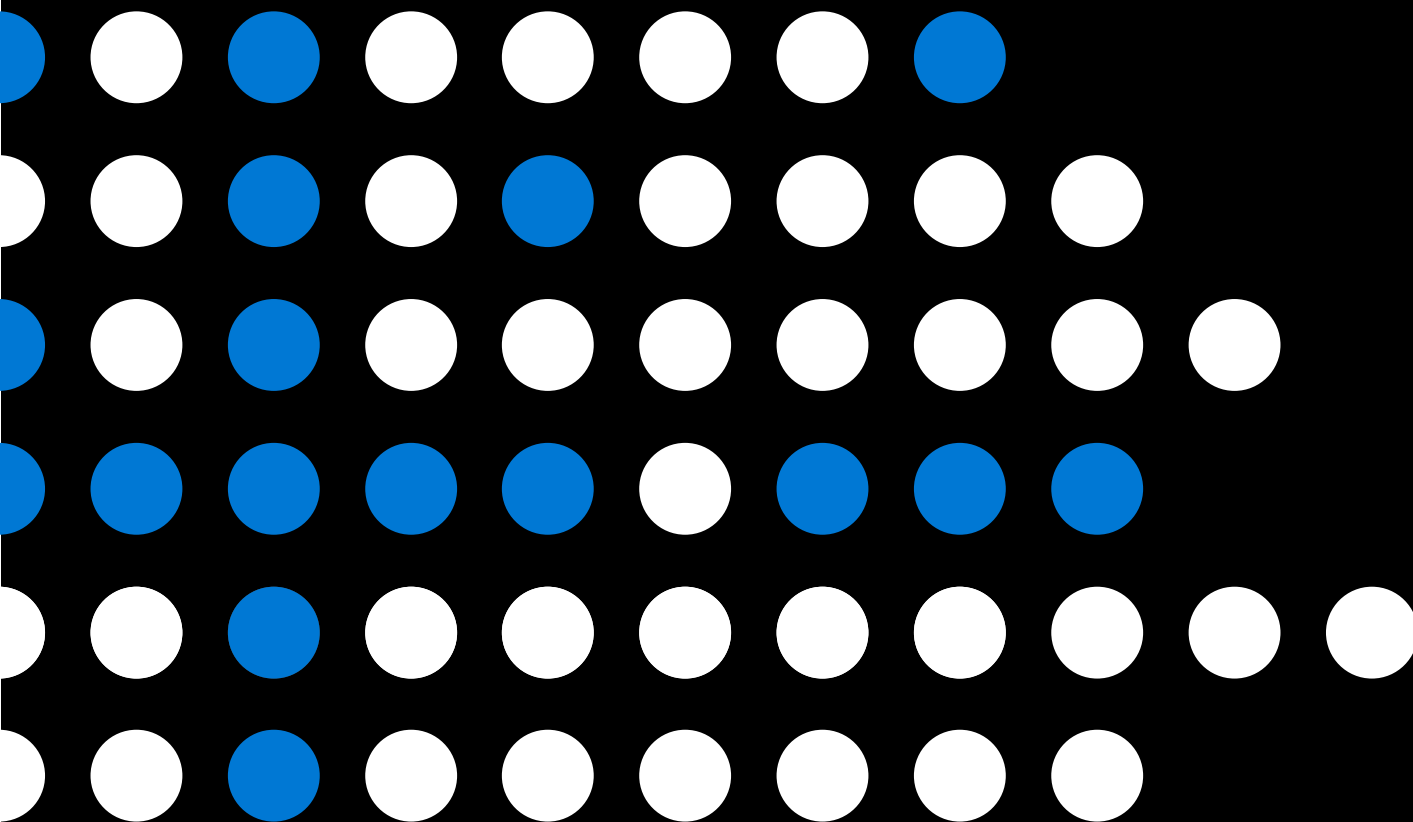
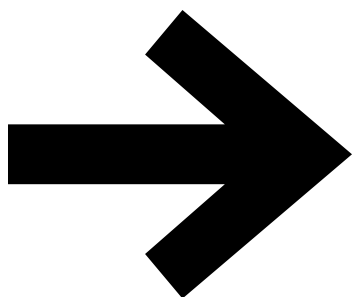


SQL Server on Linux: データ ワークロードのプラットフォーム フォーム移行とモダナイ ゼーションに関するガイド





01 /

SQL Server on Linux の概要

02 /

SQL Server on Linux を
最大限に活用する

03 /

ツールと管理

04 /

移行とアップグレードのた
めのツールとサービス

05 /

まとめ

本書の対象読者

この e-Book は、データ プラットフォームの柔軟性を高め、パフォーマンスを向上させ、セキュリティを強化することを目指すデータベース・アーキテクト、管理者、および開発者を対象としています。この e-Book を読むことで、Windows や Linux だけでなくコンテナなど、自分の選択したプラットフォームで SQL Server の最新機能を活用するために SQL Server on Linux をどう利用できるかを学ぶことができます。この e-Book では、SQL Server をインストールするための Linux システムの準備、混在環境の管理、既存のデータやデータベースを SQL Server on Linux に移行する作業について、実際の実行手順を、関係する技術的詳細と共に説明します。

SQL Server on Linux の概要

SQL Server on Linux の登場で、データと分析の責任者はさまざまなオペレーティングシステムを選択できるようになり、数多くの使用用途で、ライバル製品ではなく SQL Server on Linux を選択することが多くなっていくと予想されます。

