者ガイド』を参照してください。

ネットワークのプランニング

この項では、VVR でのレプリケーションに使えるネットワークプロトコルについて説明します。また、レプリケーションのモード（同期または非同期）によって、必要なネットワーク帯域幅についても説明します。

ネットワーク帯域幅の選択

ネットワーク回線の種類による利用可能な帯域幅

2 つの地点を結ぶのに使うネットワーク回線の種類によって、利用可能な最大帯域幅が異なります。たとえば T3 回線の場合は 45 Mbps です。ただし、考慮しなければならない重要な要素は、その接続回線を他のアプリケーションと共有して使うのか、それともレプリケーション専用として使うのか、という点です。他のアプリケーションと回線を共有する場合は、それらのアプリケーションが必要とする帯域幅を確認し、ネットワーク回線の最大帯域幅からその分を差し引いた分だけが、レプリケーションで使える帯域幅と考えてください。また、回線を共有しているアプリケーションが時間帯によってネットワーク使用量が異なる場合には、使用量がピークとなる時間帯が、VVR によるネットワーク使用のピークと重なる可能性があるかどうかを考慮することも必要です。さらに、VVR および下位のネットワークプロトコルによる、オーバーヘッドが生じるため、使える帯域幅がわずかに（通常は 3 - 5 %）減少します。

レプリケーションモードの違いによるネットワーク帯域幅への必要条件

レプリケートされるすべての書き込みリクエストは、最終的にネットワークを経由してセカンダリに送信されます。利用するレプリケーションモードによって、プライマリとセカンダリ間のデータ転送にクリティカルパスが必要であるかどうかが決定します。

同期モードでレプリケーションを行う場合は、プライマリノードでアプリケーションに書き込み完了を知らせる前に、書き込みに関する情報がセカンダリに到達している必要があるため、プライマリとセカンダリはクリティカルパスのネットワークを形成する必要があります。そのため、アプリケーションの書き込みが、ネットワークの帯域幅を上回る場合、セカンダリへのデータ転送が完了するまで、次の書き込みが行われないために、プライマリでの書き込み遅延が増大することになります。

一方、非同期モードでレプリケーションを行う場合にはプライマリで書き込み完了を知らせる前にセカンダリへ書き込み情報の送信が完了している必要がないため、ネットワークの帯域が不足してもプライマリで書き込みの遅延が増えることはありません。その代わりに、送信できなかった書き込みリクエストは SRL に蓄積されます。ネットワーク帯域の不足が慢性的になると、その結果として SRL はオーバーフローしてしまいます。とはいえ、プライマリでの書き込み量

がピークを過ぎた後で、SRL に蓄積されていたすべての情報をセカンダリに送るだけの能力をネットワークが有している場合には、書き込みが集中するピークの時間帯においては、SRL を一時的にネットワークの帯域不足を補うバッファとして使うことが可能です。この機能を前提にシステムを構築した場合はセカンダリサイトが常に最新の状態ではない、ということに注意してください。

いくつかのパラメータを設定し、プライマリとセカンダリ間のネットワークをクリティカルパスにすることで前述のような状態を避けることが可能です。`latencyprot` や `srlprot` の機能を有効にすると、その効果を得られます。これらの機能については、12 ページの「遅延保護および SRL 保護の選択」で詳細に説明しています。

ネットワークの帯域不足によって発生する問題を回避するには次の原則を適用します。

- 同期モードを使っている場合、アプリケーションの書き込みのピーク速度以上のネットワーク帯域幅が必要です。そうでない場合は、ネットワーク遅延により書き込み速度が抑制され、書き込みの遅延が増加します。ただし、レプリケーションによるネットワーク回線使用がピークに達している時間以外では、回線に余裕が生じるため、21 ページの「ピーク時の制約」で説明しているチェックポイントを使って新規ボリュームの同期を行うことも可能です。
- 非同期モードを使っており、プライマリの書き込みがピークのときにはセカンダリのデータ更新が遅延してもかまわない場合は、アプリケーションの書き込みの平均速度と同じだけのネットワーク帯域幅があれば十分です。この場合、同期により生成される追加データを処理できるだけのネットワーク帯域が存在しないために、同期を行う場合にはアプリケーションをシャットダウンする必要があります。
- 非同期モードを使っており、セカンダリの更新の遅延増加を防ぐために `latencyprot` を設定している場合は、SRL に溜めておく未処理件数の数、つまり最高遅延水準点として設定した値の大きさによって、ネットワーク帯域の必要条件が異なってきます。最高遅延水準点が小さい場合、レプリケーションは同期モードと同様になり、アプリケーションの書き込みのピーク速度にほぼ等しいネットワーク帯域幅が必要です。最高遅延水準点が大きい場合、セカンダリは数時間遅れる場合があります。この場合、帯域幅をアプリケーションの平均書き込み速度に対応させます。ただし、RPO に達するとは限りません。

ネットワークプロトコルの選択

VVR では、プライマリとセカンダリ間で 2 種類のメッセージ（ハートビートメッセージとデータメッセージ）をやりとりします。ハートビートメッセージは、UDP 転送プロトコルを使って送信されます。VVR では、データメッセージのやりとりに TCP 転送プロトコルまたは UDP 転送プロトコルを選択可能です。

データメッセージに使うプロトコルは、ネットワークの特性に基づいて選択します。TCP は、パケットを失うようなことがあるようなネットワークにおいても、UDP と比較してより適切に動作すると言えます。ただし、特定のネットワーク環境で適切な動作をするプロトコルを判別するには、両方のプロトコルを試してみる必要があります。

TCP プロトコルを使う場合、VVR は利用可能な帯域幅を使うために、必要に応じて複数の接続を作成します。これは基本的に、異常なパケットが多数発生する場合に役立ちます。

メモ : プライマリとセカンダリでは、同じプロトコルを指定する必要があります。プロトコルが異なると、ノード間通信が行えず、RLINK が接続されません。これは、クラスタ環境内のノードにおいても同様です。

VVR は、デフォルトでは UDP 転送プロトコルを使います。ネットワーク転送プロトコルの設定については、『Veritas Volume Replicator 管理者ガイド』を参照してください。

VVR で使うネットワークポートの選択

VVR では、プライマリとセカンダリ間の通信に、UDP 転送プロトコルおよび TCP 転送プロトコルを使います。この項では VVR で使うデフォルトポートを示します。

UDP を使ってデータをレプリケートする場合、VVR はデフォルトで次のポートを使います。

ポート番号	説明
UDP 4145	IANA 認証ポート。プライマリとセカンダリ間のハートビート通信に使います。
TCP 8199	IANA 認証ポート。プライマリ上の `vradmind` デーモンとセカンダリ間の通信に使います。
TCP 8989	差分同期を行う場合に `in.vxrsyncd` デーモン間の通信で使います。
UDP Anonymous ポート (OS に依存)	プライマリとセカンダリ間でのデータレプリケーションのため、各プライマリ - セカンダリの通信で使うポート。データポートは、各ホストに 1 つずつ必要です。

TCP を使ってデータをレプリケートする場合、VVR はデフォルトで次のポートを使います。

ポート番号	説明
UDP 4145	IANA 認証ポート。プライマリとセカンダリ間のハートビート通信に使います。
TCP 4145	TCP リスナーポート用の IANA 認証ポート。
TCP 8199	IANA 認証ポート。プライマリ上の vradmind デーモンとセカンダリ間の通信に使います。
TCP 8989	差分同期を行う場合に in.vxrsyncd デーモン間の通信で使います。
TCP Anonymous ポート	プライマリとセカンダリ間でのデータレプリケーションのため、各プライマリ - セカンダリの通信で使うポート。データポートは、各ホストに 1 つずつ必要です。

vrport コマンドを使うと、VVR で使っているポート番号を参照および変更できます。操作方法については、『Veritas Volume Replicator 管理者ガイド』を参照してください。

