

Oracleホワイト・ペーパー
2014年2月

SAP HANA高可用性機能の分析

ORACLE®

概要	2
SAP HANAアーキテクチャ概要	3
SAP HANAデータベース	3
SAP HANAアプライアンス	4
分析：SAP HANAの高可用性（HA）機能	5
SAP HANA：スケーラビリティ機能の分析	5
SAP HANA：ディザスタ・リカバリ機能の分析	15
SAP HANA：バックアップとリカバリ機能の分析	20
SAP HANA：全HA機能の概要分析	23
顧客事例	25
高可用性：留意事項	25
結論	26
参考資料	27

概要

今日のビジネスにおいて、高可用性（HA）はすべてのエンタープライズレベルのITインフラストラクチャにとって極めて重要な検討事項となっています。重要なHA機能が欠落したテクノロジー・プラットフォームはエンタープライズレベルとは言えません。

SAPによるとHANAは、リアルタイム分析とリアルタイム・アプリケーションのための最新プラットフォームとされています。SAP HANAのドキュメント[1]でも"SAP HANAは、高可用性を十分考慮して設計されている。"と記載されています。しかし、このホワイト・ペーパーの分析が示すとおり、この主張は正しくありません。

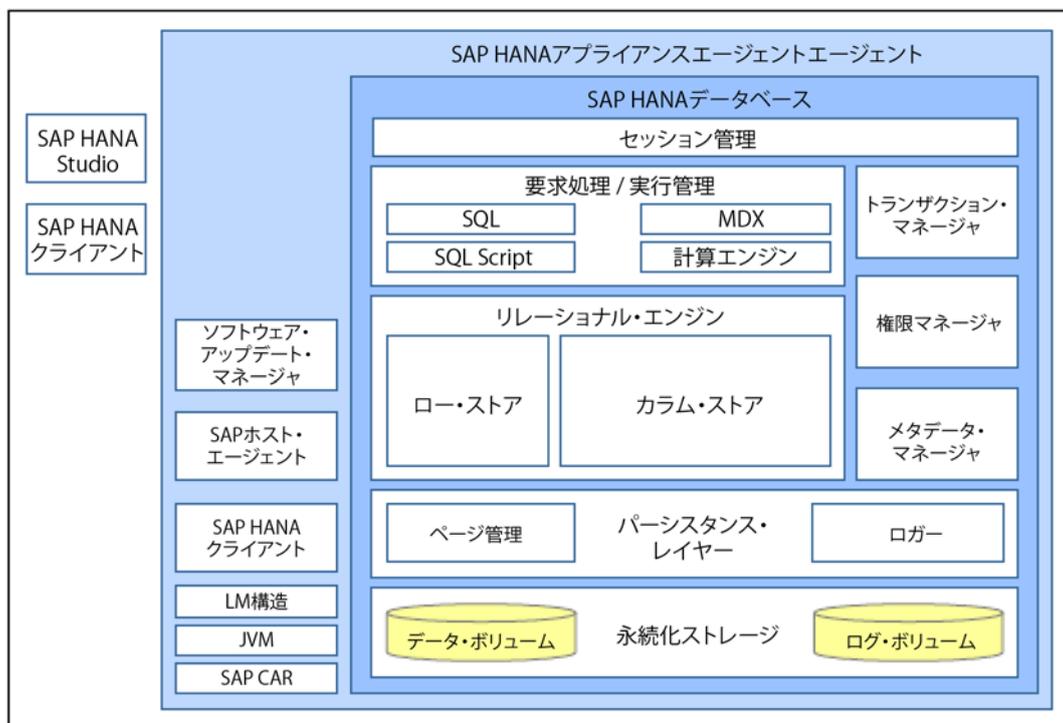
このホワイト・ペーパーでは、SAP HANAサポートパッケージスタック（SPS）07 [2]で提供されるHA機能について詳細に分析します。また、SAP HANAのHA機能とOracle Maximum Availability Architecture（MAA）[3]の一部としてOracle Databaseで提供されるHA機能との比較についてもご説明します。

このホワイト・ペーパーから明らかのように、HAに関する限り、HANAのテクニカル・アーキテクチャには重大な欠点があり、強固なHAアーキテクチャを実装するのに不可欠な重要な機能がいくつか欠落しています。SAP HANAは、エンタープライズレベルとは言えません。

このホワイト・ペーパーは、ITの管理チームや技術チームのメンバーで、SAP HANAが自分たちの基幹アプリケーションに対して有効であるかどうかに関心があったり、HANAアーキテクチャが定評のある高可用性テクノロジーと比べてどうなのかをもっと良く知りたいと考えている方を対象としています。

SAP HANAアーキテクチャ概要

SAPで使用される“SAP HANA”という用語は、データ分析向けのSAPのインメモリ・アプライアンスを意味しています。SAP HANAはハードウェアとソフトウェアの組み合わせであり、SAPのハードウェア・パートナーと提携して最適化されたアプライアンスとして提供されます。この用語はまた、SAPのインメモリ・データベースを指す場合にも使われます。このデータベースは、ローベース、カラムベース、オブジェクト・ベースの各データベース・テクノロジーを組み合わせたハイブリッドのインメモリ・データベースです。次の図に、SAP HANAアーキテクチャの概要を示します。



図：1：SAP HANAアーキテクチャ[5]

SAP HANAデータベース

SAP HANAデータベースは、2つのデータベース・エンジンから構成されます。

- カラムベース・ストア。列にリレーショナル・データが格納され、分析業務で集計して使用される大量のデータを含むデータ・マート表の保持に最適化されています。
- ローベース・ストア。行にリレーショナル・データが格納されます。ローストアは書き込み処理に最適化されており、圧縮率は低く、問合せのパフォーマンスもカラムベース・ストアに比べると大幅に劣ります。

データを格納するエンジンは、表作成時に表単位で選択できます（後で変更することも可能です）。ローストアの表は、起動時にメモリに読み込まれます。一方、カラムストアの表は、起動時でもオンデマンド（SAP HANAデータベースの通常運用中）でも読み込まれます。

どちらのエンジンも共通のパーシスタンス・レイヤー[6]を共有しており、両エンジンに対するデータの永続性が保たれます。インメモリ・データベース・ページ（Oracleブロックに相当）に対する変更は、永続的なストレージのデータ・ボリュームに書き込むセーブポイントによって永続化されます。この書き込みは少なくとも5分ごとに自動的に実行されますが変更も可能です。さらに、SAP HANAデータベースでコミットされるトランザクションはすべて、永続的なストレージのログ・ボリューム（OracleのREDOログに相当）に同期して書き込まれるログ・エントリという形でパーシスタンス・レイヤーのロガーによって永続化されます。

SAP HANAデータベースでは、SQL (JDBC/ODBC) 、MDX (ODBO) およびBICS (SQL DBC) がサポートされます。BICSは、SAP HANA独自のSQLスクリプト言語で、データ処理量の多いアプリケーションのロジックをSAP HANAデータベースに落とし込むのに使用できるSQLの拡張機能です（Oracleのストアド・プロシージャに相当）。

SAP HANAアプライアンス

SAP HANAデータベースから構成されるSAP HANAアプライアンスには、前述のようにデータベースと連携して、管理、運用するのに必要なコンポーネントが追加されています。アプライアンスにはSAP HANA Studioのインストール・ファイルが含まれます。SAP HANA Studioは、EclipseベースのSAP HANAとSAP HANAクライアント用の管理/データモデリング・ツールで、アプリケーションをSAP HANAデータベースに接続するために必要なライブラリ・セットです。SAP HANAのSoftware Update Manager (SUM) は、SAPマーケットプレイスなどのソースからホスト・エージェントを使用してSAP HANAのアップデートを自動的にダウンロードしてインストールできるフレームワークです。

SAPはCiscoやDell、IBM、HP、富士通といったハードウェア・ベンダーと提携して、インテルXeonベースのアプライアンス・モデルでSAP HANAソフトウェアを実行するのに必要なインフラストラクチャを提供しています。