イノベーションとモダナイゼーションは、データ プラットフォームから始まります。同時に、データプラットフォームが革新されてさまざまなものができる、自分の作業に最適なプラットフォームを特定するのが難しくなる可能性があります。お客様は、データ資産から最大の効果が得られるようにするために、プラットフォーム、プログラミング言語、およびインフラストラクチャの選択に関してはますますの柔軟性を求めるようになっていきます。データ プラットフォームについての柔軟性があれば、任意のオペレーティング システムおよびクラウド上で、任意のデータおよび言語でインテリジェントなアプリケーションを構築できます。しかし、この目標を達成するには、多様なデータと関連するニーズを管理するという問題を解決できる最新のデータ資産ソリューションが必要です。

マイクロソフトでは、実際の環境に合わせて最適なオペレーティング システム、または複数のオペレーティング システムの最適な組み合わせを (環境の優先オペレーティング システムがオープン ソース、プロプライエタリ、またはその混在のどれであるかにかかわらず) 選択できる、高品質のエンタープライズ レベル データベース プラットフォームを提供することに取り組んでいます。以前は Windows のみで使用されていた SQL Server のデプロイメント オプションが、Linux およびコンテナのプラットフォームでも利用できるようになりました。

出典 : [Microsoft DBMS バイヤーが SQL Server on Linux を利用する 4 つの方法 \(Gartner\)。](#)

SQL Server 2017 では、お客様が選択されたプラットフォーム上で SQL Server を進化させる取り組みを継続しており、新機能や、さらに多くの Linux ディストリビューション (Red Hat Enterprise Linux (RHEL)、Ubuntu、および SUSE Enterprise Linux (SLES) など) およびコンテナプラットフォーム (Docker や Kubernetes など) のサポートを通じてデータ管理とデータ主導アプリケーションを改善および機能強化しています。

企業は、自社業務に最も適したプラットフォーム (または複数プラットフォームの組み合わせ) 上にデプロイすることができます。プログラミング言語に関しても同様です。Linux 上で動作する Oracle、PostgreSQL、DB2、Sybase などのシステムを使用する企業は、業界をリードするリレーショナル データベース管理システム (RDBMS) である SQL Server 2017 に移行して、企業で選択した任意のオペレーティング システムを使用することができ、Windows Server オペレーティング システムをインストールする必要はありません。C#、Java、Ruby、その他の言語などを使い慣れたエンジニアは、SQL Server 2017 を使用することにより、作業に最適な言語およびプラットフォームを使用してアプリを構築し、ニーズに応じた任意のクラウド プロバイダーまたはオンプレミス サーバー上でアプリをホストすることができます。

SQL Server 2017 は、さまざまなデータソースに由来するリレーショナルおよび

非リレーショナル データ、グラフ データを処理し、オープンソース アプリケーションを含むあらゆる種類のアプリケーションを実行するように設計されています。この e-Book では、この RDBMS の運用面の概要を示します。SQL Server の機能を活用することにより、オープンソース開発者や IT スペシャリストが求める形で強力なインサイトを提供することに重点が置かれています。SQL Server 2017 は、アドバンスド アナリティクスやエンドツーエンドのビジネス インテリジェンスなどの今では標準となった機能に加えて、さまざまなワークロードを通じたインメモリのパフォーマンス、ミッションクリティカルな高可用性、比類のないセキュリティ機能、保存されているデータと流れているデータを保護するための GDPR、PCI、HIPAA などのプライバシー規制へのコンプライアンスを備えており、このすべてを、お客様が選んだ言語とプラットフォームでご利用いただけます。

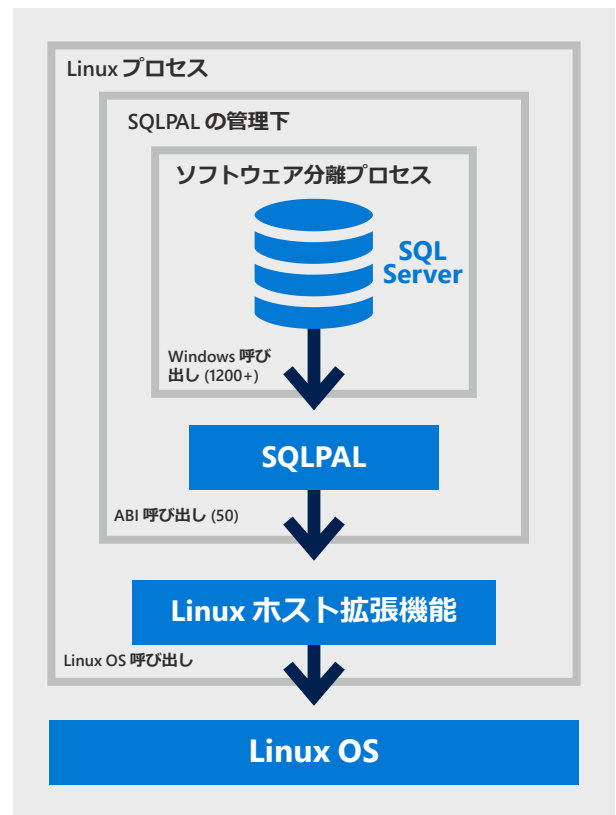
豊富なプラットフォームの選択肢

SQL Server on Windows および SQL Server on Linux は、共通のコードベースを共有しています。つまり、SQL Server のコア エンジンには、Linux 上で機能させるために加えられた変更はありません。SQL Server には、SQL Server と基礎となるオペレーティング システムとの間での呼び出しと通信の抽象化を担当するプラットフォーム抽象化レイヤー (SQLPAL) が導入されています。ホスト拡張機能は、単なる 1 つのネイティブ Linux アプリケーションです。低