ファイアウォール環境での VVR の設定

この項では、ファイアウォール環境で動作する VVR の設定方法について説明します。VVR で使うデフォルトのポート番号については、16 ページの「VVR で使うネットワークポートの選択」を参照してください。ネットワークアドレス変換（NAT: Network Address Translation）ベースのファイアウォールを介したレプリケーションについて詳しくは、40 ページの「VVR とネットワークアドレス変換ファイアウォール」を参照してください。

TCP 使用時のファイアウォール環境での VVR の設定方法

ファイアウォール環境で、ハートビートに使うポート、vradmind デーモンで使うポートおよび in.vxrsyncd デーモンで使うポートを有効にします。ポートの情報の表示や、VVR で使うポートの変更には、vrport コマンドを使います。

UDP 使用時のファイアウォール環境での VVR の設定方法

1 ファイアウォール環境で次のポートを有効にします。

- ハートビートに使うポート
- vradmind デーモンで使うポート
- in.vxrsyncd デーモンで使うポート

ポートの情報の表示や、VVR で使うポートの変更には、vrport コマンドを使います。

2 プライマリとセカンダリ間でデータのレプリケーションに使うポートを限定します。デフォルトでは、オペレーティングシステムによって、匿名ポート番号が割り当てられます。大半のオペレーティングシステムは、32768 - 65535 の間で Anonymous ポート番号を割り当てます。データポートは、各プライマリ - セカンダリ接続について 1 つずつ必要です。VVR で使うポートのリストまたは範囲を指定するには、vrport コマンドを使います。

3 手順 2 で設定したポートをファイアウォール環境で有効にします。

パケットサイズの選択

レプリケーションに UDP 転送プロトコルを使う場合、VVR でホスト間の通信に使う UDP パケットのサイズは、レプリケーションのパフォーマンスにおいて重要な要素となります。VVR で使う UDP パケットのサイズは、デフォルトで 8400 バイトです。断片化した IP パケットをサポートしていないなどの特定のネットワーク環境では、パケットサイズを小さくする必要がある場合があります。

現在のネットワークで多くのパケットが失われる場合、レプリケーションに利用できる帯域幅は減少します。RLINK 上で vxrlink stats を実行し、タイムアウトエラーが多数発生していたら、この状況が発生していることを示します。

この場合、パケットサイズを縮小すると、ネットワークパフォーマンスが改善することがあります。ネットワークで多くのパケットが失われれば、大きなパケットが失われるたびに、大規模な再転送を行う必要が生じるということになります。この場合、問題が改善するまでパケットサイズを縮小してみてください。

IPSEC や VPN ハードウェアなどの一部のネットワーク要素がパケットにデータを追加している場合は、パケットサイズを減らして、パケットに追加するバイトのための空き領域を確保し、MTU を超えないようにします。そうしない場合、各パケットは 2 つに分割されます。

VVR の packet_size 属性の変更方法について詳しくは『Veritas Volume Replicator 管理者ガイド』を参照してください。

ネットワークの最大転送単位の選択

VVR によって転送される UDP パケットまたは TCP パケットのうち、サイズがネットワークの最大転送単位（MTU）より大きいパケットは、オペレーティングシステムの IP モジュールによって MTU サイズの IP パケットに分割されます。IP 断片化をサポートせず、現在のネットワークデバイスよりも MTU が小さいルーターをパケットが通過するために、ネットワークでの損失が発生することがあります。この場合、ネットワークで最も小さい MTU を持つルーターの MTU サイズと同じサイズに MTU を設定します。

SRL のサイズ設定

SRL のサイズは、レプリケーションのパフォーマンス決定の重要な要素です。この項では、SRL のサイズの決定に関する特記事項を説明します。Volume Replicator Advisor（VRAdvisor）ツールを使って適切な SRL サイズを決定する方法については、『Veritas Volume ReplicatorAdvisor ユーザーズガイド』も参照してください。

特定のセカンダリに対して SRL がオーバーフローした場合、そのセカンダリと接続している RLINK の状態は、STALE となり、プライマリと完全同期を実行するまでこの状態が続きます。再同期は時間がかかるプロセスであり、処理中はセカンダリ上のデータを使えなくなるため、SRL のオーバーフローを回避することが重要になります。SRL のサイズは、次の 4 つの制約を十分に満たす大きさに設定する必要があります。

- プライマリのアプリケーション書き込み速度が RLINK を介したレプリケーション速度を上回る可能性がある場合は、書き込み速度がピークの期間に SRL のオーバーフローが発生しないこと
- セカンダリ RVG の同期中にオーバーフローが発生しないこと
- セカンダリ RVG のリストア中にオーバーフローが発生しないこと
- ネットワークまたはセカンダリノードの停止時間が計画していた時間よりも長くなったとしても、オーバーフローが発生しないこと

メモ : SRL のサイズは最低でも 110 MB 以上必要です。SRL のサイズが 110 MB よりも小さい場合は、110 MB より大きいボリュームを SRL に設定することを促すエラーメッセージが出力されます。

SRL のサイズを決定するには、これらの各制約を満たすサイズを個別に決定する必要があります。さらに、すべての制約を満たすように、算出した最大値以上の値を選択します。この分析の実行には、次の情報が必要です。

- セカンダリノードで想定されるダウンタイムの最大値
- ネットワークに想定されるダウンタイムの最大値
- プライマリデータボリュームとセカンダリデータボリュームを同期する方法
  同期の実行時に、プライマリでアプリケーションを停止する場合、SRL が使われないため、上記制約の同期実行中にオーバーフローしないことは、SRL のサイズ決定の要因にはなりません。それ以外の場合、この情報にはネットワーク経由でのデータをコピーするための所要時間、またはテープやディスクへのデータのコピーに要する時間、セカンダリサイトへのコピーの送信に要する時間、セカンダリデータボリュームへのデータのロードに要する時間が含まれる可能性があります。

メモ : 自動同期オプションを使ってセカンダリの同期を行う場合は、このパラグラフの内容は関係ありません。

セカンダリデータボリュームに障害が発生したとしても、完全同期を行う必要がないようセカンダリでバックアップを実行する場合は、次の情報も必要になります。

- セカンダリのバックアップスケジュール
- 障害が発生したセカンダリデータボリュームの検出および修復に必要な必要な時間の最大値
- 修復したセカンダリデータボリュームへバックアップデータをリストアするのに必要な時間

ピーク時の制約

設定によっては、レプリケーションがアプリケーションからの書き込みに追いつけずに遅延したり、その遅延分（SRL に蓄積されていた書き込み情報）がすべてセカンダリに送信されたり、という状況が頻繁に発生する可能性があります。たとえば、アプリケーションの書き込みのピーク速度が RLINK で使っているネットワークの帯域幅を上回っている場合は、日中の業務時間中にすべてのデータを転送することができず夜中に遅延分を解消することもあります。当然、同期 RLINK の場合は、このような状況は発生しません。ネットワーク帯域の不足によってボリュームへの書き込みが抑制されアプリケーションの各書き込みで待ち時間が発生してしまい、アプリケーションの動作が遅くなりレプリケーションと同期した進行が行われるためです。

非同期 RLINK の場合、セカンダリへ未送信のプライマリでの書き込み情報件数はプライマリの SRL のサイズによって決定します。アプリケーションの書き込みのピーク速度がレプリケーションで使うネットワークの帯域幅を上回ることが判明している場合は、SRL のサイズを決定する際にこの項の説明をよく考慮してください。

ここで、ある間隔ごとに連続して書き込みを行うアプリケーションを例に、必要な SRL のサイズを計算してみます。

1. 各時間内のネットワークで転送可能なデータ量（BW_N）を算出します。
2. 各時間間隔 *n* における SRL ボリュームの使用状況（LU_n）は、ネットワーク帯域幅（BW_N）とアプリケーションの書き込み（BW_{AP}）の差分（$LU_n = BW_{AP(n)} - BW_N$）から算出します。

