



Microsoft SQL Server 2017 unter Linux

Technisches Whitepaper



Veröffentlichung: September 2017

Themengebiet: Microsoft SQL Server 2017 für Linux und Docker-Container

Copyright

Die in diesem Dokument enthaltenen Informationen geben den aktuellen Standpunkt von Microsoft Corporation zu den besprochenen Themen zum Zeitpunkt der Veröffentlichung wieder. Da Microsoft auf sich wandelnde Marktbedingungen reagieren muss, dürfen diese Informationen nicht als verbindliche Zusage seitens Microsoft interpretiert werden, und Microsoft kann die Richtigkeit der Informationen nach dem Zeitpunkt der Veröffentlichung nicht garantieren.

Dieses Whitepaper dient ausschließlich zu Informationszwecken. MICROSOFT GIBT FÜR DIE IN DIESEM DOKUMENT ENTHALTENEN INFORMATIONEN KEINE GARANTIEERKLÄRUNGEN AB, WEDER AUSDRÜCKLICH NOCH STILLSCHWEIGEND NOCH GESETZLICH.

Die Einhaltung aller geltenden Urheberrechtsgesetze obliegt dem Benutzer. Ohne die im Urheberrecht festgelegten Rechte einzuschränken, darf kein Teil dieses Dokuments vervielfältigt, in einem Datenabfragesystem gespeichert oder eingeführt werden oder in irgendeiner Form oder mittels irgendwelcher Methode (elektronisch, mechanisch, mittels Kopie, Aufnahme oder anderweitig) oder für irgendwelchen Zweck ohne die ausdrückliche schriftliche Erlaubnis der Microsoft Corporation übermittelt werden.

Microsoft kann über Patente, Patentanmeldungen, Marken, Urheberrechte oder andere gewerbliche Schutzrechte verfügen, die den Gegenstand dieses Dokuments betreffen. Sofern es in einem schriftlichen Lizenzvertrag von Microsoft nicht ausdrücklich geregelt ist, erteilt die Bereitstellung dieses Dokuments Ihnen keine Lizenz für diese Patente, Marken, Urheberrechte oder andere gewerbliche Schutzrechte.

© 2017 Microsoft Corporation. Alle Rechte vorbehalten.

Microsoft, Active Directory, Azure, Bing, Excel, Power BI, SharePoint, Silverlight, SQL Server, Visual Studio, Windows und Windows Server sind Marken der Microsoft-Unternehmensgruppe.

Alle anderen Marken sind Eigentum ihrer jeweiligen Inhaber.

Inhalt

Zusammenfassung	5
Industrielandschaft und Trends	5
SQL Server 2017: Die Leistung von SQL Server jetzt auch für Linux und Container	6
SQL Server 2017 für Linux und Container	6
Native Linux-Erfahrung mit Unterstützung für vorhandene SQL Server-Tools	7
Plattformabstraktion mit Containern	8
Umsetzen von DevOps-Methoden mit Containern.....	8
Dev/Test	8
Kontinuierliche Integration und Bereitstellung	9
Neue Funktionen in SQL Server 2017 unter Linux	10
Featuregruppe der adaptiven Abfrageverarbeitung	10
Automatische Plankorrektur.....	11
Neuerstellung eines fortsetzbaren Onlineindexes	11
Graphenverarbeitung mit SQL Server 2017.....	12
Unternehmenskritische Hochverfügbarkeit mit Linux	13
Hohe Verfügbarkeit auf Instanzebene mit AlwaysOn-Failoverclusterinstanzen	13
Hochverfügbarkeit auf Datenbankebene mit AlwaysOn-Verfügbarkeitsgruppen	14
Flexibilität für HA-Architekturen.....	14
AlwaysOn-Verfügbarkeitsgruppen für Linux.....	14
Lastenausgleich von lesbaren sekundären Replikaten.....	15
Verfügbare Sicherheits- und Leistungsfunktionen in SQL Server 2017 unter Linux.....	15
In-Memory-Onlinetransaktionsverarbeitung (OLTP)	15
Columnstore	16
Always Encrypted.....	16
Transparent Data Encryption	17
Sicherheit auf Zeilenebene	18
Dynamische Datenmaskierung.....	18
Auditing.....	18