分析：SAP HANAの高可用性（HA）機能

SAP HANAのHA機能の分析は次の3つの要素に対して実施できます。

1. スケーラビリティ対応
2. ディザスタ・リカバリ（DR）対応
3. バックアップとリカバリ対応

次の節で各要素をさらに詳細に掘り下げます。

SAP HANA：スケーラビリティ機能の分析

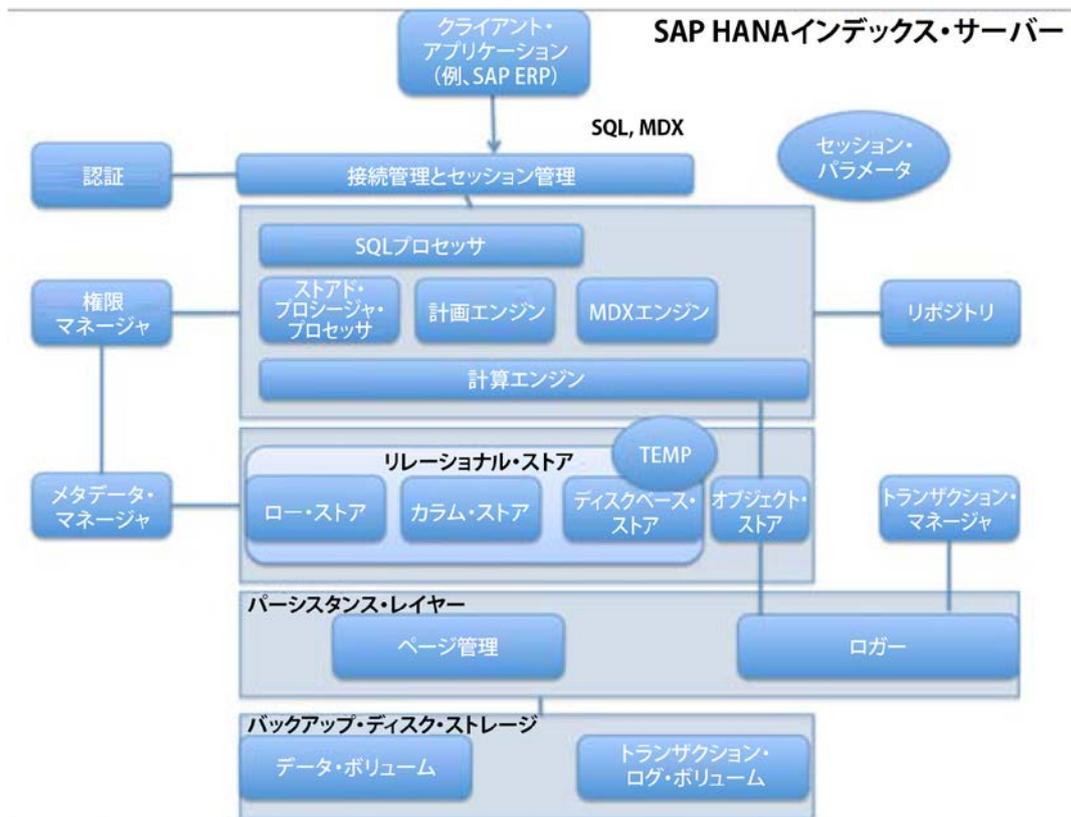
HANAでは、1つのSAP HANAクラスタ[7]で複数のサーバーを使用してスケーラビリティを実現しています。基盤となるデータにシェアード・ナッシング・アプローチを採用しており、そのためデータを複数のサーバー（ホスト）に分割する必要があります。SAPではこのレイアウトは通常、分散型SAP HANAデータベース/システムと呼ばれています。このクラスタは通常N台のアクティブ・サーバーとM台のスタンバイ・サーバーを運用し、共有ファイル・システムにアクセスします。各サーバーは、4CPU/512GBまたは8CPU/1TBで、最大認定構成は1クラスタでサーバーが56台です[8]。

分散型HANAシステム：ネーム・サーバー、インデックス・サーバー、統計サーバー

インデックス/ネーム/統計システムの各コンポーネントは、分散型HANAシステムにとって不可欠の注目すべきコンポーネントです。シングルホスト・システムでは、こうしたコンポーネントはすべて1台のホスト上の1つのSAP HANAデータベース・インスタンスにインストールされます。分散型SAP HANAシステムでは、別々のホスト上の複数のデータベース・インスタンスにインストールされます。

- ネーム・サーバー—SAP HANAシステムのトポロジ情報を管理します。分散型システムではSAP HANAデータベース・インスタンスが複数のサーバーに分散し、ネーム・サーバーはコンポーネントがどのサーバーで動作し、どのデータがどのサーバーにあるかを把握しています。
- インデックス・サーバー—実際のデータ・ストアとデータ処理用エンジンを搭載しています。
- 統計サーバー—SAP HANAシステムに属するすべてのコンポーネントの状態、パフォーマンス、リソース消費に関する情報を収集します。SAP HANA Studioのような監視クライアントは、この統計サーバーにアクセスしてさまざまなアラート監視状態を取得します。また、統計サーバーによって詳細分析のための測定データの履歴も提供されます。

インデックス・サーバーは、中核となるデータベース・エンジンです。次の図はIndex Serverのアーキテクチャを示しています。



図：2：SAP HANAインデックス・サーバー・アーキテクチャ[9]

分散型HANAシステムの設定では、分散システムを構築するホストに対するネーム・サーバーやインデックス・サーバーのロールに関する複雑な構成を伴います。動作は次のとおりです[10]。

- ネーム・サーバー

マスター・ネーム・サーバーは最大3台のホストで構成でき（"マスター 1、マスター 2、マスター 3"という呼称を使います）、他のサーバーは、"スレーブ"として構成する必要があります。ただし、定常状態では、"マスター n"に構成されたホストのうち1台だけが、実際に"マスター"としてのネーム・サーバー・ロールが割り当てられ、他のすべてのサーバーは、"スレーブ"としての実際のネーム・サーバー・ロールが割り当てられます。マスター・ネーム・サーバーは、アクティブ・インデックス・サーバーの起動時にストレージ/データ・ボリュームの割り当てを受け持ちます。[注：SAPの補足資料では、スレーブ・ネーム・サーバーの機能は明らかではありません。]

マスター・ネーム・サーバー・ホストに障害が発生した場合、"マスター n"ネーム・サーバーとして構成された残りのホストのうちの1台がアクティブ・マスター・ネーム・サーバーとなります。

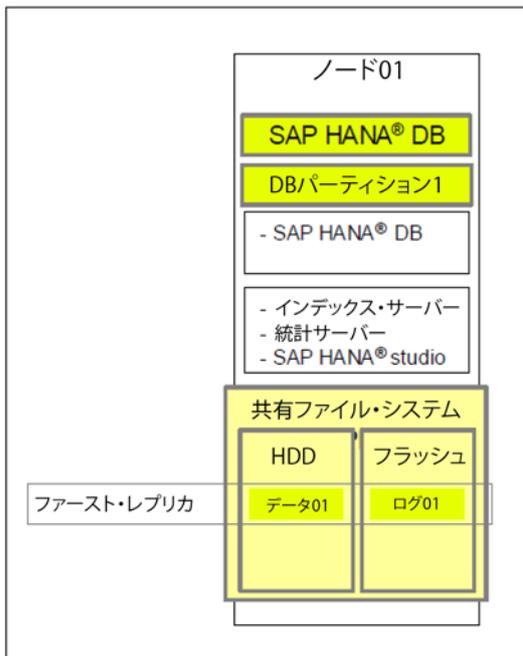
- インデックス・サーバー

ホストはセットアップ時に、"ワーカー"または"スタンバイ"インデックス・サーバーとして構成できます。定常状態では、1台のワーカー・ホストが、"マスター"としての実際のインデックス・サーバーのロールを取得します。このロールはマスター・ネーム・サーバーと同一ホストに割り当てられます。他のワーカー・ホストは、"スレーブ"としての実際のインデックス・サーバーのロールを取得します。スタンバイ・インデックス・サーバーは、"スタンバイ"としての実際のインデックス・サーバーのロールを取得します。マスター・インデックス・サーバーによって他のスレーブ・インデックス・サーバーのメタデータが提供されます。スレーブ・インデックス・サーバーはアクティブなデータベース・サーバーで、データ・ボリュームが割り当てられ、データベース接続を受け付けます。スタンバイ・インデックス・サーバーとして構成されたホストは、データベース処理には使用されません。データベースのすべてのプロセスがこのホストで動作しますが、アイドル状態でありSQL接続は受け付けません。

アクティブ・インデックス・サーバーに障害が発生した場合、アクティブ・マスター・ネーム・サーバーが、スタンバイ・インデックス・サーバー・ロールを持つホストの1台にボリュームを割り当てます。

ストレージ・レイアウト：シングル・インスタンスHANAシステム

複数のホストからなる分散型HANAシステムがデータを永続化し、ストレージにアクセスする方法の理解を深めるには、シングル・ノードHANAシステムのストレージ・レイアウトがどうなっているのかを確認することが有効です。次の図では、IBMのシステム/ストレージから見たレイアウトを示しています。



図：3：ストレージ・レイアウト：シングル・ノードSAP HANAシステム[5]

上図の2種類のストレージに実装されているのはIBMのGeneral Parallel File System (GPFS) です。共有ディスク型のクラスタ・ファイル・システムで、クラスタの複数のノード上で実行されるアプリケーションに対し、同時ファイル・アクセスを実現します。

- データ・ストレージ (SASディスク上) は、ここではHDDと表記されていますがセーブポイントを保存します。
- ログ・ストレージ (SSDドライブまたは、PCIeフラッシュ・ドライブ上) は、ここではFlashと表記されていますが、データベースのログを保存します。