レベルのオペレーティング システム機能は、入出力 (I/O)、メモリ、および CPU 使用率を最適化するためのネイティブ コールです。ホスト拡張機能が開始されると、SQLPAL がロードされて初期化され、それにより SQL Server が起動します。SQLPAL 起動ソフトウェアは、単なるスレッドと割り当ての集合であるプロセスを分離し、残りのコードに対して必要な変換を提供します。

この新しいレイヤーが SQL Server アーキテクチャに追加されたことで、Windows において SQL Server の強みとなってきたエンタープライズレベルのコア機能と利点を、SQL Server 用にどの OS を使用しても利用できるようになっています。

データ プラットフォームのこの進化に対しては、お客様からのはっきりとした反応がありました。SQL Server を使ったミッション クリティカル データの管理に、お客様は信頼を寄せてくださっています。業界アナリストの反応も好意的なものでした。たとえば、Gartner は、2 年連続でマイクロソフトのことを、あらゆる運用データベース管理システム上で実行するための最も徹底したビジョンと最高の能力を備えた「リーダー」とであると評価しています。¹ Linux とコンテナのサポートが加わった SQL Server 2017 は、どのプラットフォーム上でも、エンタープライズクラスの機能を備えたフル機能の同じデータベースエンジンを提供します。SQL Server 2017 は、Linux 環境でスムーズに実行できます。これにより、既存の SQL Server デプロイメントとの統合が可能になります。



Linux だけではありません。Windows、Linux、および MacOS でサポートされている Docker でも、SQL Server を実行できるようになりました

Linux ベースのコンテナは、Docker を実行している任意の Linux マシン (物理または仮想) にデプロイでき、ホスト オペレーティング システムに変更を加えることなく実行できるようになっています。コ

¹ Gartner 社の全資料は、[マイクロソフト](#)からの依頼により入手可能です。Gartner 社は、調査出版物で言及しているいかなるベンダー、製品、サービスも支持しておらず、最高評価を獲得したベンダーや他の何らかの評価を獲得したベンダーのみを選択するようテクノロジー ユーザーに推奨しているわけではありません。Gartner 社による調査出版物は、Gartner 社の調査組織の意見で構成されており、事実の表明と解釈すべきではありません。Gartner 社は、明示的または暗黙的にかかわらず、商品性や特定目的への適合性の保証を含む、この調査に関するあらゆる保証を放棄しています。

ンテナ サポートにより、Docker Swarm、Red Hat OpenShift、Kubernetes、マイクロソフトの Azure Container Services(AKS)などのコンテナ オーケストレーション ソリューションで SQL Server を実行することができます。SQL Server on Linux 用の管理パックにより、管理者は System Center Operations Manager を使って、ハードウェアからデータベース エンジンのインスタンスや個々のデータベースに至るまで、すべてを監視できます。同じように、Docker Community Edition (CE) for Mac を使うと、開発者は Linux コンテナを MacOS で実行できます。

SQL Server は、Windows コンテナへのインストールもサポートしています。開発プロセスでコンテナを使用することの主なメリットの1つは、さまざまな環境で作業できることです。アプリケーション開発者は、SQL Server をコンテナ内で実行することにより、データベースをアプリケーションに依存するものとして取り扱うことができます。コンテナを使用すると、開発チームは、運用環境と機能的に同一な開発 / テスト環境で作業を行うことができます。さらに、コンテナ オーケストレーション ソリューションは、コンテナのデプロイメントを自動的に管理し、テストおよびデプロイメントの自動化をサポートします。

お客様事例：

Convergent Computing (CCO) は、戦略的なテクノロジー コンサルティング会社です。CCO は Linux と Windows の両方でデータベースを SQL Server 2017 に移行してから 8 か月半もたたないうちに、コストを 80% 削減し、投資回収率 (ROI) を達成しました。

時には
Linux のほうが高速
“SQL Server 2017 on
RHEL は、SQL Server
2016 Enterprise Edition
on Windows のかつて
のナンバーワンであっ
た非クラスター化 TPC-
H@1000GB の結果より
も優れた結果をもたら
します。

- ▶ 6% 高いパフォーマンス
- ▶ 5% 低い価格 / 性能

出典: [マイクロソフト, Red Hat, and HPE
Collaboration Delivers Choice & Value to Enterprise
Customers](#)

業界をリードするパフォーマンス

SQL Server 2017 のパフォーマンス特性は Linux においても同様の効果を示します。SQL Server は、オンライン トランザクション処理 (OLTP) と分析ワークロードの両方に関して、Linux デプロイメントにおいても同じパフォーマンス、場合によってはさらに優れたパフォーマンスを提供します。どちらのデプロイメントにおいても SQL Server の機能は同一であり、コア リレーショナル データベース、列ストアへの投資、およびインメモリ OLTP がすべて有効です。SQL Server 2017 は、SQL Server on RHEL について 1 TB TPC-H ベンチマーク結果 (非クラスター化) の世界記録を有しています。¹SQL Server on Linux は、最も要求の厳しいトランザクション処理アプリケーションの要件を満たすことができ、マイクロソフトはいくつかの企業との密な連携を通じてこれらのメリットを証明してきました。マイクロソフトはまた、Linux 用に SQL Server 2017 データウェアハウス ファストトラック リファレンス アーキテクチャを提供して、企業がエンタープライズ データ ウェアハウス ソリューションを構築するのを支援しています。ファストトラック プログラムは、SQL Server on Windows サーバーまたは SQL Server on Linux サーバーのコア機能を使用することにより、最適化されたパフォーマンスを提供する平衡対称型マルチプロセッシング (SMP) データ ウェアハウスを提供します。