Erste Schritte mit SQL Server 2017 unter Linux.....	19
Schlussfolgerung.....	19
Handlungsempfehlungen.....	19
Feedback	20

Zusammenfassung

Die IT-Landschaft unterliegt ständigen Änderungen und einer zunehmenden Diversifizierung. Dies gilt besonders für Betriebssysteme, darunter auch Linux. SQL Server 2017 stellt die besten Features der relationalen Datenbank-Engine von Microsoft für die Linux-Unternehmensinfrastruktur bereit. Microsoft ist bestrebt, Kunden die Möglichkeit zu bieten, die beste Plattform für ihre Daten und Anwendungen auszuwählen. Dazu gehört auch die Interoperabilität mit Open-Source-Lösungen. So werden zum Beispiel Linux-Distributionen wie Red Hat Enterprise Linux (RHEL), Ubuntu und SUSE für relationale und nicht relationale Workloads immer beliebter. Mit SQL Server 2017 können Anwendungsentwickler ihre Anwendungen in jeder beliebigen Sprache entwickeln, wie zum Beispiel Node.JS, .NET, Java und PHP, und ihre Lösungen auf Plattformen wie Windows und Linux sowie in Docker-Containern bereitstellen, und dies alles in einer privaten Cloud (On-Premises), auf Microsoft Azure oder in Clouds von externen Anbietern.

Industrielandtschaft und Trends

Die IT-Landschaft unterliegt ständigen Änderungen und einer zunehmenden Diversifizierung, insbesondere im Bereich der heterogenen Betriebssysteme (Linux, Windows usw.). Außerdem werden IT-Unternehmen mit vielen Datentypen, verschiedenen Entwicklungssprachen und einer Mischung aus On-Premises-, Cloud- und Hybridumgebungen konfrontiert. Immer komplexere Umgebungen zu unterhalten, ist eine gewaltige Aufgabe, bei der Menschen und Prozesse Schritt halten müssen, während gleichzeitig die Betriebskosten gesenkt werden sollen.

Microsoft ist bestrebt, Kunden die Möglichkeit zu bieten, die beste Plattform für ihre Daten und Anwendungen auszuwählen. Dazu gehört auch die Interoperabilität mit Open-Source-Lösungen. Ein Drittel der virtuellen Azure-Maschinen läuft unter Linux. Microsoft-Plattformen wie HDInsight und R Server können sowohl auf Windows als auch auf Linux bereitgestellt werden. Dieses Engagement reicht auch in den Bereich der Anwendungsentwicklung, zum Beispiel für .NET Core als Open-Source-Projekt und die kürzlich angekündigte Visual Studio Code-Erweiterung für SQL Server. Außerdem sind SQL Server-Treiber, Verbindungstools und APIs für nahezu jede Umgebung verfügbar, sodass jede Anwendung unabhängig von der Programmiersprache und Umgebung in SQL Server integriert werden kann.

SQL Server 2017: Die Leistung von SQL Server jetzt auch für Linux und Container

SQL Server 2017 basiert auf den branchenführenden¹ Fähigkeiten von SQL Server 2016, das bei den folgenden Benchmarks ganz vorne liegt:

- **Leistung:** SQL Server ist führend bei den TPC-E-Leistungsbenchmarks für die Verarbeitung von Transaktionen, den TPC-H-Leistungsbenchmarks für Data Warehousing und den Leistungsbenchmarks für führende Geschäftsanwendungen². Im April 2017 hat Hewlett Packard Enterprise einen neuen TPC-H-1TB-Weltrekord für SQL Server 2017 unter Linux veröffentlicht.³
- **Sicherheit:** Gemäß dem Ausschuss für öffentliche Sicherheit des National Institute of Standards and Technology (NIST) weist SQL Server im Vergleich mit den wichtigsten Datenbank Anbietern die geringste Anzahl von Sicherheitsrisiken auf (NIST, 2016).
- **Gesamtbetriebskosten:** Die Gesamtbetriebskosten von SQL Server sind bedeutend niedriger als die vergleichbarer Datenlösungen für Unternehmen. In einigen Fällen waren die Gesamtbetriebskosten für SQL Server 2016 um ein Zwölftel geringer als die vergleichbarer Produkte/Funktionen.