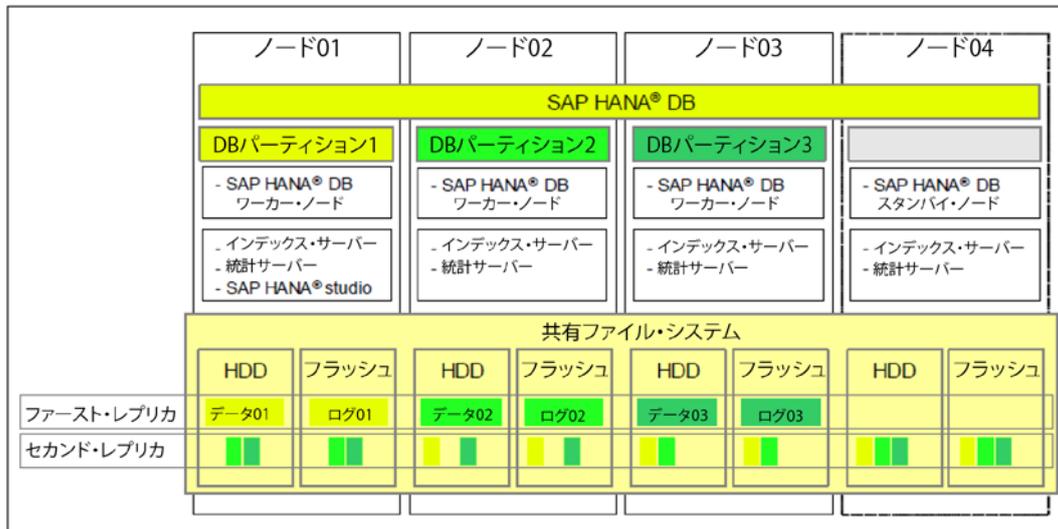
この単一ノードは、1つの単一データベース・パーティションから構成される1つの単一SAP HANAデータベースを示しています。セーブポイント (data01) とログ (log01) はどちらも1回だけ保存され (つまり、複製されません) 、上図では"ファースト・レプリカ"と表示されています。

ストレージ・レイアウト：分散型HANAシステム

拡張可能な分散型HANAシステムは単一サーバー・ソリューションと多くの点で異なります。

- HANAのソリューションは、複数の構成要素からなるクラスタ構成で、2つの個別の10Gbイーサネット・ネットワークで相互接続されています。1つはSAP HANAアプリケーション用で、もう1つはファイル・システムとの通信用です。
- SAP HANAデータベースは複数のパーティションに分割され、SAP HANAデータベースの単一インスタンスを形成します。
- クラスタの各ノードはサーバーのローカル・ストレージ・デバイス上に個々のセーブポイントとデータベース・ログを保存します。
- スタンバイ・ノードでもSAP HANAアプリケーションは実行されますが、データは一切保持されず、処理には関係しません。アクティブ・ノードのどこかに障害が発生した場合、スタンバイ・ノードが障害ノードのデータ (つまり、データベース・パーティション) も含めてロールを引き継ぎます。引き継ぎは自動処理で、マスター・ネーム・サーバーによって調整されます。
- GPFSファイル・システムはクラスタの全ノードにまたがっており、各ノードのデータをクラスタの他の全ノード (スタンバイ・ノードも含めて) が利用できるようになっています。

次の図ではこのソリューションを説明するために、例として4ノード構成を示しています。ノード04がスタンバイとして機能します。



図：4：ストレージ・レイアウト：分散型HANAシステム[5]

SAP HANAソフトウェアによって、アプリケーション要求は内部的にクラスタ全体の個別のワーカー・ノードに分配されます。ワーカー・ノードはデータを処理して中間結果を交換し、それを結合して要求元に返します。各ノードは個別のデータ・セットを維持し、セーブポイントで永続化してデータベース・ログに対するデータ変更をロギングします。

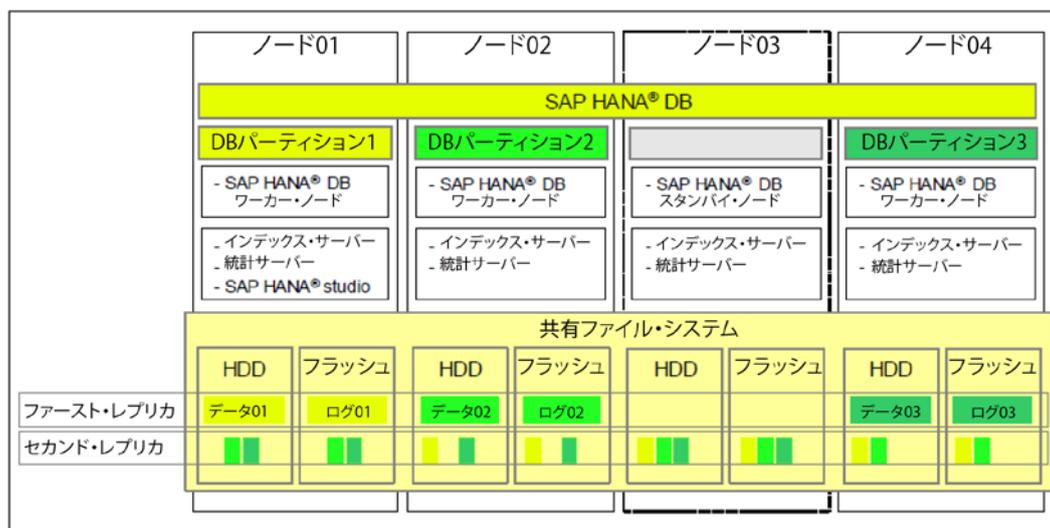
前述したシステム・コンポーネントで、データを永続化するのはアクティブ・マスター・ホスト上のネーム・サーバー・サービスのみです。スレーブ・ネーム・サーバー・ホストは、マスターと通信しますが、データは永続化しません。スタンバイ・ホスト以外のすべてのホスト上のインデックス・サーバー・サービスはデータを永続化します。統計サーバー・サービスは1つのホスト上でのみ動作し、そのホストでデータを永続化します。

GPFSのような分散ファイル・システムは個別ノードのストレージ・デバイスを1つのファイル・システムに結合することで、SAP HANAソフトウェアがクラスタ内の場所に関係なくすべてのデータにアクセスできるようにするとともに、個別のデータベース・パーティションのセーブポイントとデータベース・ログが、パーティションが割り当てられたノードにローカル接続されたストレージ・デバイス（ディスクとフラッシュ）に適切に保存されることを保証します。

分散型HANAシステム：ノードのフェイルオーバー

障害を起こしたノードのデータベース・パーティションを引き継げるように、スタンバイ・ノードは障害ノードのセーブポイントとデータベース・ログを読み込んで、データベース・パーティションを復元し、障害ノードに代わって運用を再開する必要があります。上記の図を例として使って、ノード03に問題が発生して停止したと仮定します。マスター・ノード（ノード01）はこの障害を認識してスタンバイ・ノード（ノード04）に障害ノードを引き継ぐよう指示します。メモリ内のデータベース・パーティション3を再構築して、クラスタ内でのノード03のロールを引き継げるようにするために、ノード04はGPFSファイル・システムからノード03のセーブポイントとデータベース・ログを読み込み、メモリにセーブポイントのデータを復元し、ログを再適用してメモリ内のパーティション・データをノード03の障害前の状態に正確に一致させます。これが完了すると、ノード04はアクティブ・インデックス・サーバーとして動作し、データベース・クラスタが復元されます。障害が発生していないノードのデータ・アクセスは継続可能ですが、一時停止の範囲はHANAの補足資料からは不明です。

ノード03の障害原因が修正されれば、新しいスタンバイ・システムとしてクラスタに再び統合できます。



図：5：フェイルオーバー分散型HANAシステム[5]

ノードに修復不能なハードウェア・エラーが発生すると、ノードのデータを保持するストレージ・デバイスは使用できなくなるか、壊れてしまう可能性もあります。こうした危険を緩和するため、GPFSファイル・システムでは各ノードのデータを他の複数のノードに複製するように構成して、（ローカル・ストレージを持つ）1台のノードの停止によるデータ損失を防止できます。複製は、GPFSによって同期して実行されます。すなわち、書き込み処理が完了するのはローカルと複製先の両方に書き込まれたときです。

HA分析：HANAクラスタリング・テクノロジーの分析

前節から明らかのように、分散型HANAシステムのHAサポートは実に複雑ですが、これは主としてSAPがシェアード・ナッシング・アプローチを基盤データに採用しているためです。データを保持する基盤ストレージ・システムはHANAクラスタ内の複数のホストに共有されていることに注意してください。ただし、定常状態では、各ノードのデータ・パーティションへのアクセスは排他的です。これが複雑さの原因です。その概要は次のとおりです。