¹TPC. “TPC-H Result Highlights HPE Proliant DL380 Gen9”, 2017 年 3 月。
<http://www.tpc.org/3327>

オープンソースの価格設定

データベースとそこに保存されているデータは、値段が付けられないほど貴重であることは言うまでもありませんが、だからといって、データで作業するために巨額の費用が必要なわけではありません。SQL Server 2017 on Linux を使用すると、競合他社の製品を使用する場合よりも低い総所有コスト (TCO) と、短期間での投資回収 (ROI) を実現できます。SQL Server on Linux には、インメモリ機能、セキュリティ、データ ウェアハウス、高可用性 / 災害復旧 (HADR) など、必要な機能と機能が追加コストなしで組み込まれています。

不要なライセンスの取得やトレーニング時間、ハードウェアのコストを避けることができ、SQL Server への初期投資費用を確実に回収できます。

お客様事例：

オーストラリアのスタートアップ企業である [Ajilius](#) は、PostgreSQL システムを SQL Server on Linux データプラットフォームに移行した結果、開発速度が向上し、コストが削減されました。この新しいプラットフォームは管理が容易で、ホスティング費用は 30% 削減されます。さらに、SQL Server の搭載された機能を使用することにより、クエリのパフォーマンスを向上させ、セキュリティを強化し、可用性を向上させることができます。

SQL Server on Linux のその他のオプション

マイクロソフトは、オープン ソース ソリューションを活用する取り組みを進めており、その一環として、SQL Server を Linux 上でも使用できるようにしました。これは「SQL Server を、オンプレミスでもクラウドでも開発言語、データ型、およびオペレーティング システムの選択肢があるプラットフォームにする」という目標に向かってマイクロソフトが大きく前進したことを意味します。SQL Server 2017 on Linux では、マイクロソフトのサポートと信頼に裏打ちされたさらに多くのオプションをご利用いただけます。これらのオプションの中には、無料で利用できる SQL Server 2017 Express があります。これは搭載されたセキュリティを備えた高速かつインテリジェントなアプリケーションの開発を開始するのに役立ちます。スケーラビリティと高可用性の要件に応じて、SQL Server 2017 の Standard または Enterprise をお選びいただくこともできます。これらのエディションのどれについても、一度の開発で、サイズに関係なく、高度なセキュリティ機能とパフォーマンス機能にわたる SQL Server の一貫したプログラミング サーフエス領域のどこにでもデプロイできます。

“ SQL Server は世界ナンバーワンのデータベースであり、操作や管理が簡単です。市場に出回っている BI ツールの 99% は、容易に SQL Server に統合でき、セキュリティ、高可用性、および災害復旧のための優れた標準を提供します。書籍、フォーラム、コンサルタントなど豊富な資料もあります。現在、SQL Server on Linux は当社の主要プラットフォームです。

- Ron Dunn、Ajilius 社開発主任

SQL Server on Linux を最大限に 活用する

SQL Server 2017 は Windows ベースの同等製品と同じ高品質のエクスペリエンスを提供するように設計されていますが、ビジネス目標を最大限にサポートするよう Linux システムを準備するためにできることがあります。目標がパフォーマンス、高可用性、グラフデータベース、またはセキュリティのどれに関するものであっても、ここに示す計画上の注意点は SQL Server on Linux を最大限に活用するのに役立ちます。

SQL Server on Linux は、コマンドラインから mssql-server パッケージをインストールして mssql-conf のセットアップを実行することによってインストールできます。以下は、RHEL に SQL Server 2017 をインストールするための手順の例です。

```
//Download the Microsoft SQL Server Red Hat repository configuration file sudo curl -o /etc/yum.repos.d/mssql-server.repo https://packages.microsoft.com/config/rhel/7/mssql-server-2017.repo // install SQL Serversudo yum install -y mssql-server

.// run the mssql-conf setup sudo /opt/mssql/bin/mssql-conf setup
```

Ubuntu および SUSE Linux Enterprise Server に SQL Server をインストールする方法については、「[クイックスタートガイド](#)」、「[SQL Server をインストールし、データベースを作成する \(Ubuntu\)](#)」、「[SQL Server をインストールし、データベースを作成する \(SUSE Linux Enterprise Server\)](#)」を参照してください。Docker で SQL Server 2017 コンテナ イメージを実行するには、mssql-server-linux コンテナ イメージを入手して実行する必要があります。そのための方法の詳細については、「[Docker で SQL Server 2017 コンテナ イメージを実行するためのクイックスタートガイド](#)」を参照してください。

パフォーマンスの最適化

SQL Server のパフォーマンスを最適化するため、Linux および Windows に適用されるディスクレイアウト設計にはさまざまなベスト プラクティスがあります。たとえば、ディスク ストライピングを使用すると、実際の環境の中に、ログ、データ、および tempdb ファイル用の追加のデータ ディスクを追加できます。ディスク ストライピングによりこれらのファイルをマウントし、tempdb、トランザクション ログ、およびデータ ファイルを構成して、複数ディスクにわたって I/O のストライプ化を実現できます。ディスク ストライピングにより、1 つの大きなディスクを複数の小さなディスクとして分割し、1 秒あたりの I/O 操作 (IOPS) を最適化できます。関連するデータ ディスクに必要な IOPS 測定値数と帯域幅を分析できます。

```
Use Master;
GO
ALTER DATABASE tempdb MODIFY FILE (NAME = tempdev, FILENAME = '/var/opt/mssql/data/tempdb01/tempdb01.mdf', SIZE = 1024, FILEGROWTH = 8192 KB);
GO
ALTER DATABASE tempdb MODIFY FILE (NAME = templog, FILENAME = '/var/opt/mssql/data/log/tempdb01/templog.ldf', SIZE = 1024KB, FILEGROWTH = 8192KB);
GO
```