メモ : 共有環境では、クラスタに存在するすべてのノードの書き込み速度を考慮する必要があります。アプリケーションの書き込み速度（BW_{AP}）は、各ノードの書き込み速度の合計を反映している必要があります。

3 各時間間隔の SRL の使用状況を算出し、すべての LGn を足し合わせることで、SRL のログサイズ（LS）を算出できます。

$$LS_n = \sum_{i=1...n} LU_i$$

算出した LS_n の最大の値をピーク時の SRL サイズとして使います。この計算例の結果は、22 ページの「表 1. ピーク時に必要な SRL サイズの計算例」を参照してください。3 列目のアプリケーションは、5 ページの「アプリケーションの特性の理解」で説明しているように、アプリケーションによる書き込みを測定し、その測定データより 1 時間毎の書き込み速度の概算を示しています。4 列目のネットワークはネットワークの帯域幅を示しています。5 列目の SRL 使用状況は各時間におけるアプリケーションの書き込み速度とネットワーク帯域の差分を示しています。6 列目の SRL サイズの累積値は、1 時間毎の SRL 増加の累積値を示しています。6 列目の最大値は、37 GB です。このアプリケーションに対する SRL のサイズはこのサイズ以上にする必要があります。

ピーク時の SRL の最大サイズは、設定または構成によって抑えることが可能です。例を次に示します。

- `latencyprot` の特性を有効にすると、RLINK での未送信の書き込み情報の件数を制限し、アプリケーションの書き込み速度を低く抑えることが可能です。
- ネットワークの帯域幅を拡大すると未送信の書き込み情報の件数が少なくなり、必要な SRL のサイズが小さくなります。この例では、アプリケーションの書き込み速度の最大値が 15 GB/ 時 であるため、ネットワークの帯域幅も同程度あれば各時間間隔における SRL の増加が 0 となります。

表 1-1　　表 1. ピーク時に必要な SRL サイズの計算例

開始時刻	終了時刻	アプリケーション（GB/ 時）	ネットワーク（GB/ 時）	SRL の使用状況（GB）	SRL サイズの累積値（GB）
7 a.m.	8 a.m.	6	5	1	1
8	9	10	5	5	6
9	10	15	5	10	16
10	11	15	5	10	26
11	12 p.m.	10	5	5	31
12 p.m.	1	2	5	-3	28
1	2	6	5	1	29

表 1-1　表 1. ピーク時に必要な SRL サイズの計算例

開始時刻	終了時刻	アプリケーション（GB/ 時）	ネットワーク（GB/ 時）	SRL の使用状況（GB）	SRL サイズの累積値（GB）
2	3	8	5	3	32
3	4	8	5	3	35
4	5	7	5	2	37
5	6	3	5	-2	35

メモ : 共有環境では、アプリケーション列の値にすべてのノードの書き込み速度が含まれている必要があります。たとえば、1 時間当たりでは、`seattle1` の書き込み速度は 4 GB、`seattle2` の書き込み速度は 2 GB なので、アプリケーションの書き込み速度は 6 GB/ 時になります。

同期の実行時の制約

RDS に新規でセカンダリを追加した場合は、そのセカンダリのデータボリュームは初期化された状態です。そのため、プライマリでアプリケーションを起動していないためにデータボリュームにデータが存在しない場合を除いて、プライマリとセカンダリのデータボリュームを同期する必要があります。SRL がオーバーフローしたときやレプリケーションをいったん停止した後で再開した場合や、セカンダリデータボリュームに障害が発生した場合も、セカンダリの同期を行う必要があります。

この項で説明する制限は、同期方法として自動同期を使わない場合の制限です。なお、この項での制限は、自動同期以外の方法を使っている場合でも、セカンダリと同期中にプライマリでアプリケーションを停止するときには適用されません。ただし多くの場合、アプリケーションをプライマリ上で実行しながら、プライマリとセカンダリのデータボリュームを同期する必要があります。この操作は『Veritas Volume Replicator 管理者ガイド』に説明されている方法のいずれかを使って実行します。

同期実行中もアプリケーションは動作し続け、書き込み情報は SRL に累積されます。同期中に SRL がオーバーフローした場合は、同期プロセスを再実行する必要があります。したがって、同期中に SRL がオーバーフローしないことが必要であるため、同期中の SRL に書き込まれるアプリケーションの書き込み情報量が SRL のサイズを超えないことが絶対条件です。同期完了後にレプリケーションが開始すると、SRL 上の情報はセカンダリに送信され、最終的にはセカンダリの遅延が解消されます。

可能ならば、アプリケーションの書き込みが少ない時間帯に同期を実行するようにスケジュールを組む必要があります。アプリケーションの書き込みが少ない時間帯に同期プロセスを完了することが可能な場合は、この期間中に受信するすべての書き込みを格納できるだけのサイズに SRL が設定されていることを確認する必要があります。サイズ設定が適切でないと、SRL がオーバーフローする場合があります。最適な SRL サイズを特定する方法について詳しくは、『Veritas Volume ReplicatorAdviser ユーザーズガイド』を参照してください。ただし、アプリケーションの書き込みが増加した場合、同期が進行中であっても SRL サイズの変更が必要な場合があります。SRL サイズの変更については、『Veritas Volume Replicator 管理者ガイド』の「SRL のサイズ変更」の項を参照してください。アプリケーションの書き込みが少ない時間帯に同期プロセスを完了できない場合は、平均値、または安全を期してピーク値を使うように SRL のサイズを設定します。詳しくは、5 ページの「アプリケーションの特性の理解」の項を参照してください。

セカンダリのバックアップ実行時の制約

VVR は、セカンダリデータボリュームを周期的にバックアップするための機能を提供しています。23 ページの「同期の実行時の制約」で説明したように、完全同期を実行しなければ解決できない問題もあります。その場合、セカンダリのバックアップが使用可能であれば、ネットワークを介した完全同期を行うより速く、セカンダリをオンラインにすることが可能です。

セカンダリのバックアップを行うには、セカンダリのチェックポイントを作成し、セカンダリのすべてのデータボリュームについて raw レベルのコピーを作成します。障害が発生したら、セカンダリデータボリュームをこのローカルコピーからリストアし、チェックポイントからレプリケーションを実行すると、セカンダリのデータを最新の状態にするための SRL のログ再生がチェックポイントから行われ、最新の状態にするための時間を大幅に節約できます。

この処理の制約は、SRL の容量です。つまり、バックアップを作成したチェックポイント以降にプライマリで行われたアプリケーションによる書き込み情報のすべてを、蓄えられるだけの容量が SRL に必要です。この制約は、次の 2 つの要因に大きく関係します。

- アプリケーションの書き込み速度
- セカンダリのバックアップのスケジュール

アプリケーションの書き込み速度とセカンダリのバックアップのスケジュールから、SRL の最小サイズを算出できます。実際には、これらの値を使って算出した計算値にマージンを加算し、次のような他の要因によりレプリケーションが行えない間の書き込み情報を SRL に蓄積できるようにします。

- システム管理者がデータボリューム障害を検出するまでに必要な時間の最大値
- 障害の発生したドライブを修復または交換するのに必要な時間の最大値

■ バックアップテープのデータをディスクにリストアするのに必要な時間

このような制約を満たすために必要な SRL のサイズを算出するには、まずセカンダリのバックアップの間隔と前述の要因から算出される時間を加算し、SRL に情報を格納しなければならない時間を算出します。次に、アプリケーションの書き込み速度のデータを使って、アプリケーションがこの時間内に生成する可能性がある書き込み情報の最大データ量を算出します。