洞察#1：SAP HANA－HAにあらざる不完全なクラスタリング・テクノロジー



1. 複雑さ

SAP HANAクラスタのシステム・モデルは、ネーム、インデックス、統計の各サーバー・プロセスから構成され、非常に複雑です。これは、Oracle Real Application Clusters (RAC) よりもずっと複雑です。この複雑さの原因は主としてシェアード・ナッシング・アプローチを機能させるのに必要とされるデータのパーティショニングにあります。さらにこの複雑性を高める原因は、こうしたシステム構成要素（たとえば、マスター、スレーブ、ワーカー、スタンバイなどそれぞれに異なる意味合いがある）に関する複数構成と各ノードが担う実際のロールです。大規模なHANAクラスタが1つでもあれば、管理上の悪夢のような問題につながる可能性があります。

2. 非アクティブ－アクティブ

HANAクラスタは、Oracle RACと異なりアクティブ－アクティブのクラスタではありません。旧来のコールドスタンバイのクラスタ・テクノロジーと類似しています。従来のコールドスタンバイのクラスタ・テクノロジーは、N+1トポロジを実装し、N台のアクティブ・ノードと、スタンバイとして待機する1台のノードで構成されます。この構成はよくありません。その理由は、データを分割する上で最適なHAを構成するには、HANAクラスタはN+Mトポロジを実装する必要があるからです。このトポロジではN台のアクティブ・ノードと複数の、たとえばM台の非アクティブ/スタンバイ・ノード（M個のデータ・パーティションに対応する）を、貴重なシステム・リソースを犠牲にして障害発生に備えて待機させることとなります。

3. 許容できない目標復旧時間と不完全なリカバリ・テクノロジー

HANAが採用しているリカバリ・アーキテクチャは数十年前の初歩的なもので、結果としてMTTRが非常に長くなり、今日のミッション・クリティカルな環境では到底受け入れられません。HANAノードに障害が発生すると、スタンバイ・ノードのうちの1台がアクティブ・インデックス・サーバーのロールを引き受けます。このノードは、共有ファイル・システムから障害ノードのセーブポイントとデータベース・ログを読み込み、セーブポイントのデータをメモリ内に再構築してログを再適用することでメモリ内のパーティション・データを他のノードが障害を起こす前の状態に正確に一致させる必要があります。そうして初めてこのノードでアプリケーション接続を受け入れる準備が整います。これは非常に時間がかかるうえ処理の間中、対応するデータ・パーティションはオフラインとなり、深刻なアプリケーションの停止時間につながります。RACではこうした問題は皆無です。すなわち、ノードが停止してもアプリケーションは存続ノードのいずれかに再接続して処理を継続できます。

4. OLTPに不向きなシェアード・ナッシング・モデルによる非効率性

HANAは、シェアード・ナッシング・パーティション化モデルによる欠陥に昔から悩まされています。アプリケーション自体がパーティションを認識しない限り（アプリケーション設計としてはいずれにせよ、お粗末だが）、データのパーティショニングを機能させるためには、データ・アクセスをマルチホップ・プロセスにする必要があります。つまり、マスター・ネーム・サーバーは着信接続を検索し、対応するデータ・パーティションを管理する適切なインデックス・サーバーに転送しなければなりません。これによりデータ・アクセスの遅延が増大し、負荷分散もうまく機能しません。特にローカルのデータ・セットにアクセスが集中している場合はなおさらです。このためHANAはほとんどのOLTPワークロードに対しては適切な選択肢とならず、小規模トランザクションでは過大な負荷がかかります。これとは対照的にRACの場合、アプリケーション要求はサービスを実行する任意のノードに転送され、データ・アクセスを接続時間と実行時間の両面から効率的に負荷分散できます。

5. 制約の多さ

HANAクラスタのHAトポロジは、非常に多くの制約があります。SAP HANAのドキュメント[1]によると、"注意：フェイルオーバーに関するホストのロールは、通常インストール中に構成されます。...ホストのロール構成を切り替えることはできませんが、互いに関連するワーカー・ホストとスタンバイ・ホストの台数は増減できません。"とあります。これは重大な制限です。つまり、3台のワーカー・ノード（N1、N2、N3）と2台のスタンバイ・ノード（N4、N5）からなる5ノード・システムの場合、たとえばN3とN4のロールを入れ替えることはできませんが、N4をワーカー・ノードに追加することはできません。こうした運用上の制約は、HANAのアーキテクチャにもともと設計上の欠陥があることを示しています。

ホストの追加と削除

HANAクラスタのホストは、ネットワーク構成によって追加できます。このネットワークには公衆ネットワーク（アプリとホスト間の通信用）とプライベート・ネットワーク（ホスト間の通信用）が含まれます。ただし、HANAクラスタにホストを追加することは、クラスタ全体の可用性に重大な影響をもたらします。SAPのドキュメント[1]では次の記載があります。

ホストを追加した時点でシステムを復元できるように、ホストを追加してから完全なデータ・バックアップを新たに実行するまで本番運用は避けるよう推奨しています。

つまり、ホストをHANAクラスタに追加する前にHANAデータベース全体を静止する必要があるということです。このことはHANAのスケールアウトにはかなり大きな停止時間コストがかかることを示しています。

洞察#2：SAP HANA – HAにあらす



スケールアウト中の停止時間

SAPは複数のホストを使ったHANAクラスタの負荷分散によってスケールアウトできるとうたっています。しかし、このスケールアウトに対応したノードを追加する前に、データベース全体を静止してフル・バックアップを取る必要があります。これはアプリケーションの可用性に重大な影響を与えます。これは明らかにOracle Real Application Clusters (RAC) より劣ったソリューションです。Oracle RACにはこうした問題は一切ありません。

ホストを削除する場合も次のように同様の前提条件があります[1]。

システムからホストを削除した後で、データ・バックアップを実行し、ホスト削除時点でデータベースを復元できるようにする必要があります。

とはいえ、これはより複雑な運用となります。たとえば、SAPのドキュメンテーション[1]には次のように記載されています。

「Configure Hosts for Failover Situation」ダイアログ・ボックスで構成済みのロールを変更するおもな理由はホストを削除する準備をすることです。この場合、ネーム・サーバー・ホストの構成済みのロールをスレーブに変更し、インデックス・サーバー・ホストの構成済みのロールはスタンバイに変更してから、ホストのデータベース・インスタンスを停止して、そのホストを削除します。

そのうえ、このホストのインデックス・サーバーの表をシステム内の残りのホスト上のインデックス・サーバーに移動する必要があります。それでようやくホストを削除できます。

このことは、データの再分配とデータの再パーティショニングに関する複雑な問題をもたらします。これは次で説明します。

分散型HANAシステムにおけるデータ[再]分配

SAP HANAはシェアード・ナッシング・モデルを採用しているため、基盤となるデータはHANAクラスタのさまざまなインデックス・サーバー・ノードに分割、分配する必要があります。

パーティショニングは主に次の2つの方法で実施します。

1. 別々のホストで動作するインデックス・サーバーごとに異なる表を割り当てることができます（データベース・パーティショニング）。
2. インデックス・サーバーごとに表の異なる行を格納する方法で表を分割できます（表パーティショニング）。

表パーティショニングに関しては、利用できる方法は、ハッシュ・パーティショニングやレンジまたはリスト値によるパーティショニング、ラウンドロビンなどいろいろあります。また、これらの方法を組み合わせた表パーティショニングもサポートされています。

関連トピックに表の[再]分配があります。分散型HANAシステムで表を作成する場合、表は1台のインデックス・サーバーに割り当てる必要があります。この機能は、表分配と呼ばれています。デフォルトで新しい非パーティション表は、ラウンドロビン手法を使って利用可能なインデックス・サーバー全体に分配されます。たとえば、3台のインデックス・サーバーA、B、C（マスターを含む）が用意されている場合、最初に作成される表はサーバーAに配置され、次はサーバーB、その次はサーバーCといった具合です。特定のインデックス・サーバー上で表やパーティションを作成するよう明示的に指定することも可能です。

表/表パーティションは作成時に特定のホスト上の特定のインデックス・サーバーに割り当て（パーティション化）が行われますが、この割り当てはホストの追加/削除や現在の表分配/パーティショニングが最適でない場合などのさまざまな状況に合わせて変更する必要があります。この機能は、表の再分配と呼ばれています。

SAP自身が、表分配/パーティショニングが次第に非効率になる場合があることを認めています。次は、SAP HANA管理ガイド[1]からの引用です。

- 現在の表分配が最適でないと思われる場合は、表を再分配することが有効なこともあります。
- 本番運用中にインデックス・サーバーに対する表と表パーティションの初期の割り当てが最適でないことが明らかになる場合があります。たとえば、頻繁に結合される表が別々のサーバーに配置されている場合などです。
- 分散型SAP HANAシステムでは、パーティション化された表は、異なるインデックス・サーバー全体に分配されます。各パーティションの配置場所は、表を最初に分割するときに手動で指定することもデータベースによって決定することもできます。この初期パーティショニングが徐々に最適でなくなる可能性があります。たとえば、パーティションが巨大化した場合です。