SQL Server on Linux に接続するデータベース アプリケーションのパフォーマンスを最大化するのに役立つ、その他のベストプラクティスとインストール後オプションを次に示します。

- ▶ Linux および SQL Scheduling の効率的な動作を維持するため、ALTER SERVER CONFIGURATION コマンドを使用することによって、すべての NUMANODE や CPU に対して PROCESS AFFINITY を設定することをお勧めします。
- ▶ 高性能環境で tempdb が同時実行時に速度低下するリスクを低減するには、[マイクロソフト サポート](#)によるガイドラインに従って、複数の tempdb ファイルを構成します。SQL Server 2016 で導入された自動 tempdb 構成機能には、SQL Server on Linux のインストール時に複数の tempdb ファイルを構成するオプションはありません。しかし、ADD FILE Transact-SQL (T-SQL) コマンドを使用すれば、tempdb ファイルを追加することができます。
- ▶ SQL Server 2017 でインメモリ OLTP を使用する場合は、データベースの互換性レベルが 140 (SQL Server 2017) であることを確認する必要があります。以前のバージョンからアップグレードした場合、互換性レベルが変更されていない可能性があります。

ることに注意してください。使用しているデータベースの互換性レベルを確認するには、次の T-SQL コマンドを使用します。

```
ALTER DATABASE YourDBName
SET COMPATIBILITY_LEVEL = 140;
GO
```

メモリ構成の設定

- ▶ mssql-conf を使用してメモリ制限を設定し、Linux オペレーティングシステムのために十分な空き物理メモリがあることを確認します。

Linux OS の設定

- ▶ マルチノードの NUMA (Non-Uniform Memory Access) インストールでは、NUMA システム上で SQL Server が最大限の効率で動作できるよう、自動 NUMA 負荷分散機能を無効にする必要があります。

```
sysctl -w kernel.numa_balancing=0
```

- ▶ 仮想アドレス空間のカーネル設定値を 256K に変更することもできます (SQL Server のインストールには、既定値の 65K では不十分な場合があるため)。

```
sysctl -w vm.max_map_count=262144
```

- ▶ noatime 属性を使用して、SQL Server のデータとログ ファイルを保存するために使用されるファイル システムで最終アクセスのタイムスタンプを無効にします。

- ▶ パフォーマンス エクスペリエンスの一貫性を最大にするには、透過的 巨大ページ (THP) オプションを有効にしておく必要があります。ほとんどの場合、これは既定で有効になっています。
- ▶ Hyper-V Dynamic Memory などの仮想マシン (VM) 機能は、SQL Server インストールでは使用しないようにしてください。VM を使用する場合は、十分な固定メモリ サイズを割り当ててください。

列ストア インデックスの作成方法やインメモリ OLTP 用の構成など、SQL Server on Linux のいくつかのパフォーマンス機能を有効にする方法の詳細については、「[SQL Server on Linux のパフォーマンス機能のチュートリアル](#)」を参照してください。

高可用性構成

SQL Server 2017 は、Windows および Linux プラットフォームでのリソースの使用方法を改善し、ミッションクリティカルなアップタイム、フェールオーバー、および高可用性による管理性を向上させます。さまざまな HADR シナリオ (Always On フェールオーバークラスターインスタンス (FCI)、Always On 可用性グループ、およびログ配布) を実現し、幅広い可用性サービスレベル契約 (SLA) を締結できるよう助けます。

Always On フェールオーバー クラスター インスタンスでの高可用性

Linux 上で FCI を設定する際に知っておくとよい機能を次に示します。

クラスタリング。クラスタリングレイヤーはクラスター コントローラーの Pacemaker によって管理され、通信とリソース管理が調整されます。RHEL の場合、これは RHEL 高可用性アドオンによって提供されます。SLES の場合は、必要なパッケージが SUSE Linux [エンタープライズ高可用性拡張機能 \(HAE\)](#) によって提供されます

インスタンスとノードの数。Linux ベースの FCI では、存在できるインスタンスは 1 つだけです。つまり、コンテナを使用するのでない限り、1 つの Linux サーバーに対して存在してよい SQL Server インスタンスは 1 つだけです。

IP アドレスとホスト名。SQL Server の各インスタンスには、それぞれ独自の IP アドレスとホスト名が必要です。これらは、特定のインスタンスと通信するために Pacemaker によって内部的に使用されます。Pacemaker のセットアップには、独自の IP を持つ仮想リソースの作成が含まれているため、アプリケーションはクラスター内部リソースを特定することなくデータに接続できます。

共有ストレージ。Linux または Windows Server 上のすべての FCI には、何らかの形の共有ストレージが必要です。Linux で共有ストレージに選択可能なオプションは iSCSI および NFS (Network File System) であり、Windows Server で選択可能なオプションは SMB (Server Message Block) です。複数の場所にまたがる構成では、1 つのデータセンターに保存されたデータを複数の場所で同期させる必要があります。ユーザー データとログ ファイルの既定の場所として、すべてのインスタンスで /var/opt/mssql/data にシステム データベースが必ず存在している必要があります。