メモ: 1 つのボリュームのみに障害が発生した場合でも、すべてのボリュームをリストアする必要があります。

セカンダリのダウンタイムによる制約

セカンダリノードとのネットワーク接続またはセカンダリノード自体がダウンすると、プライマリノード上の RLINK がネットワークの切断を検出し、使っているレプリケーションのモードに応じた対応を行います。`synchronous` 属性が `fail` に設定された RLINK の場合、接続がリストアするまで、ネットワーク切断から後のすべての書き込みリクエストを失敗させます。そのため、SRL への書き込みは発生しないため、SRL のサイズによるダウンタイムの制限はありません。ハード同期モード以外の RLINK の場合は、接続がリストアするまで、プライマリの書き込みリクエストがすべて SRL に蓄積されます。そのため、SRL のサイズは、想定される最大ダウンタイムの間にアプリケーションが生成する可能性がある最大出力量の書き込み情報を格納できるだけのサイズが必要になります。

ダウンタイムの最大時間の予測が困難な場合もあります。ハードウェアやネットワーク接続の障害に対し、規定の時間内で修理が完了することをベンダーが保証する場合があります。当然、保証されている時間内に修理が完了しない場合は、SRL のオーバーフローが発生する可能性があるため、SRL のサイズを決める際には、安全のためにマージンを加えておくことを推奨しています。

このような制約を満たすために必要な SRL サイズの予想値を計算するには、まずセカンダリノードおよびネットワーク接続で発生する最大ダウンタイムとして妥当と考えられる値を算出します。次に、アプリケーションの書き込み速度のデータを使って、アプリケーションがこの時間内に生成する可能性がある書き込み情報の最大データ量を算出します。SRL オーバーフロー保護として `autodcm` モードを有効にした場合、SRL がオーバーフローしたとしても、DCM に変更分が記録されるため、SRL の容量によるダウンタイムの許容時間の制限は厳格でなくなります。ただし、DCM 再生によるセカンダリの同期を行う場合には、DCM 再生中はセカンダリのデータボリュームは整合性を失う状態になるため、不測の事態に対処できるだけの容量を SRL に割り当てることは重要であることに注意してください。

その他の要因

前述の各制約を満たす SRL サイズの計算が終了したら、さらにいくつかの要因を考慮する必要があります。

同期を実行時、およびセカンダリのダウンタイム、バックアップ実行時の制約に相当する状況が発生した直後に、アプリケーションの書き込み速度がピークに達する可能性もあります。その場合、ネットワークがレプリケーションによる情報転送と SRL に蓄積された情報のデータ転送の両方を実行するだけの帯域を有していない場合には、セカンダリへの情報転送がさらに遅延することになります。その結果、他の制約によって算出した SRL の最大サイズに、ピーク使用時の制約から算出したサイズを加える必要がある場合も考えられます。これは、ピーク時の制約を通常は適用する必要がない同期 RLINK にも適用されます。同期 RLINK も、ネットワーク切断後は SRL に記録されている情報を転送し終わるまで非同期 RLINK として機能するためです。

当然、別の状況が発生し、さらに制約が必要になる可能性もあります。たとえば、同期が完了した直後に長時間のネットワーク障害が発生したり、ネットワーク障害の後にセカンダリノードで障害が発生した場合などです。発生確率が低い障害に対して対応するかどうか、またその障害発生時にどの程度の対応時間が必要か考慮する場合は、SRL のオーバーフローが起きたときにそれを解消するために発生するシステムのダウンタイムのコストと、ストレージを増設するためのコストを比較する必要があります。

SRL に書き込まれるすべてのデータにはヘッダー情報も含まれるため、SRL のサイズを算出した後に、もう 1 度そのサイズを調整する必要があります。調整時には、書き込みリクエストの一般的なサイズを考慮する必要があります。各要求にはヘッダー情報用に別のディスクブロック（512 バイト）が少なくとも 1 つ使われるため、調整は次のようになります。

平均書き込みサイズ	ヘッダー情報として SRL のサイズ算出に追加が必要な割合（%）
512 バイト	100%
1 KB	50%
2 KB	25%
4 KB	15%
8 KB	7%
10 KB	5%
16 KB	4%
32 KB 以上	2%

例

この項では、VVR 設定における SRL サイズの算出例を示します。最初に、サイトの情報を収集します。この例で使うサイトの情報は、次のとおりです。

アプリケーションの書き込みのピーク速度	1 GB/ 時
ピーク時間	午前 8 時 - 午後 8 時
オフピーク時のアプリケーションの書き込み速度	250 MB/ 時
平均書き込みサイズ	2 KB
セカンダリサイトの数	1
RLINK のタイプ	synchronous=override
同期の所要時間 :	
アプリケーションの停止	なし
テープへのデータのコピー	3 時間
セカンダリサイトへのテープの輸送	4 時間
データのロード	3 時間
合計	10 時間
セカンダリノードの最大ダウンタイム	4 時間
ネットワークの最大ダウンタイム	24 時間
セカンダリのバックアップ	使わない

同期の RLINK を設定するため、レプリケーションに使うネットワークの帯域は、ピーク時の書き込み速度に対応できるだけの回線を使い、書き込みの遅延を回避する必要があります。この場合、ピーク時の制約は適用されないため、最大の制約は 24 時間のネットワークのダウンタイムに対応することです。この時間内に SRL に蓄積されるデータ量は、次のようになります。

(ピーク時のアプリケーションの書き込み速度 × ピークの時間) +
(オフピーク時のアプリケーションの書き込み速度 × オフピークの時間)

この場合、計算は次のようになります。

```
1 GB/ 時 × 12 時間 + 1/4 GB/ 時 × 12 時間 = 15 GB
```

平均書き込みサイズが 2 KB であるため、ヘッダー情報の分を考慮して、25 % 増やします。24 時間のダウンタイムは非常に余裕を持たせたダウンタイムと言えるため、他の制約を処理するための調整はこれ以上必要ありません。この結果、SRL の容量は少なくとも 18.75 GB 必要であることがわかります。

2

レプリケーション パフォーマンスの チューニング

VVR のパフォーマンスに影響を与える要因として、SRL のレイアウトおよび VVR のバッファサイズ設定があります。この章では、SRL のレイアウトや VVR のバッファサイズを決定する方法を説明します。また、他の VVR カーネルパラメータの値の設定方法についても説明します。

SRL のレイアウト

この項では、SRL がアプリケーションのパフォーマンスに及ぼす影響と、適切な SRL レイアウトによりパフォーマンスがどのように改善されるかについて説明します。

SRL のレイアウトの違いによるパフォーマンスへの影響

プライマリでのアプリケーションからの書き込みは、まず SRL に書き込まれた後、データボリュームに書き込まれます。VVR はレプリケーションのモードなどのレプリケーション設定に関係なく、同じ方法で書き込みを管理します。RVG 内の異なるデータボリュームに書き込まれたデータはすべて同じ SRL に書き込まれることになるので注意してください。この結果、SRL のスループットがパフォーマンスに影響する場合があります。SRL を使うことによって、パフォーマンスに大きな悪影響が出ることはありません。次の理由があります。

- ✔ SRL への書き込みがシーケンシャルであるのに対しデータボリュームへの書き込みが行われる領域はランダムです。通常、シーケンシャル書き込みはランダム書き込みよりも高速に処理されます。

- ✔ アプリケーションの読み取り操作を実行時、SRL は使われません。そのため、アプリケーションの作業の大部分が読み取りである場合、SRL がビジーとなることはありません。

アプリケーションからのデータボリュームへの書き込み速度が SRL ボリュームへの書き込み速度を上回っている場合、アプリケーションのパフォーマンスが低下する場合があります。次の項では、パフォーマンスが改善されるように SRL をレイアウトする方法を説明しています。