このため表の再分配/再パーティショニングの実行が定期的に必要なになりますが、特に再パーティショニングはコストのかかる手動操作であることはSAPも認めているところです[1]。

上記のように表のパーティショニング処理が負担となる場合がある理由は次のとおりです。

- 実行時間が長い。巨大な表では最大数時間かかる。
- メモリ消費量がかなり高い。
- 排他ロックを必要とする（許可されるのはSELECT操作のみ）。
- 事前にデルタ・マージを実行する。
- ログにすべてを書き出す（バックアップとリカバリに必要）。

推奨事項—大量のデータを挿入する前に、あるいはデータがまだ少量の間に表を（再）パーティション化します。表がパーティション化されずに表のサイズが構成可能な絶対しきい値に達するか、表が1日当たりある一定割合で増大する場合、システムによってアラートが発行されます。

SAPによると管理者は表の再パーティショニングではなく、表の再分配を実行すべきだとのことです。たとえば、下記のSAPドキュメンテーションからの抜粋[1]をご覧ください。

表の（再）パーティション化や手動によるパーティションのマージも可能ですが、場合によっては、表のパーティショニングを最適化するために提供されている再分配処理を使用する方が効果的なこともあります（たとえば、パーティションの仕様変更が不要な場合など）。再分配処理は複雑なアルゴリズムを使用して現在の分配状態を評価し、その状態に基づいて分配の改善方法を決定します。

この再分配は手動で実行する必要があるが、大量のデータ・セットでは処理に時間がかかるのは避けられません。

注：SAPのドキュメンテーションではこうしたデータの再分配による可用性への影響について、現時点では明らかにしていません。

洞察#3：SAP HANA—HAにあらす コストのかかるデータ再分配を必要とする スケールアウト・アーキテクチャ



SAPがシェアード・ナッシング・アプローチに依存しているということは、基盤となるデータは、個別のインデックス・サーバー全体にパーティション化される必要があり、データ・アクセスの性能を最適化するために、あるいはノードのカーディナリティを変更した後で、定期的にこのデータ・セットを利用可能なノードに再分配しなければならないことを意味しています。大規模なデータ・セットの場合、これは非常に効率が悪く、コストのかかる処理になる可能性があります。一方、Oracle RACでは各ノードがすべてのデータに等しくアクセスできるためこうした制限が一切ありません。ノードは動的に追加/削除が可能で、データ全体の可用性やレイアウトに影響を与えません。

SAP HANA：ディザスタ・リカバリ機能の分析

HANAのディザスタ・リカバリ（DR）対応は2つの機能が中心になっています。

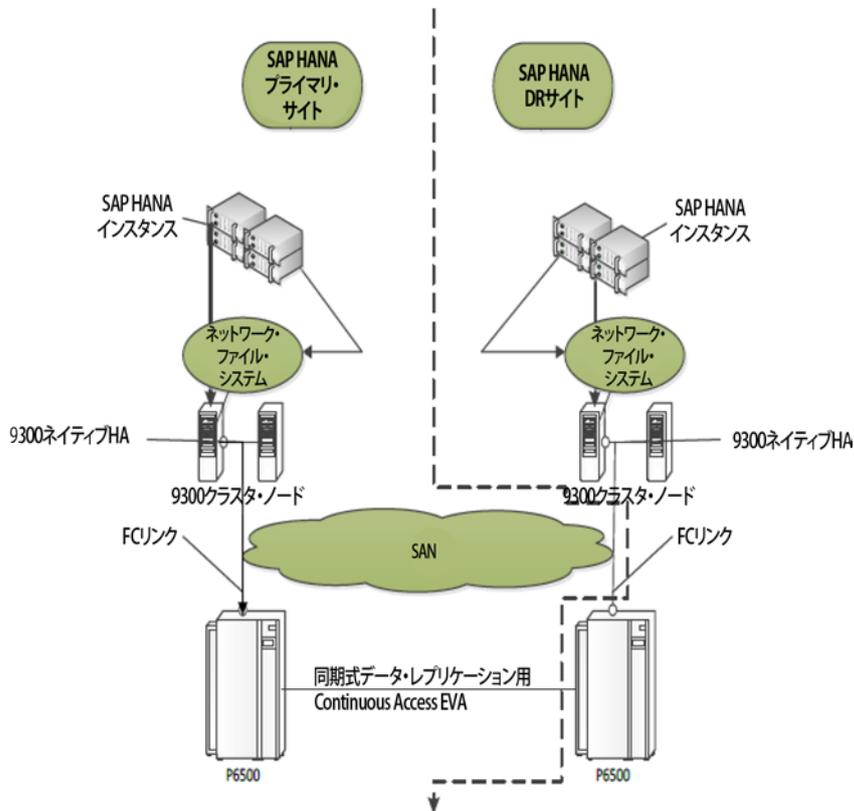
- ストレージ・レプリケーション：プライマリ・ストレージとバックアップ・ストレージ間でネットワーク経由の連続レプリケーション（ミラーリング）機能（おそらく同期式）。このソリューションはストレージ・パートナー（たとえば、HPやHDS）によって提供され、初期のHANAリリースで提供された唯一のDR対応機能でした。
- システム・レプリケーション：プライマリ・システムによるセカンダリ・システムの継続的な（現在：同期式）更新機能（インメモリ表の読み込みを含む）。この機能は、SPS05で初めて搭載されました。

ストレージ・レプリケーション

SAPは特定の認定ハードウェア・パートナーを利用して、ストレージレベルのレプリケーション・ソリューションを提供しています。これにより、永続化ボリュームやファイル・システムのコピーをリモートのネットワーク化されたストレージ・システムに配布できます。同期レプリケーションの場合、SAP HANAのトランザクションが完了するのは、ローカルの永続化されたトランザクション・ログがリモートに複製されたときです。SAPの推奨では、同期ストレージ・レプリケーションが使えるのはプライマリ・サイトとDRサイト間の距離が100km以下の場合だとされています。いずれにせよ、このソリューションでは、プライマリ・サイトとセカンダリ・サイト間に信頼性に優れた高帯域幅で低遅延のネットワーク接続が必要とされます。

この構成のイメージに関しては、例として[11]や[12]を参照ください。これらの参考資料からも明らかのように、ストレージの情報をリモートのストレージ・アレイに反映する間、ミラーリングの動作中にリモート・アレイにサーバーをマウントすることはできません。

ハードウェア・ベンダーのパートナーの中では、HPがHANA向けにDRを提供するハードウェア・ソリューションの検証を重要視してきました[13]。次の図は、HPによるHANA-DR向けソリューションの概要を示しています[14]。



図：6：HPベースのストレージ・ミラーリングを使用したSAP HANA DR

洞察#4：SAP HANA—HAにあらざ 不十分なDRソリューション、DR対応で推奨される ストレージ・ミラーリング



HANAが基盤となるストレージ・ミラーリング・ソリューションをハードウェア・ストレージのパートナーに依存しているということは、HANA独自の組み込みDRソリューションがまだ不完全だということになります。データベース業界では周知のように、ストレージ・ミラーリングに基づくDRソリューションはリスクを抱えています。ミラーリングの実行中はリモートのストレージ・ボリュームを使用できないということは、無駄なDRリソースがあるというだけでなく、ミラーされた情報が有効である保証もないということです。プライマリ・ストレージ・アレイで起きたビット破損は、リモート・アレイにも反映され、両方も使えなくなります。RTOは何時間にもなる可能性があります。フェイルオーバー・プロセスに未検証のストレージ・ボリュームでのサーバーのコールド・スタートが必要だからです。また、まったく異なる技術によるベスト・プラクティスをDRガイドライン全体に統合する必要があるため、管理・運用上の効率性の悪さにつながっています。

システム・レプリケーション

SPS 05で追加されたシステム・レプリケーション[15]によって、SAPはOracle Data Guard相当のネイティブDRソリューションを提供しようしました。とはいえ、留意すべき重大な不具合がいくつかあります。まず基本機能から確認しましょう。

システム・レプリケーションでは、アクティブ・プライマリ・システムの完全なコピーで構成されるセカンダリ・スタンバイ・システムが必要です。このため、プライマリがN'台のワーカー・ホストで構成される場合、セカンダリもN'台のワーカー・ホストが必要になります。これは、スケールアウトするためにはプライマリ・サイトとセカンダリ・サイトに同数のホストを追加する必要があるということも意味しています。プライマリにホストを追加するにはシャットダウンとセカンダリからの登録解除が必要なことに注意してください。iniファイルの構成は両システムで同一でなければなりません。SQLコマンドにより一方のシステムに手動で加えられた変更はすべて、他方のシステムにも手動でコピーする必要があります。