FCI リソース グループ。Linux 上の FCI のノードを、リソース グループ内に作成する必要があります。FCI リソースを作成するには、pcs resource コマンドを使用します。FCI がオンラインであることを確認した後、SSMS または sqlcmd を使用して通常の SQL 文を発行できます。

```
sudo pcs resource create FCIResourceName
ocf:mssql:fcf op defaults timeout=60s
--group RGName
```

Linux で SQL Server FCI を作成する方法の詳細については、[マイクロソフトのドキュメント](#)を参照してください。

高可用性および Always On 可用性グループ

Linux ベースの SQL Server インストール環境の Always On 可用性グループについては、次のような特性と構成設定を考慮することができます。

Linux 上で SQL Server 可用性グループの設定を構成します。Linux サーバー上に可用性グループを作成するには、エンドポイントと証明書により、各 Linux ノード上で可用性グループを有効にする必要があります。T-SQL (または Windows の場合は SSMS の新しい可用性グループ ウィザード) を使用して、適切なクラスターの種類を指定して可用性グループを作成してください。SQL Server 上に可用性グループを作成した後、Pacemaker で対応するリソースを作成する必要があります。Linux の SQL Server 2017 で可用性グループを開始するには、「[SQL Server on Linux の可用性グループを作成および構成する](#)」を参照してください。

クラスターの種類。Linux 上の可用性グループには、クラスターの種類を「外部」または「なし」に設定します。種類の「外部」は可用性グループで Pacemaker を使用することを意味し、「なし」は Pacemaker を使用する必要がないことを意味します。クラスターの種類を「外部」に設定して Pacemaker を使用する場合は、可用性グループ内の SQL Server のインスタンスを照会したり、フェールオーバー調整して高可用性を維持したりすることができます。クラスターの種類を「なし」を設定する場合は、プライマリ レプリカからセカンダリ レプリカへの手動フェールオーバーのみがサポートされます。この設定は、主に読み取りスケールアウトのシナリオを対象としています。このコンテキストにおけるリスナー接続には、任意のノード上で実行可能な Pacemaker で作成

される IP アドレス リソースを使用できます。詳細については、「[Linux でのリスナー機能](#)」を参照してください。

レプリカとクラスター ノードの数。SQL Standard Edition では、可用性グループ内の 2 つのノード、および可用性グループごとに 1 つのデータベースがサポートされます。一方、Enterprise Edition では可用性グループ内に最大 9 つのノードが存在できます。構成のみのノードは、これらの制限に含まれません。別のレプリカに自動的にフェールオーバーする機能が有効となるように 2 レプリカを設定するには、構成のみのレプリカを使用します。[レプリカとクラスター ノード](#)についての情報を参照してください。

構成のみのレプリカとクォーラム。Pacemaker が動作し続けるように正しく設定されていることを確認します。これには、構成のみのレプリカなどの SQL Server の要件に加えて、クォーラムおよび STONITH が Pacemaker の観点から適切に実装されていることを確認すること含まれます。詳細については、「[構成のみのレプリカとクォーラム](#)」を参照してください。

Pacemaker の **SQL Server リソースエージェント**。SQL Server 2017 では、sys.availability_groups に順序番号オプションが追加されました。これにより、Pacemaker は、プライマリレプリカに対してセカンダリレプリカがどの程度最新のものをかを特定することができます。Pacemaker は、

可用性グループ構成が変更されるたびに sequence_number を更新します。構成変更の例としては、フェールオーバー、レプリカの追加や除去などがあります。

REQUIRED_SYNCHRONIZED_SECONDARIES_TO_COMMIT。さらに SQL Server 2017 では、クラスター リソースの設定にこのオプションが用意されています。これは、Pacemaker から変更できます。この設定により、プライマリレプリカが各トランザクションをコミットする前に、指定された数のセカンダリレプリカがトランザクションデータをログに記録することが保証されます。高可用性とデータ保護のためには、このパラメーターの値を設定できます。デザインパターンとその機能の詳細については、「[設定が既定のリソース設定に与える影響](#)」を参照してください。

セキュリティ機能

SQL Server 2017 には、Linux デプロイメントでの搭載されたセキュリティ機能がいくつか用意されており、保存されているデータであれ流れているデータであれ組織の機密データを保護するのに役立ちます。

トランスペアレント データ暗号化 (TDE) は、保存状態にあるデータベースを暗号化するもので、アプリケーションの変更を必要としません。

Backup 暗号化では、バックアップ作成中にデータ ファイルを暗号化することができます。

常時暗号化は、クライアント側で機密データを暗号化します。

SQL 認証は、ユーザー名とパスワードによる認証レイヤーを追加します。

Active Directory (AD) 認証は、Active Directory と Kerberos によるシングルサインオン機能を提供します。

詳細なアクセス許可は、個々のテーブルやデータ列へのアクセスを制御するのに役立ちます。

行レベル セキュリティ (RLS) では、カスタマイズ可能なポリシーに基づいてテーブルの行へのアクセスを管理できます。

動的データ マスク (DDM) は、非特権ユーザーに対してあいまい化を施すことにより、機密データの露出を制限します。

ファイングレイン監査機能は、データ監査ポリシーを適用し、ユーザーの活動を追跡するのに役立ちます。

これらのセキュリティ機能の詳細については、「[SQL Server セキュリティの概要](#)」を参照してください。

ツールと管理

Linux デプロイメントと Windows デプロイメントが含まれる混在環境を管理するには、さまざまなツールが必要です。マイクロソフトでは、そうした環境を簡単に管理できるよう、SQL Server 用のクロスプラットフォーム ツールをいくつか提供しています。