SRL のストライピング

SRL ボリュームを複数の物理ディスクで構成されたストライプボリュームにすることで、書き込み速度の向上によるパフォーマンスの改善が行われる場合があります。

SRL に使うディスクの選択

VVR への書き込みリクエストはすべて SRL と要求されたデータボリュームの両方に書き込まれるため、SRL を形成するディスクとデータボリュームを形成するディスクは重複しないようにしてください。SRL ボリュームとデータボリュームが同じディスク上に存在する場合、ディスク内の SRL 部分とデータボリュームの部分の間でディスクヘッドが何度も行き来することになるので、これはパフォーマンス上問題となります。100 % 以上パフォーマンスが低下する可能性もあります。

SRL のミラー化

信頼性を向上させるために、SRL をミラー化することを推奨します。SRL に障害が発生した場合、レプリケーションは即時停止します。この状態をリカバリする唯一の方法は完全再同期ですが、その操作はかなり時間がかかるため可能な場合は避けるべきです。特定の状況下では、SRL の損失によってデータボリュームが失われることさえあります。SRL が使えなくなる危険は SRL をミラー化することで最小限に抑えることができます。

VVR のチューニング VVR

この項では、VVR のパフォーマンスに関係する、システムのカーネルパラメータの調整方法について説明します。パフォーマンスを最適化するには、使用可能なシステムリソースに応じて、一部のカーネルパラメータ値を調整する必要もあります。

カーネルパラメータを変更するには、vxtune ユーティリティまたはシステム専用インターフェースを使って値を設定します。Solaris の場合、/kernel/drv/vxio.conf ファイルにカーネルパラメータの値を定義します。カーネルパラメータの値を変更する手順について詳しくは『Veritas Volume Replicator 管理者ガイド』を参照してください。共有ディスクグループ環境では、それぞれの VVR バッファ領域をノードごとに同じ値に設定する必要があることに注意してください。

VVR バッファ領域

VVR ではデータの複製に次のバッファが使われます。

- ✔ プライマリ上の書き込みバッファ領域
- ✔ プライマリ上のリードバックバッファ領域
- ✔ セカンダリ上のバッファ領域

プライマリ上の書き込みバッファ領域

VVR では、レプリケーション先が専用ディスクグループと共有ディスクグループのどちらであるかに応じて、書き込み処理が異なります。また、共有ディスクグループ環境では、レプリケーションの同期モードと非同期モードにより VVR の書き込み処理が異なります。

書き込みリクエストが発行された場合、プライマリ上の書き込みバッファ領域から書き込みバッファが割り当てられます。ディスクグループが共有されており、書き込み情報がログ所有者に発行された場合は、書き込みバッファがログ所有者の書き込みバッファ領域から割り当てられます。

ディスクグループが共有されており、VVR のレプリケーションが同期モードで行なわれ、書き込み情報が非ログ所有者に発行された場合は、書き込み情報はログ所有者に送信されます。ログ所有者上では、VVR は書き込み情報を書き込み転送バッファ領域に受信し、書き込みバッファ領域にコピーします。このプロセスを書き込み転送と呼びます。

ディスクグループが共有されており、VVR のレプリケーションが非同期モードで行なわれ、書き込み情報が非ログ所有者に発行された場合は、VVR は書き込みに関するメタデータ情報をログ所有者とやりとりします。VVR は、非ログ所有者のメタデータ情報を受信すると、非ログ所有者にローカルで書き込みを行ないます。このプロセスをメタデータ転送と呼びます。

専用ディスクグループまたは書き込み転送を使う共有ディスクグループでは、データがプライマリ SRL に書き込まれすべてのセカンダリに同期モードで送信されるまで、バッファは解放されません。一方、非同期モードの RLINK の場合は、プライマリでのアプリケーションからの書き込み速度を維持することが優先されるために、セカンダリへ送信するためのデータがプライマリの書き込みバッファ上に保存されます。その結果、プライマリ上の書き込みバッファ領域は減少します。このとき、VVR は書き込みバッファの一部の領域を開放します。この処理が行われた場合、セカンダリへ転送するデータは、バッファ上の物ではなく SRL からリードバックしたデータが使われます。プライマリでの書き込みバッファの領域が開放されるために、プライマリでの書き込みパフォーマンスが低下することはありません。

プライマリ上のリードバックバッファ領域

VVR によって書き込みバッファから解放された書き込みデータをセカンダリに送信する準備ができると、そのデータは SRL からリードバックされます。SRL からリードバックされたデータは、プライマリのリードバックバッファ領域に格納されます。

6 ページの「レプリケーションのモードの選択」で説明するように、SRL からデータをリードバックする必要があるため SRL への書き込みパフォーマンスに影響があります。これは、SRL で書き込みの情報の順次 I/O 以外に、リードバックのための I/O が発生するためです。また、SRL からデータをリードバックすること自体が、システムの負荷になるために、セカンダリへのデータ送信の速度が低下する可能性があります。

セカンダリへ未送信のデータが書き込みバッファから解放される可能性があるのは、非同期モードの RLINK を設定している場合だけであることに注意してください。同期モードでレプリケーションを実行している場合は、SRL からのリードバックは発生しません。

セカンダリ上のバッファ領域

プライマリから送信されたデータは、セカンダリのバッファ上に格納されます。次に、バッファ上のデータをセカンダリのデータボリュームに書き込みます。同期モードの場合であっても、セカンダリのデータボリュームにデータが書き込まれる前に、プライマリで VVR がアプリケーションに書き込み完了を返す場合があります。特に同期モードの場合、セカンダリのバッファ領域が不足していたときには、セカンダリはプライマリから送信されたデータの受信を拒否しますので、プライマリで次の書き込みが行えなくなります。このとき、プライマリで次の書き込みが行えないのは、データが送信不可能な状態、と判断されます。そのため、最終的に、同期モードでレプリケーションをするのには、ネットワークの帯域が不十分な場合と同様の結果になります。

非同期モードのセカンダリの場合は、12 ページの「遅延保護および SRL 保護の選択」で説明した保護を有効にしていない限り、セカンダリに未送信な書き込み

情報の件数によるプライマリの書き込みが制限されることはありません。よって、非同期モードでレプリケーションを実行している場合は、プライマリでアプリケーションの書き込み速度低下は発生しませんが、RDS 内で、同期モードによるレプリケーションも行っている場合は、書き込み速度は低下します。

VVR バッファ領域のカーネルパラメータ

VVR で利用できるバッファのサイズは、アプリケーションとレプリケーションのパフォーマンスに影響します。次のカーネルパラメータで、必要条件に応じてバッファ領域を調節できます。

- vol_rvio_maxpool_sz
- vol_min_lowmem_sz
- vol_max_wrspool_sz
- vol_max_rdback_sz
- vol_max_nmpool_sz

vxmemstat コマンドを使うことで、VVR で使うバッファ領域を監視できます。これらの各カーネルパラメータについては、以降の項で説明します。カーネルパラメータの値を変更する手順については、『Veritas Volume Replicator 管理者ガイド』を参照してください。

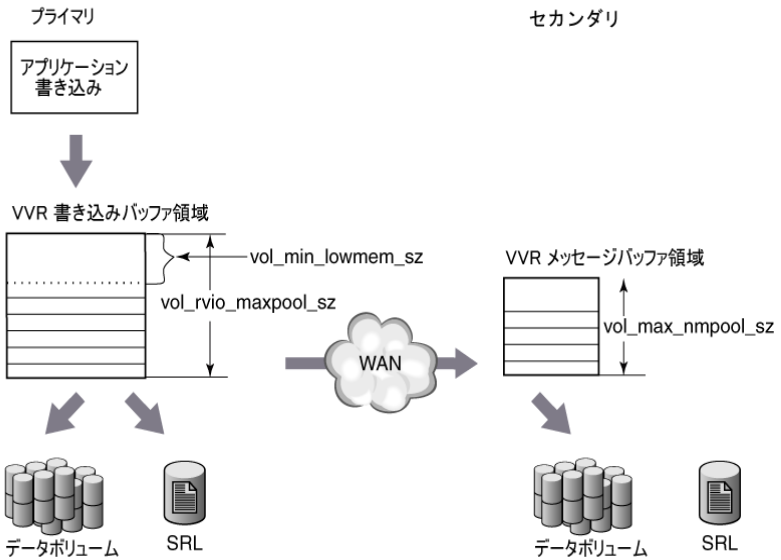
専用ディスクグループ内のプライマリの書き込みバッファ領域のカーネルパラメータ

次のカーネルパラメータは、専用ディスクグループ内のプライマリ上の書き込みバッファ領域について定義します。

- vol_rvio_maxpool_sz
- vol_min_lowmem_sz

プライマリでの書き込みを保存するためのバッファは、カーネルパラメータ vol_rvio_maxpool_sz で定義します。このパラメータのデフォルト値は 128 MB です。