プライマリ・システムの各サービス・インスタンスは、セカンダリ・システムのサービス・インスタンスと通信します。セカンダリ・システムはローカルの場合もあれば、リモートの場合もあります。

セカンダリ・システムのインスタンスは、リカバリ・モードで動作します。このモードでは、すべてのセカンダリ・システム・サービスが常にプライマリ・システム・サービスと通信してデータとログを受信して永続化します。ただし、ログは直ちには適用されません。代わりに、増分データのスナップショットが定期的に時折非同期でプライマリ・システムからセカンダリ・システムに送信されます。セカンダリ・システムに引き継ぐ必要がある場合、適用されるのは最新のデータ・スナップショットを取った後で変更されたログの部分だけです。

スナップショットに加えて、プライマリ・システムはメモリに現在ロードされている表の列に関するステータス情報も送信します。それに応じてセカンダリ・システムは表の列を事前にロードします。それ以外ではセカンダリ・システムはアイドル状態でユーザーからのリクエストや問い合わせには応じません。

システム・レプリケーションでサポートされるレプリケーションには、次のとおりさまざまな手法があります。

- 同期式：ログがいつプライマリ・システムで永続化されても同時にセカンダリ・システムに送信されます。プライマリ・システムは、ログがセカンダリ・システムで永続化されたという返答を受け取って初めてトランザクションをコミットします。つまり、データ損失ゼロのソリューションです。
- インメモリ同期：プライマリ・システムがトランザクションをコミットするのはセカンダリ・システムからログを受け取ったという返答が来た後ですが、永続化はまだされていません。つまり、前記の同期式の場合よりプライマリ・システムでのトランザクションの遅延が短いということです。データ転送の時間がかかるだけでディスクI/Oの待ち時間がないからです。このオプションで性能は向上しますが、データ損失に対しては若干弱くなります。
- 非同期式（SP06で追加）：プライマリ・システムは、REDOログ・バッファをセカンダリ・システムに非同期に送信します。つまり、プライマリ・システムは自身のログ・ファイルに書き込んだ時点でトランザクションをコミットし、ネットワーク経由でセカンダリ・システムにログ・ファイルを送信するということです。つまり、セカンダリ・システムによる確認を待ちません。このオプションで性能は向上します。セカンダリ・システムのログI/Oの完了を待つ必要がないからです。ただし、データ損失に対しては弱くなります。

セカンダリ・システムへの接続が失われるかセカンダリ・システムがクラッシュすると、プライマ

リ・システムはタイムアウト後、処理を再開します。

システム全体のフェイルオーバーを必要とするような障害が発生した場合、管理者はセカンダリ・システムに指示してリカバリ・モードから本稼働に切り替えます。セカンダリ・システムをプライマリ・システムにするために最後のトランザクション・ログを再生し、問い合わせの受付を開始します。

引き継ぎ処理が完了し、データセンターが運用モードに戻ったら、旧プライマリをアクティブ・プライマリ・システムと共にセカンダリとして直ちに登録する必要があります。旧プライマリは完全に初期化してからでないとセカンダリとして使用できません。管理ガイド[1]を参照してください。

これは、『Configuring the Secondary System』に記載された通常のセカンダリの設定で使用される手順と同じです。すでに永続化されているデータは再利用できないため、完全に初期化します。つまり、“旧プライマリ”が再度同期されるまでフル・レプリカが送信されます。

クライアント・ライブラリでフェイルオーバーをサポートするために、管理者はプライマリ・システムとセカンダリ・システムのホスト名一覧を用意する必要があります。あるホストが利用できない場合、クライアントはタイムアウトすると一覧から次のホストを利用します。利用できるホストが1台もない場合は、接続エラーが生成されます。

HA分析：HANAのDR機能を分析

HANAのDR対応は、かなり初歩的でOracle Data Guardと比べると見劣りします。

洞察#5：SAP HANA－HAにあらず 初歩的なディザスタ・リカバリ機能



1. 柔軟性の欠如

プライマリ・システムがN'台のワーカー・ホストからなる場合、セカンダリ・システムにもN'台のワーカー・ホストが必要です。同様にN'台のワーカー・ホストをプライマリに追加する必要がある場合は、同数をセカンダリにも追加しなければなりません。それだけでなく、プライマリにホストを追加する間、セカンダリ・システムはシャットダウンする必要があります（DR保護機能がないことを示しています）。Data Guardにはこのような制限はありません。

2. DRリソースの無駄

Active Data Guardと異なり、セカンダリ・システムはリカバリ・モードでしか動作できません。セカンダリ・システムは、リカバリ・プロセスが停止した後であっても、読取り専用でオープンすることすらできません。当然のことながら、スタンバイを開発やテストに利用できるData Guardのスナップショット・スタンバイのような機能はありません。セカンダリ・システムはバックアップにも使えません。セカンダリ・システムが実際に動作する保証もありません。フェイルオーバーしてみなければ分からないのですが、それでは遅すぎるでしょう。システム・レプリケーションはDR機能に関してData Guardより10年以上遅れています。

3. 非同期スタンバイのためのデータ損失ゼロの非サポート

システム・レプリケーションの非同期スタンバイでは、スタンバイにフェイルオーバーした場合、データ損失が発生する可能性があります。Data Guardではこのデータ損失のリスクを排除するため、Oracle Database 12cでData Guard Far Syncを利用しています。SAP HANAにはこうした機能がありません。

4. 許容できないRTO

Data Guardのファスト・スタート・フェイルオーバーのような自動フェイルオーバー機能はサポートされていません。フェイルオーバーにかかる時間もログの適用が保留されているため遅延が生じます（セカンダリ・システムではログを継続的に適用しないためです）。さらにシステム・レプリケーションのクライアント・フェイルオーバーでは、ホスト名の一覧に対する反復操作が必要になります。ホストが応答不能の場合、クライアントはTCP/IPのタイムアウトで停止します。OracleのFast Application Notification (FAN)のように自動的にクライアントに通知する機能がありません。

5. 基本機能に限定されるロール移行

システム・レプリケーションでは、Data Guardのフェイルオーバーに似た機能しかサポートされていません。データ損失ゼロを保証し、システムのアップグレードや更新プロジェクトで停止時間を最小化するのに極めて有効なData Guardのスイッチオーバーのような機能はサポートされていません。そればかりか、引き継ぎ終了後、システム・レプリケーションでは旧プライマリはバックアップを使って完全に初期化する必要があります。これは時間やネットワークを消費する作業で、初期化中は新しいプライマリは保護されません。

6. DRの複数の中核機能が欠如

システム・レプリケーションはDR機能全般に関して、Oracle Data Guardより少なくとも10年は遅れています。Data GuardのFar Sync、ローリング・アップグレード、Reader Farm、ネットワーク圧縮、破損ブロックの自動修復、書き込み損失に対する保護、複数スタンバイ、柔軟なカスケード・スタンバイといった機能によって、Data GuardはDRの標準ソリューションとなっています。HANAのシステム・レプリケーションは実質的には初歩的なディスク・バックアップ・ソリューションで、限られた距離内で異なるストレージ・アレイにディスクを配置しただけです。

SAP HANA：バックアップとリカバリ機能の分析

SAP HANAはインメモリ・テクノロジーを採用しつつ、2種類のデータをストレージに完全に永続化しています。すなわち、トランザクション・ログとセーブポイントという形のデータ変更です（[1]、[16]）。

トランザクションREDOログはコミット実行時の変更を記録するのに使用されます。システムが停止すると、データベースを最新の一貫性のある状態にリストアするために、セーブポイントを復元してログに記録された変更を再生します。完了したトランザクションは再実行し、未完了のトランザクションはロール・バックします。

セーブポイントで、変更されたデータはすべてストレージにページ形式で書き込まれます（たとえば、ブロック）。システムが起動されると、最新のセーブポイントからログを処理できます。セーブポイントは、データベースのすべてのプロセスとインスタンスに対して管理され、トランザクションの一貫性を保証します。デフォルトでセーブポイントは5分ごとに実行されますが、変更も可能です。

HANAでは次の2種類のバックアップがサポートされていますが、バックアップを実行できるのは、どちらもデータベースがオンラインのときのみです。

- フル・データ・バックアップ形式のセーブポイントのバックアップ。データベースを特定の時点にリストアするのに使用できます。
- 致命的なストレージ障害からリカバリするための定期的な小規模ログ・バックアップ。データ損失を最小化します。