SQL Operations Studio

SQL Operations Studio は、SQL Server 用の新しいクロスプラットフォーム ツールです。VSCode をベースに構築されたこの無料ツールは、Windows、MacOS、および Linux 上で動作し、オンプレミス、コンテナ、またはクラウドの SQL Server の管理に最適です。これは複数のタブ ウィンドウ、機能豊富な T-SQL エディター、IntelliSense、キーワード補完、コードスニペット、コードナビゲーション、ソース管理統合 (Git) などの搭載された機能を備え、キーボードベースの最新 T-SQL コーディング エクスペリエンスを提供します。SQL Operations Studio は、オンデマンドの T-SQL クエリを可能にし、テキスト、JSON、または Microsoft Excel などの一般的な形式で結果を表示および保存します。これを使用することにより、データベース接続を整理し、任意のプラットフォーム上で慣れ親しんだオブジェクトブラウジング方法でデータベース オブジェクトをブラウズできます。

Visual Studio Code および SQL Server の拡張機能

Visual Studio (VS) Code は、最新の Web アプリケーションやクラウド アプリケーションのビルドとデバッグのために再定義され最適化されたコード エディターです。VS Code は無料であり、好みのプラットフォーム (Linux、MacOS、および Windows) で利用できます。VS Code 拡張マーケットプレイスには、VS Code の拡張機能の豊富なエコシステムがあります。

VS Code の mssql 拡張機能を使用することにより、SQL Server に接続し、T-SQL ステートメントでクエリを実行し、結果を表示できます。結果を JSON または CSV ファイルとして保存し、そのデータをアプリケーション内で数回のクリックにより使用することができます。T-SQL コードを入力する際には、T-SQL IntelliSense (コード補完)、構文強調表示、lint、コードナビゲーション、コード スニペットなどの豊富な T-SQL 言語機能を利用できます。

VS Code を使用すれば、SSIS 拡張機能を使用して SQL Server Integration Services (SSIS) カタログとの接続を確立し、T-SQL ステートメントを使用して SSIS プロジェクトを SSIS カタログにデプロイすることもできます。

mssql-cli コマンドラインクエリツール

SQL Server でクエリを実行するための対話式的コマンドライン ツールである mssql-cli には、自動補完や構文強調などの多くの最新機能が含まれています。Python に基づく mssql-cli は、インストールのために優先インストーラー プログラム (pip) を使用します。インストールが完了したら、mssql-cli --help を使用してコマンドラインからツールを起動できます。複数のプラットフォームで機能するオープンソース ツールである mssql-cli は、Windows、MacOS、Linux にインストールすることができます。mssql-cli のインストール方法と使い方の詳細については、[mssql-cli GitHub リポジトリ](#)にあるドキュメンテーションを参照してください。

SQL Server Management Studio (SSMS)

SQL Server Management Studio は Linux 上で実行にはなっていませんが、Windows サーバーから、Linux OS 上の SQL Server インスタンスをリモートで管理するために使用できます。SSMS は、データベースのバックアップ、復元、表示、編集の操作、さらには T-SQL のクエリ、スクリプト、データベース オブジェクトの作成と編集を支援する強力なデータベース管理ツールです。具体的な手順については、「[SSMS を使用して Linux から SQL Server インスタンスにリモートで接続する方法](#)」を参照してください。

System Center 管理パック (管理パック)

SQL Server 2017 用 Microsoft System Center 管理パックは、Windows および Linux のデプロイメントにおいて、SQL Server 2017 データベース エンジン、データベース、およびその他の関連コンポーネントの検出と監視を可能にするもう 1 つのツールです。進行中の監視アプローチと管理システムを中断する必要はありません。管理パックの新しいバージョンをインポートして構成するだけで、SQL Server の追跡が容易に行えるようになります。管理パックを使用すると、Windows と Linux の両方の SQL 2017 を監視できます。さらに、管理パックは Linux 上でのエージェントレス監視をサポートしています。エージェントレス モードを使用すると、監視ワークロードを SQL Server 監視プールに含まれる管理サーバーにシフトできます。これにより、SQL Server ホストから System Center Operations Manager (SCOM) とデータ処理のオーバーヘッドを取り除いてそれを SQL Server の監視プールに移行することができます。管理パックが提供する監視には、パフォーマンス、可用性、構成の監視、パフォーマンスとイベントのデータ収集が含まれます。すべての監視ワークフローには、あらかじめ定義されたしきい値と補足的なナレッジベースの記事があります。

移行とアップグレードのための ツールとサービス

既存のデータとデータベースを SQL Server on Linux に移行するには、さまざまな方法があります。マイクロソフトでは、お客様が SQL Server へのこの移行を達成できるようにするために、多くの代表的なリレーショナル データベース管理ソリューション用の移行ツールをご用意しています。

SQL Server Migration Assistant (SSMA)