図 2-2　　プライマリとセカンダリ間での VVR によるバッファの使用



書き込みバッファに、書き込みリクエストの処理に必要な空き領域がない場合は、書き込みが保留されます。つまり、VVR は現在進行中の書き込みが完了して、メモリを解放した後に新しい書き込みを処理します。さらに、31 ページの「プライマリ上の書き込みバッファ領域」で説明しているように、バッファ領域が小さい場合、VVR はセカンダリに未送信の書き込みもバッファから解放し、データ送信時には SRL からリードバックするようにします。これらの問題を同時に改善するには、vol_rvio_maxpool_sz の値を大きくします。プライマリで行われる書き込みを十分に保存できる領域を vol_rvio_maxpool_sz で定義することで、バッファによる同時書き込み数を増加させ、SRL からのリードバックを減少させます。vol_rvio_maxpool_sz カーネルパラメータの値を減らす場合は、この処理を実行しているシステムの RVG をすべて停止させます。

バッファの解放を実行するかどうかは、VVR が書き込みバッファの空き領域を調べ、それがしきい値よりも小さい場合に行われます。の最大書き込みサイズの 2 倍に相当する約 520 KB が、しきい値となります。このしきい値が小さすぎる場合、プライマリで書き込み情報がバッファ上に長い間存在することになり、バッファ領域を圧迫します。そのために、新規書き込み用のバッファが不足する事態が発生し、新規書き込みが行えなくなることもあります。

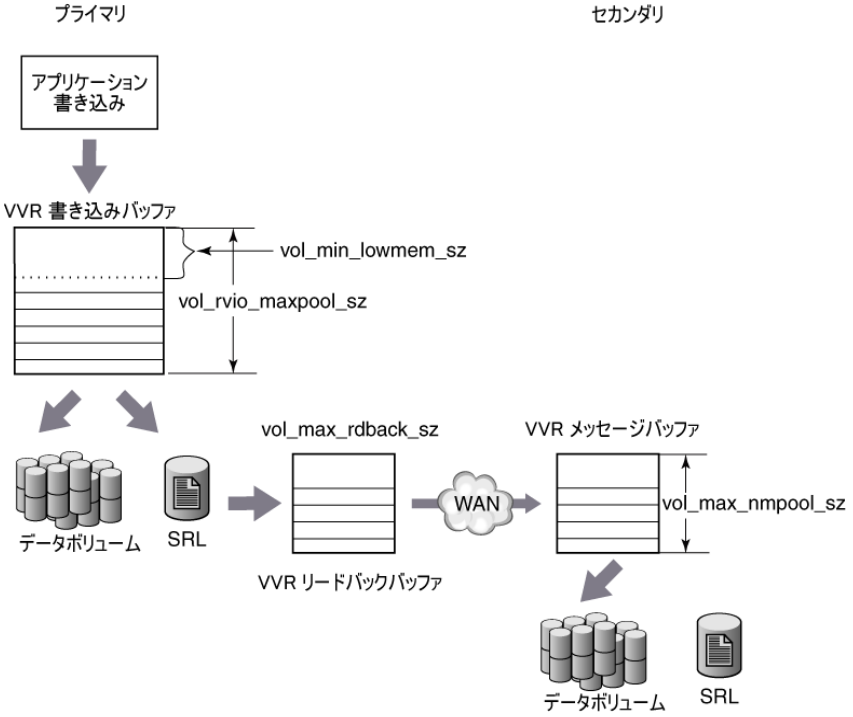
しきい値を大きくするには、カーネルパラメータ vol_min_lowmem_sz の値を大きくします。このパラメータには、最低 520 KB 以上を設定してください。パラメータ値を決定する場合は、データボリュームへの同時書き込み数を N、平均 I/O サイズを I とした場合、3 x N x I に相当する値をパラメータに設定してください。なお、平均 I/O サイズとして、8 KB 未満を設定する場合は、8 KB に切り上げて算出します。vol_min_lowmem_sz カーネルパラメータは自動調整可能で、VVR は受信している書き込み情報に応じてカーネルパラメータ値を増減します。vxtune ユーティリティまたはシステム専用インターフェースを使ってこのカーネルパラメータに指定した値は初期値として使われ、アプリケーションの書き込み動作に応じて変更することができます。

同期モードでレプリケーションを行う場合、SRL からのリードバックは発生しません。そのため、このカーネルパラメータによるチューニングは、非同期モードのレプリケーションを実行する場合だけであることに注意してください。

最大同時書き込み数（N）と平均書き込みサイズ（I）を決定する際には、vxrvg stats コマンドの実行結果を参照してください。

リードバックバッファ領域のカーネルパラメータ

リードバックで使うバッファの大きさは、カーネルパラメータ vol_max_rdback_sz で定義します。デフォルトでは、64 MB です。より多くのデータのリードバックに対応するためには、vol_max_rdback_sz の値を増やします。複数の RVG で、複数の非同期モードのセカンダリにレプリケーションを実行する場合も、この値を大きくする必要があります。



バッファ領域の使用状態をモニタする場合は、vxmemstat コマンドを使います。領域がすべて使われていることが判明した場合、カーネルパラメータ vol_max_rdback_sz の値を大きくして、リードバックのパフォーマンスを改善させます。カーネルパラメータ vol_max_rdback_sz の値を小さくする場合、すべてのセカンダリとのレプリケーションを、変更前に一時停止しておく必要があります。

共有ディスクグループ内のプライマリのバッファ領域のカーネルパラメータ

共有ディスクグループ環境で、次のカーネルパラメータは、非同期モードでのレプリケーション時のプライマリ上でのバッファ領域について定義します。

- vol_rvio_maxpool_sz
- vol_min_lowmem_sz
- vol_max_rdback_sz

非同期モードでは、カーネルパラメータは、34 ページの「専用ディスクグループ内のプライマリの書き込みバッファ領域のカーネルパラメータ」で説明したように機能します。vol_rvio_maxpool_sz カーネルパラメータはすべてのノードに適用されます。vol_min_lowmem_sz vol_max_rdback_sz カーネルパ

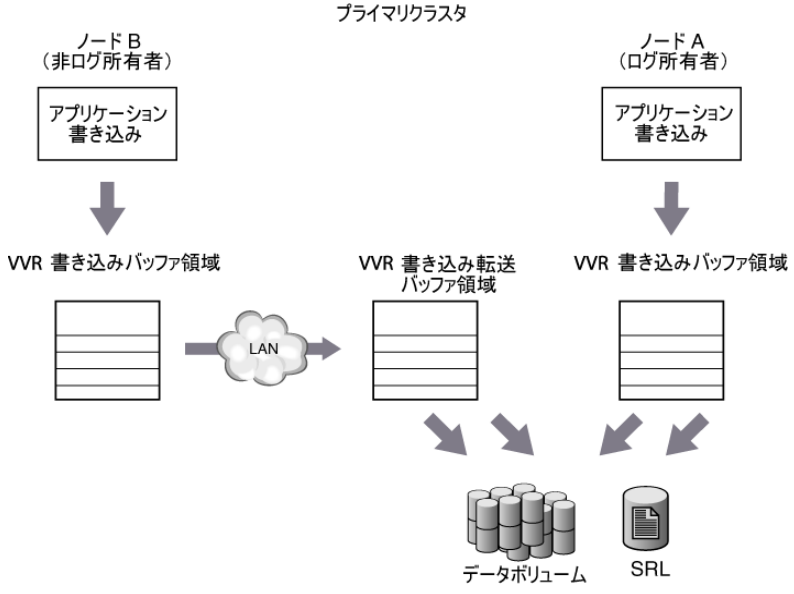
ラメータは、ログ所有者ノードに対して適用されます。ただし、いずれかのノードがその後ログ所有者になるので、これらのカーネルパラメータはすべてのノードで同じ値に設定する必要もあります。