また、HANAではサード・パーティのバックアップ製品もサポートするためのバックアップ・インターフェースが公開されています。*Backint for SAP HANA Interface*と呼ばれ、現在のサポート対象はLinuxのみです。サード・パーティのツールは、バックアップとリカバリを実現するために、このインターフェースを使ってHANAと通信する必要があります。分散型HANAシステムをバックアップするために、複数のBackint通信プロセスが同時に動作している場合もあります。こうしたサード・パーティ製のツールで指定できるバックアップ先の場所は、固定されており（データ・バックアップもログ・バックアップも）、変更できません。このバックアップ先は実際には最終的にバックアップが書き込まれる場所を指定するパイプとして機能します。

リカバリでは、データ・バックアップ、ログ・バックアップおよびログ領域が使用されます。リカバリを開始するとき、使用するすべてのデータ・バックアップとログ・バックアップはファイル・システムでアクセス可能か、サード・パーティのバックアップ・ツールで利用可能かのどちらかが必要です。データ・バックアップから問題なくリカバリされたら、ログ・バックアップとログ領域からログ・エントリが自動的に再生されます。データベースを特定時点にリカバリするには、リカバリ時点までのデータ・バックアップとすべてのログ・バックアップが必要です。データベースのリカバリはいったん開始したら中断できません。リカバリが失敗した場合は、完全リカバリを繰り返す必要があります。

HANAが使用するバックアップ・カタログは、HANAデータベース自体に保存され、ログ・バックアップの一部としてバックアップとバージョン管理が行われます。バックアップ・カタログには次のようなさまざまな情報が含まれます。

- データベースに対して実行されるバックアップ
- バックアップの開始時間と終了時間
- バックアップが動作中かどうか
- バックアップが成功したかどうか
- バックアップされたボリューム
- ログ・バックアップとログのどの部分が含まれているか
- バックアップ先とそのサイズ
- 宛先の種類（ファイルかBackintか）
- 外部バックアップID（サード・パーティのバックアップ・ツールを使用している場合）

こうした情報を利用したバックアップ・カタログの機能は次のとおりです。

- リカバリが可能かどうかの判断
- データベースのリカバリにどのデータ・バックアップとログ・バックアップを使用するか
の判断
- どのバックアップが不要かの判断

HA分析：HANAのバックアップとリカバリ機能を分析

DR機能同様、HANAのバックアップとリカバリ機能のサポートもごく基本的で、Oracle RMANとOracle Secure Backupに大きく遅れを取っています。



洞察#6：SAP HANA－HAにあらざ 非常に基本的なバックアップ機能とリカバリ機能

1. 必要最小限－フル・バックアップのみ

HANAのバックアップ機能とリカバリ機能は実のところ必要最小限です。基本的にサポートしているのはフル・バックアップ（セーブポイントによって）とREDOログのバックアップのみで、自社製またはサード・パーティのバックアップ・ツールを使ってディスクを利用します。つまり、データ・バックアップはすべて時間とネットワーク資源を消費するフル・バックアップだということです。増分バックアップはサポートされていません。最新をうたっている技術が増分バックアップをサポートしていないなどまったく論外です。

2. データ・ファイルやログ単位のきめ細かい処理は不可

HANAの場合、バックアップとリカバリは常にデータベース全体に適用されます。個別のデータベース・オブジェクトのバックアップやリカバリはできません。たとえば、RMANと違ってHANAは個別のデータ・ファイルや個別の表をリカバリすることはできません。

3. テープ・バックアップの非サポート/限定的なメディア管理サポート

RMANとOracle Secure Backupの組み合わせと異なり、HANAでは指定されたディスクの宛先にしかバックアップできません。テープにバックアップするには、サード・パーティのバックアップ製品を使う必要がありますが、Linuxしかサポートされていません。

4. バックアップ・カタログの設計の不備－重大なデータ損失のリスク

RMANと異なり、HANAのバックアップ・カタログはHANAデータベース本体内に配置されています。つまり、HANAデータベースが破損したり破壊されたりすると関連するバックアップ情報がすべて失われ、既存のバックアップを取り込めないため、管理者は最後の有効なカタログのコピーを使わざるを得ず深刻なデータ損失につながる可能性があります。SAP HANA管理ガイド[1]には、バックアップ・カタログが失われた場合に実行可能なデータベース復旧シナリオについての説明がないことに注意してください。

5. 非効率的なリカバリ・プロセス

HANAの場合、データベースのリカバリ・プロセスはいったん開始すると中断できません。リカバリが失敗した場合は、完全リカバリを繰り返す必要があります。これは極めて非効率的です。大規模データベースの場合はなおさらです。HANAと異なり、Oracle RMANベースのリストアはいつでも中断できるように最適化されており、再開するときは前回リストアされていないファイルをリストアするだけで済みます。これはOracleのリカバリ・プロセスにも当てはまります（Oracle version 7から利用可能です）。

6. エンタープライズレベルのバックアップ機能とリカバリ機能が複数欠落

HANAのバックアップ機能とリカバリ機能は、Oracle RMANとOracle Secure Backupより少なくとも10年は遅れています。明らかに欠落している機能には、バックアップの暗号化、バックアップの圧縮、ブロックのリカバリ、増分更新バックアップ、バックアップの最適化、パラレル・バックアップ・チャンネル、メディア管理の統合があります。HANAが初歩的なバックアップ機能とリカバリ機能しかサポートしていないことは、HANA自体がエンタープライズレベルとは言えないことを示しています。

SAP HANA：全HA機能の概要分析

スケーラビリティ、ディザスタ・リカバリ、バックアップとリカバリのサポート以外にも、包括的なHAソリューションでは、計画外停止時間と計画停止時間に関するさまざまなケースに対処する他の諸機能のサポートも必要とされます。たとえば、人為的エラーは停止時間の最大の原因の1つです。管理者が誤って重要なデータベースの表や重要なデータベース・ファイルを削除してしまった場合のために、包括的なHAソリューションには、こうしたエラーを効率的に取り消す機能が搭載されている必要があります。SAP HANAにはこうした機能がありません。同様に、計画停止時間が生じるさまざまなケース（たとえば、ソフトウェアやハードウェアのアップグレード、テクノロジー更新など）を減らす機能もサポートする必要があります。ところが、HANAでは何1つサポートされていません。

一方、Oracle Databaseに統合されたOracle Maximum Availability Architecture (MAA) では、ユーザーに包括的な業界最高の統合HAテクノロジーが提供されます。このテクノロジーは最小コストで導入でき、停止時間を回避して、計画外の障害からの迅速なリカバ리를可能にし、さらに計画停止の影響を最小限に抑えます。

次の一覧は、Oracle MAAを構成する重要なHA機能とそれにより得られるメリットおよび、SAP HANAの同等ソリューションとの比較です。

洞察#7：SAP HANA－HAにあらず Oracle Maximum Availability Architecture (MAA) の持つ 幅広さと奥深さが欠如

Oracle MAA の機能	メリット	HANA の対応機能	説明
Oracle RAC	アクティブ・アクティブ・クラスタリングとサーバーHAによるスケーラビリティ	HANA 分散システム	従来のコールド・クラスタ・テクノロジーに基づいており、データの再分配コストが高い
アプリケーション継続性	存続ノードでアプリケーション・コンテキストを自動再生	該当なし	
<ul style="list-style-type: none"> フラッシュバック・テクノロジー フラッシュバック問合せ フラッシュバック・バージョン問合せ フラッシュバック・トランザクション問合せ フラッシュバック・トランザクション フラッシュバック表 フラッシュバック・ドロップ フラッシュバック・データベース 	人為的エラーの迅速かつ効率的な取り消し	該当なし	
データベース・ブロック・チェック、データベース・ブロック・チェックサム、書き込み損失に対する保護、自動ブロック修復	データ破損に対する統合保護機能	該当なし	
Automatic Storage Management (ASM)	統合ボリューム管理機能、データベース用に最適化されたディスク障害からの保護機能	該当なし	
Recovery Manager (RMAN)	統合されたバックアップとリカバリ	HANA バックアップとリカバリ	基本的なバックアップとリカバリ対応
Oracle Secure Backup	テープ・バックアップ・ソリューション	該当なし	
Oracle Active Data Guard	包括的なデータ保護とディザスタ・リカバリおよびデータの可用性	HANA システム・レプリケーション	基本的なディザスタ・リカバリ対応
Oracle GoldenGate	柔軟なアクティブ・アクティブ・レプリケーション	該当なし	
Oracle Site Guard	サイトとシステム・スタック全体にわたるフェイルオーバー・オーケストレーション	該当なし	
Global Data Services	ローカル・データセンターとリモート・データセンターにまたがるアプリケーション・サービスのフェイルオーバーと負荷分散	該当なし	
オンライン・システム再構成	システム更新（ハードウェア、ネットワーク、ストレージ、データセンターなど）のための計画停止時間を短縮	制限付き	制限付きサポート たとえば、ホストを追加するには本番運用を停止する必要あり
ローリング・アップグレード (RAC、Data Guard、GoldenGate)	ソフトウェア・アップグレードのための計画停止時間を短縮	制限付き	
エディションベースの再定義	オンラインによるアプリケーションのアップグレード	該当なし	
オンライン表再定義とオンライン操作	アプリケーションを停止させずにデータベース構造を変更	該当なし	
Maximum Availability Architecture (MAA)	HA ベスト・プラクティスの統合セット		該当なし