SSMA を使用すると、データベース スキーマを SQL Server スキーマに変換し、そのスキーマをアップロードして、データを移行先サーバーに移行することができます。移行プロセスは GUI によって進められ、GUI に表示される指示に従って、変換、アップロード、およびデータ移行のプロセスを完了することができます。変換時に問題が発生した場合は、GUI でハイライトされます。SSMA は Windows アプリケーションなので、Linux 上のリモート SQL Server インスタンスに接続できる Windows マシンがある場合は SSMA を使用することをお勧めします。SSMA は、Oracle、MySQL、SAP Adaptive Server Enterprise (ASE)、DB2、Microsoft Access など、さまざまなソース データベースをサポートしています。また SSMA は、ソース データベースの評価、ソース データベース スキーマから SQL Server スキーマへの変換、スキーマとデータの移行、移行のテストなどの移行タスクを自動化するのにも役立ちます。ソース データベースに基づいて SSMA のバージョンを選択する方法については、[マイクロソフトのドキュメント](#)を参照してください。

Data Migration Assistant (DMA)

DMA では、移行前に、新しいバージョンの SQL Server のデータベース機能に影響を与える可能性のある互換性の問題を検出することができます。問題が検出された場合、DMA により、先に進む前に問題の解決のために実行できる事柄が提案されます。バージョンのアップグレードにより使用できるようになる新しい機能を含め、ターゲット環境のパフォーマンスと信頼性を改善するための推奨情報が提示されます。移行ワークフローは、データベーススキーマ、データとユーザー、サーバーロール、および SQL Server ログイン情報を移行する際に役立ちます。Active Directory (AD) ログインの移行は SQL Server 2017 on Linux によって正式にサポートされていますが、それには SQL Server 2017 Linux で AD ログインを設定するための追加設定が必要です。(詳細については、[設定手順](#)を参照してください。) 標準の SQL 認証は、追加設定なしで決められているとおりに機能します。

さらに、Data Migration Service (DMS) があれば、オンプレミスにある既存の SQL Server、Oracle、MySQL データベースを、マイクロソフト Azure に移行する作業が簡単になります。ターゲットデータベースが SQL Database、SQL Database Managed Instance、Azure 仮想マシン上の SQL Server のどれであってもそう言えます。自動化されたワー

クフローにより、移行実行前に必要な変更が指示されます。プロセス ガイドとして提示されるオプションは、理解しやすく実装も容易です。

データベース実験アシスタント (DEA)

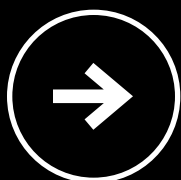
DEA は、アップグレードのターゲットの SQL バージョンを評価するのに役立ちます。これは、お客様が古い SQL Server のバージョン (2005 以降) から、新しいバージョンの SQL Server にアップグレードするためのもう 1 つの重要なツールです。DEA は、互換性エラー、劣化したクエリとクエリ プラン、およびその他のワークロード比較レポートのクエリ メトリックなどの分析レポートを提供し、それはアップグレードを正常に実行するのに役立ちます。このツールは、本番データベースの自動化されたワークロード キャプチャと再生を設定したり、古いインスタンスと新しいインスタンスの両方を使用して収集されたトレースの統計分析を実行したり、収集されたデータを分析レポートによって視覚化したりするのに役立ちます。DEA は、[マイクロソフト ダウンロード センター](#)によりインストールできます。

移行パスとしての可用性グループ

データセンター間移行やエコシステム間移行は、可用性グループのレプリカを使用することによって、バックアップ時のパフォーマンス低下を発生させることな

く、また、最小限のダウンタイムで、迅速に移行できます。クロスプラットフォーム Always On 可用性グループ構成 (レプリカの一部が Windows サーバー上にあり、他のレプリカが Linux サーバー上にある構成) を使用して、あるプラットフォームから別のプラットフォームへの移行を行うことができます。ただし、Windows と Linux はレプリカ フェールオーバーの管理に異なるコントローラー (WSFC と Pacemaker) を使用するため、異なるオペレーティング システムが含まれる可用性グループではクラスタの種類 [なし] を使用し、手動でフェールオーバーを行う必要があります。また、分散型可用性グループでは 2 つのまったく異なる可用性グループがサポートされるため、ダウンタイムの延長につながるリスクを負うことなく、プラットフォーム間で組織の高可用性構成を移行できます。詳細については、[分散可用性グループ](#)の使用に関するドキュメントを参照してください。

まとめ



リソース

[ご使用のプラットフォームで SQL Server 2017 を実行する方法の詳細。](#)

最新のデータ プラットフォームは、データからインサイトを導き出すために運用データベース、データ ウェアハウス、および各種のデータ ソースが組み合わされた異機種環境にまたがります。イノベーションを実現するには、データを最大限に活用するためにプラットフォーム、プログラミング言語、およびデータ インフラストラクチャの選択に関する柔軟性が求められます。

SQL Server 2017 は、Windows、Linux、MacOS、および Docker コンテナをサポートする SQL Server の最初のバージョンです。この新しいサポートは、開発やクロスプラットフォーム オペレーティング システムに最適なプラットフォームとしての SQL Server の大きな進展であり、データ ソリューションの柔軟性を向上させます。また、SQL Server に搭載されたインメモリ OLTP、インメモリ クラスター化列ストア、可用性グループ、適応型クエリ処理、自動チューニング、およびエンタープライズレベルのセキュリティですでに利用可能な他の最先端テクノロジーを活用することもできます。SQL Server 2017 には、組織がさまざまなオペレーティング システムとプログラミング言語でデータ エコシステムを構築して拡張するための、幅広い選択肢や機能があります。

SQL Server 2017 on Linux をインストールする方法については、次の情報を参照してください:
[Red Hat Enterprise Linux \(RHEL\) クイックスタートインストールガイド。](#)

[Docker QuickStart インストール ガイド。](#)