共有ディスクグループ環境で、次のカーネルパラメータは、同期モードでのレプリケーション時のプライマリ上でのバッファ領域について定義します。

- `vol_max_wrspool_sz`
- `vol_rvio_maxpool_sz`

同期モードでレプリケートする場合、`vol_rvio_maxpool_sz` カーネルパラメータは、34 ページの「専用ディスクグループ内のプライマリの書き込みバッファ領域のカーネルパラメータ」で説明したように機能しますが、リードバックは回避しません。このカーネルパラメータは、共有ディスクグループ内のすべてのノードで設定してください。さらに、非ログ所有者により送信された書き込み情報を受信するために、ログ所有者に割り当て可能なバッファ領域サイズは、書き込み転送バッファ領域カーネルパラメータ `vol_max_wrspool_sz` で定義します。デフォルトは 16 MB です。いずれかのノードがその後ログ所有者になるので、このカーネルパラメータはすべてのノードで同じ値に設定する必要があります。

図 2-3　　共有ディスクグループ環境のプライマリでの VVR によるバッファの使用（同期 RLINK の場合）



セカンダリのバッファ領域のカーネルパラメータ

ネットワーク経由でわたされる書き込みデータを保存するセカンダリのバッファ領域は、VVR のカーネルパラメータ vol_max_nmpool_sz で定義します。デフォルトでは、16 MB です。VVR は、セカンダリの各 RVG ごとに個別にバッファ領域を割り当てます。バッファ領域のサイズはカーネルパラメータ vol_max_nmpool_sz の値です。セカンダリのバッファ領域には、ネットワーク転送の妨げとならないように、十分な大きさを定義する必要があります。

ただし、バッファを大きくしすぎた場合、問題が生じる場合があります。書き込み情報がセカンダリに到達すると、セカンダリはプライマリへ確認応答を送信し、プライマリはその応答を元に転送完了と判断します。実際に、セカンダリのデータボリュームに書き込みが完了すると、セカンダリは別の確認応答を送信し、この応答によって、プライマリ上の SRL から応答があった情報が削除可能となります。ただし、最初の確認通知から 2 番目の通知が最初の確認応答から 1 分以内に送信されないと、プライマリは RLINK を切断します。RLINK はすぐに再接続されますが、この処理によってネットワークによるデータ転送は中断され、他の問題が発生する可能性があります。この場合、1 分未満の書き込み情報を保存できるだけのサイズに、セカンダリのバッファ領域を調整する必要があります。このサイズは、ディスクへのデータ書き込み速度によって異なります。また、この速度もディスク自体、I/O バス、システムの負荷および書き込みの性質（ランダムかシーケンシャルか、少量か大量か）によって異なります。

書き込み速度が W MB/sec である場合、バッファのサイズは W * 50 MB すなわち 50 秒間での書き込み量を超えないようにする必要があります。

W の値を測定するには様々な方法があります。セカンダリ上のディスクレイアウトとボリュームレイアウトがプライマリ上のものと類似しており、レプリケーションを設定する前にプライマリで測定したプライマリの I/O 統計情報がある場合には、その情報の最大書き込み速度を W の値とします。

また、レプリケーションがすでに設定されている場合は、まずセカンダリのバッファ領域のサイズをタイムアウトとメモリエラーを回避できる大きさに設定します。

レプリケーションの書き込みがピーク速度で実行されているときに、次のコマンドを実行し、メモリエラーが発生していないこと、タイムアウトエラー数がわずかであることを確認します。

```
# vxrlink -g diskgroup -i5 stats rlink_name
```

次に、vxstat コマンドを実行して、書き込みの最低速度を取得します。

```
# vxstat -g diskgroup -i5
```

次のような結果が出力されます。

```
                        OPERATIONS            BLOCKS      AVG
   TIME(ms)
   TYP NAME           READ     WRITE      READ    WRITE   READ  WRITE

   Mon 29 Sep 2003 07:33:07 AM PDT
   vol srl1              0      1245         0      1663   0.0   9.0
   vol archive           0       750         0       750   0.0   9.0
```

```
vol archive-L01       0       384         0       384    0.0    5.9
vol archive-L02       0       366         0       366    0.0   12.1
vol ora02             0       450         0       900    0.0   11.1
vol ora03             0         0         0         0    0.0    0.0
vol ora04             0         0         0         0    0.0    0.0


Mon 29 Sep 2003 07:33:12 AM PDT
vol srl1              0       991         0      1389    0.0   20.1
vol archive           0       495         0       495    0.0   10.1
vol archive-L01       0       256         0       256    0.0    5.9
vol archive-L02       0       239         0       239    0.0   14.4
vol ora02             0       494         0       988    0.0   10.0
vol ora03             0         0         0         0    0.0    0.0
vol ora04             0         0         0         0    0.0    0.0
```

各間隔について、SRL への書き込みを除外して、書き込みブロック数の数値を加算します。また、すべてのサブボリュームへの書き込みも除外します。たとえば、archive-L01 および archive-L02 はボリューム archive のサブボリュームです。サブボリュームへの書き込みの統計情報は、ボリューム archive の統計情報に追加されています。テストを実行する間隔、合計時間および回数は、必要に応じて変更できます。この例では、間隔は 5 秒、単位はブロックで、ブロックサイズが 2 KB のマシンのため、間隔ごとの MB 数 M は、（合計値 × 2048）/（1024 × 1024）です。合計値は間隔ごとの合計です。したがって、1 秒間の MB 数は M/5 になり、バッファのサイズは（M/5）× 50 になります。もし、このセカンダリノードに複数の RVG が設定されていない場合には、バッファサイズをそれ以上大きくしないようにする必要があります。

SRL への書き込みは、アプリケーションの I/O 負荷には含まれません。ただし、非同期モードでは、セカンダリは受信している更新をセカンダリ SRL とデータボリュームの両方へ書き込むため、vol_max_nmpool_sz の値を少し大きくする必要があります。ただし、この項の始めに説明した問題を回避するため、書き込みが 1 分を超えてプールに残ることのないように、vol_max_nmpool_sz の値を設定する必要があります。

DCM 再生のブロックサイズ

データ変更マップ（DCM）が再生されると、データはブロック単位でセカンダリに送信されます。カーネルパラメータ vol_dcm_replay_size で、ネットワークの状態に応じて DCM 再生のブロックサイズを設定できます。vol_dcm_replay_size のデフォルト値は、256 KB です。カーネルパラメータ vol_dcm_replay_size の値を小さくすると、遅延が大きい環境でのパフォーマンスが向上します。

ハートビートタイムアウト

VVR では、ハートビートの機構を使って、プライマリホストとセカンダリホスト間の通信エラーを検出します。プライマリとセカンダリの間でハートビートが交わされた後で、RLINK が接続されます。RLINK はリモートホストでハートビートが認識され続けている間、接続状態が続きます。ハートビートが無応答状態にあるときの最大間隔を、ハートビートタイムアウト値と呼びます。指定したタイムアウト値以内にハートビートの確認応答が行われない場合、VVR は RLINK を切断します。