顧客事例

オラクルのHAソリューションは、幅広い顧客に採用され市場での目覚ましい成功の結果、今日のビジネスに必要な24時間365日のアップタイム要件を満足できるデータベース・テクノロジーを選択する場合の、引き続き重要な差別化要因になっています。Oracle MAAは、世界中の何千もの極めて重要な顧客のサイトに導入され、大手の銀行や小売店、メーカー、通信会社などに関連した世界的な経済活動を支えています。

さまざまなオラクルのHAソリューションを導入した顧客を集めたリストと詳細な事例研究は、Oracle Technology Networkの公開サイト[17]で入手できます。世界中のさまざまな業界の有名企業で稼働しているOracle MAAのこうした成功事例は、高可用性の分野でオラクルが比類ない技術的優位性を有していることを証明しています。

もう1つの事実、この状況ではかなり皮肉なことですが、IDCの調査報告[18]での指摘によると、SAPのERPユーザーの80%以上がOracle Databaseを使っているということです。

これに対し、SAP HANAのHAに特化した顧客の事例に関する情報は、非常に限られています。たとえば、RTO（目標復旧時間）はHAの導入では必ず検討すべき重要事項ですが、公開されている入手可能などの資料を見ても、SAP HANAが災害発生時にスタンバイ・システムに高速フェイルオーバー可能であることや事業継続性を維持できることを明らかにしたものはありません。これはインテルの報告[19]にあるOracle Data Guardのフェイルオーバー時間と対照的でしょう。それは、実にたったの2秒から30秒なのですから。

高可用性：留意事項

データベースは企業の心臓部です。データベースに停止時間が発生した場合、それが不測の障害によるものであっても、なんらかの計画メンテナンス作業によるものであっても、企業にとってはたいてい良くない結果となります。商売を逃し、評判を落とし、こうした停止時間に起因するインシデントは、全国的なニュースとなります。しかも悪い形で。

HANAとは対照的に、Oracle MAAはHAをベースに十二分に完成された確固たる製品アーキテクチャです。お客様からのフィードバックと10年以上にわたってオラクルが積んできた経験に基づいています。アーキテクチャ自体も強固ですが、MAAは数々の可用性に関連するまれに発生する問題にも対処してきました。こうした問題に関する知識は10年に及ぶ経験を通じてしか獲得できないものです。

どんな現実的なビジネスにおいても主要なITの重点分野の1つである可用性を求めるとすればHANAの採用は無謀極まりないことです。

結論

成功した企業は、重要なデータや情報システムを保護するために、高可用性テクノロジー・インフラストラクチャを維持管理することが極めて重要であることを理解しています。多くのミッション・クリティカルな情報システムを中心となっているのは、ITインフラストラクチャの可用性、セキュリティ、信頼性を担うデータベース・システムです。こうした基本的な機能のいずれかが欠けているデータベース・システムは、断じてエンタープライズレベルとは言えません。たとえ他に機能がいくつもあつたとしてもです。

数十年にわたる技術革新を経て誕生したOracle Databaseは、高可用性とデータ保護の統合において業界を代表する製品となっており、計画保守作業と不測の障害発生時の両方でユーザーがアプリケーションの稼働時間を最大化することを確実にします。一方、このホワイト・ペーパーで明らかにしたように、SAP HANAは次世代データベース・アーキテクチャをうたいながらも、今日の24時間365日事業を運営するのに必要とされる基本的な高可用性機能の多くが欠落しています。

最後の洞察：SAP HANA – HAにあらず SAP HANAは、エンタープライズレベルとは言えません。



SAP HANAにはごく初歩的なHA機能しかないため、今日の24時間365日稼働するビジネスの可用性要求を到底満たすことはできません。SAP HANAはエンタープライズ規模で導入することは不可能です。

参考資料

1. SAP HANA管理者ガイド：http://help.sap.com/hana/SAP_HANA_Administration_Guide_en.pdf
2. SAP HANAドキュメンテーション：http://help.sap.com/hana_appliance/
3. Oracle Databaseの高可用性：
<http://www.oracle.com/technetwork/jp/database/availability/index.html>
4. Oracle Databaseインメモリ・オプション：
<http://www.oracle.com/jp/corporate/features/database-in-memory-option/index.html>
5. IBM Redbook：『*In-memory Computing with SAP HANA on IBM eX5 Systems*』、
<http://www.redbooks.ibm.com/redbooks/pdfs/sg248086.pdf>
6. SAP補足資料：データセンターにおけるSAP HANA、スライド：『SAP HANA Persistence』 -
[http://www.saphana.com/servlet/JiveServlet/download/2010-9-12161/SAP%20HANA%20in%20Data%20Centers%20\(SPS6%2B\).pdf](http://www.saphana.com/servlet/JiveServlet/download/2010-9-12161/SAP%20HANA%20in%20Data%20Centers%20(SPS6%2B).pdf)
7. SAP補足資料：データセンターにおけるSAP HANA、スライド：『Scale Out』 -
[http://www.saphana.com/servlet/JiveServlet/download/2010-9-12161/SAP%20HANA%20in%20Data%20Centers%20\(SPS6%2B\).pdf](http://www.saphana.com/servlet/JiveServlet/download/2010-9-12161/SAP%20HANA%20in%20Data%20Centers%20(SPS6%2B).pdf)
8. SAP補足資料：データセンターにおけるSAP HANA、スライド：『SAP HANA Scalability』 -
[http://www.saphana.com/servlet/JiveServlet/download/2010-9-12161/SAP%20HANA%20in%20Data%20Centers%20\(SPS6%2B\).pdf](http://www.saphana.com/servlet/JiveServlet/download/2010-9-12161/SAP%20HANA%20in%20Data%20Centers%20(SPS6%2B).pdf)
9. ブログ：『SAP HANA – Core Architecture』、
<http://en.community.dell.com/techcenter/b/techcenter/archive/2012/09/28/sap-hana-core-architecture.aspx>
10. SAP HANA管理者ガイド：『*Monitoring Host Status and Auto-Failover Configuration*』、
http://help.sap.com/hana/SAP_HANA_Administration_Guide_en.pdf
11. SAP補足資料：データセンターにおけるSAP HANA、
スライド：『SAP HANA Disaster Recovery:Storage Replication』 -
[http://www.saphana.com/servlet/JiveServlet/downloadBody/4351-102-2-9243/SAP%20HANA%20in%20Data%20Centers%20\(SPS7%2B%20IFG\).pdf](http://www.saphana.com/servlet/JiveServlet/downloadBody/4351-102-2-9243/SAP%20HANA%20in%20Data%20Centers%20(SPS7%2B%20IFG).pdf)
12. SAP補足資料：『SAP HANA – High Availability, "Storage Replication"節』、
http://www.saphana.com/servlet/JiveServlet/downloadBody/2775-102-4-9467/HANA_HA_2.1.pdf
13. 『HP AppSystems for SAP HANA』、
<http://h17007.www1.hp.com/us/en/converged-infrastructure/converged-systems/appsystems/sap-hana.aspx>
14. HP補足資料：『*Disaster-Tolerant solution for HP AppSystems for SAP HANA*』、
<http://h20195.www2.hp.com/V2/GetPDF.aspx%2F4AA1-4846ENW.pdf>
15. SAP補足資料：『SAP HANA – High Availability, "System Replication"節』、
http://www.saphana.com/servlet/JiveServlet/downloadBody/2775-102-4-9467/HANA_HA_2.1.pdf

-
16. SAP補足資料：『SAP HANA – High Availability, "Backups"節』、
http://www.saphana.com/servlet/JiveServlet/downloadBody/2775-102-4-9467/HANA_HA_2.1.pdf
 17. Oracle HAのお客様事例：
<http://www.oracle.com/technetwork/database/features/ha-casestudies-098033.html>
 18. 『IDC Insight:SAP and Sybase:A Marriage Made In Database Heaven, June 2011』、
http://www.sybase.com/files/White_Papers/IDC_SAP_Sybase_DatabaseHeaven_report.pdf
 19. 『High Availability & Disaster Recovery with Oracle Data Guard Fast-Start Failover – An Implementation at Intel』、Oracle OpenWorld 2011、
<http://www.oracle.com/technetwork/database/features/availability/13441-intel-515449.pdf>



SAP HANA : HA機能の分析
(2014年2月)

著者 : Oracle HA開発

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

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

oracle.com



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

Copyright © 2014, 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、記載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や条件を含めて、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複製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。0113

Hardware and Software, Engineered to Work Together