カーネルパラメータ vol_nm_hb_timeout で、ハートビートタイムアウト値を設定できます。デフォルトの値は 10 秒です。ネットワークでの遅延が大きい場合、カーネルパラメータ vol_nm_hb_timeout のデフォルト値を大きくすれば、RLINK が誤って切断されることを回避できます。

メモリのチャンクサイズ

カーネルパラメータの voliomem_chunk_size で、VVR がシステムのメモリ割り当てや解放を行う際の粒度、すなわちメモリのチャンクサイズを設定します。メモリ領域とは、1 度のオペレーションでディスクに書き込むメモリサイズを意味しています。アプリケーションの書き込みサイズがメモリ領域よりも大きい場合、書き込み情報が分割されることにより複数のオペレーションが生じ、パフォーマンスが低下します。

デフォルトサイズは 64 KB です。書き込み量が多いアプリケーションを使っている場合は、カーネルパラメータ voliomem_chunk_size のサイズを大きくすると、レプリケーションのパフォーマンスが向上します。
voliomem_chunk_size で設定可能な最大値は、128 KB です。

VVR とネットワークアドレス変換ファイアウォール

VVR では、ハートビートの機構を使って、プライマリホストとセカンダリホスト間の通信エラーを検出します。また、VVR はハートビートメッセージ内に付加されている IP アドレスの情報を使って、相手側とのハートビートを実行しています。

ネットワークアドレス変換（NAT）ベースのファイアウォールを経由してレプリケーションを実行する場合、VVR は、ハートビートメッセージの IP アドレスではなく、NAT で変換された外向きの IP アドレスを使う必要があります。ハートビートメッセージの IP アドレスを使うと、ハートビート確認応答がファイアウォールで破棄され、レプリケーションを開始できません。

カーネルパラメータ vol_vvr_use_nat は、VVR が NAT ベースのファイアウォールを経由して通信できるように、受け取ったメッセージの変換済みアドレスを使うように VVR に指示します。構成内に NAT ベースのファイアウォールがある場合にのみ、このカーネルパラメータを 1 に設定します。

用語集

consistent フラグ（consistent）
ファイルシステムやデータベースなど、対象データを使うシステムまたはアプリケーションでデータのリカバリが可能であることを示す用語。VVR では、整合した状態のセカンダリをテイクオーバーに使用できます。

DCM（Data Change Map）
プライマリ RVG 上のデータボリュームと任意に関連付けられるビットマップを含むオブジェクト。ビットは、プライマリとセカンダリ間で異なるデータの領域を表します。ビットマップは、同期および再同期中に使用されます。

latencyprot
「遅延保護（latency protection）」を参照してください。

RDS（Replicated Data Set）
プライマリの RVG と、1 つ以上のセカンダリ ホスト上にあるそれに対応する RVG を 1 つのグループとしてまとめたもの。

RLINK
RLINK とは、プライマリ ノードおよびセカンダリ ノード上の対応する RVG 間での通信リンクのことです。

RVG（Replicated Volume Group）
1 セットのデータボリューム、1 つ以上の RLINK および、1 つの SRL から構成される VVR のコンポーネント。VVR は、アプリケーションを実行しているノードで、プライマリ RVG から 1 つ以上のセカンダリ RVG へレプリケートします。

SRL オーバーフロー保護（SRL overflow protection）
VVR の機能の 1 つ。プライマリ ノード SRL のオーバーフロー後に、セカンダリ RVG で完全再同期を行う必要がないようにします。

SRL（Storage Replicator Log）
ストレージレプリケータログ（SRL）は、RVG で利用する書き込みの循環バッファです。SRL に格納された書き込みは、プライマリからセカンダリへ転送されるまで待機するか、RVG 内のデータボリュームに書き込まれるまで待機します。

STALE
RLINK がまだ接続されていないか、オーバーフローしたことを示す RLINK 状態。

Volume Replicator オブジェクト（Volume Replicator Objects）
RVG（Replicated Volume Group）、SRL（Storage Replicator Log）、RLINK、DCM（Data Change Map）など、レプリケーションに使われるオブジェクト。

書き込み転送（write shipping）
ログ所有者以外のノードで発行された書き込みを、クラスタネットワークを経由してログ所有者に送信する処理。

更新（update）
セカンダリに送られたアプリケーションの書き込みに対応するプライマリからのデータ。

自動同期（Automatic Synchronization）
VVR の機能の 1 つ。プライマリ上でアプリケーションが実行しているときにセカンダリを同期します。

スロットル（throttling）
プライマリで書き込みにウェイトを追加し、書き込み速度を低く抑える機構。

セカンダリ RVG（Replicated Volume Group）
「RVG（Replicated Volume Group）」を参照してください。

セカンダリチェックポイント（Secondary checkpoint）
「チェックポイント（checkpoint）」を参照してください。

セカンダリノード（Secondary node）
VVR によるプライマリからのデータのレプリケーション先のノード。

チェックポイント（checkpoint）
VVR の機能の 1 つ。現在の位置よりも前のポイントから SRL を再開します。チェックポイントは、後で再開する SRL のセクションの始点と終点を示します。

遅延保護（latency protection）
非同期モードで動作する RLINK は遅れることがあるため、RLINK の遅延保護属性（`latencyprot`）を使って、未送信の書き込み情報数を制限します。未処理の書き込み要求の最大数は、`latency_high_mark` で設定されている値を超えることはできません。そのときは、未処理の書き込み数が `latency_low_mark` に低下するまで増加できません。

データボリューム（data volume）
RVG に関連付けられ、アプリケーションデータを格納しているボリューム。

同期（synchronization）
セカンダリのデータをプライマリのデータと同一にするプロセス。

同期（synchronous）
同期モードでは、プライマリ上の書き込みの正常完了をアプリケーションが認識するまで、各書き込み要求に対するセカンダリの確認応答を待機することによって、セカンダリがプライマリと同様の最新状態に保たれます。

ハートビートプロトコル（heartbeat protocol）
ハートビートプロトコルとはメッセージの連続的なやり取りであり、RDS 内のノードがネットワーク切断やノードのクラッシュをすべて検出できるようにします。このプロトコルにより、ノードは後から接続を再確立できます。

バッファ領域（buffer space）
VVR が書き込みを処理し、レプリケーションを実行するために使うメモリ。

非同期（asynchronous）
非同期モードでは、書き込みがキューに格納され、後で転送するためにプライマリの SRL に書き込み情報が保存されます。

不整合（inconsistent）
VVR では、テイクオーバーの対象として適切でない場合に、セカンダリは不整合の状態になります。アプリケーションをリカバリできないことがわかっているためです。

プライマリ RVG（Replicated Volume Group）
「RVG（Replicated Volume Group）」を参照してください。

プライマリ ノード SRL のオーバーフロー（Primary node SRL overflow）
プライマリ SRL の容量は限られているため、RLINK に対する更新処理の停止が長引くと、SRL の限度を超えてしまい、RLINK を最新の状態にするために必要な更新履歴をすべて維持することができなくなる可能性があります。このような状況になると、RLINK は STALE に設定され、手動でリカバリ、つまり同期を実行しないとレプリケーションを行うことができなくなります。

プライマリ ノード（Primary node）
プライマリ ノードとはアプリケーションを実行しているノードであり、データをここからセカンダリにレプリケートします。

メタデータ転送（metadata shipping）
非同期モードでレプリケーションを実行する場合に、書き込みを発行する非ログ所有者ノードとログ所有者間で情報をやり取りし、非ログ所有者ノードにローカルで書き込みを行なうプロセス。

リードバック（readback）
RLINK で送信するために、SRL から書き込み要求を取り込む処理。

ログ所有者（logowner）
共有ディスクグループ環境でレプリケーションを行う際に、VVR がレプリケーションを実行するノード。同期 RLINK の場合、VVR は、ログ所有者ノードでも書き込みを実行します。

索引

ひ

ふ

ほ

め

り

れ

ろ