SQL Server 2017 unter Linux knüpft an die Entwicklung von SQL Server an, indem die moderne Dateninfrastruktur durch neue Fähigkeiten erweitert wird, mit denen die Datenverwaltung und datenbezogene Anwendungen besser unterstützt und erweitert werden. In den folgenden Szenarien werden neben den neuesten Funktionen auch mögliche Einsatzmöglichkeiten für SQL Server 2017 unter Linux vorgestellt.

SQL Server 2017 für Linux und Container

SQL Server 2017 stellt die besten Features der relationalen Datenbank-Engine von Microsoft für die Linux-Unternehmensinfrastruktur bereit. Dies umfasst SQL Server Agent, Active Directory-Authentifizierung, branchenführende Hochverfügbarkeit / Notfallwiederherstellung und beispiellose Datensicherheit. Machen Sie sich bewusst, dass SQL Server 2017 unter Linux kein portiertes oder neu geschriebenes System ist. Es ist das gleiche erstklassige Managementsystem für relationale Datenbanken (Relational Database Management System, RDBMS) von Microsoft, das jetzt auch auf weiteren Betriebssystemen wie etwa Red Hat Enterprise Linux, SUSE Linux Enterprise Server und Ubuntu sowie weiteren Cloud- und Containerplattformen (wie etwa Docker) verfügbar ist.

¹ Gartner hat Microsoft bereits in zwei aufeinanderfolgenden Jahren als führenden Anbieter für operationale Datenbankverwaltungssysteme eingestuft, weil das Unternehmen hält, was es verspricht, und sich durch eine umfassende Vision auszeichnet. *Gartner empfiehlt die in dieser Marktanalyse genannten Lösungsanbieter, Produkte oder Dienstleistungen nicht und rät Technologienutzern nicht, ausschließlich die am höchsten bewerteten oder auf andere Arten ausgezeichneten Lösungsanbieter auszuwählen. Die Gartner-Forschungspublikationen beruhen auf den Meinungen der Gartner-Forschungsorganisation und sollten nicht als Feststellung eines Sachverhalts aufgefasst werden. Gartner schließt alle ausdrücklichen oder stillschweigenden Gewährleistungen hinsichtlich dieser Studie inklusive Marktgängigkeit oder Eignung für einen bestimmten Zweck aus.*

² Lenovo hat einen Weltrekord für den TPC-H 10 TB-Benchmark (Lenovo Server Performance Lab, 2016) mit SQL Server 2016 erreicht. Im Mai 2016 hat Lenovo außerdem einen neuen Weltrekord für den TPC-H 30 TB-Benchmark erreicht (Lenovo Server Performance Lab, 2016).

³ www.tpc.org, "TPC-H Result Highlights HPE Proliant DL380 Gen9", März 2017 ([Link](#))

SQL Server 2017 bietet jetzt auf allen unterstützten Plattformen, wie zum Beispiel unter Windows und Linux sowie in Containern, die besten Leistungs- und Sicherheitsfunktionen. Hochleistungsfunktionen verleihen datengestützten Anwendungen Rekordgeschwindigkeit. Diese Funktionen umfassen Columnstore, der spaltenbasierte Datenspeicherung und -verarbeitung bietet und damit eine im Vergleich zur zeilenbasierten Speicherung bis zu 10 Mal höhere Abfrageleistung und Datenkomprimierung erzielt, sowie In-Memory-OLTP. Mit dieser wird eine Transaktionsverarbeitung für speicheroptimierte Tabellen bereitgestellt, die bis zu 2,5 Mal schneller als bei festplattenbasierten Tabellen ist. Sicherheitsfunktionen wie Auditing, Transparent Data Encryption, Sicherheit auf Zeilenebene, Always Encrypted und dynamische Datenmaskierung sind serverseitige Sicherheitsmaßnahmen, die den Schutz der Daten vor unbefugtem Zugriff erheblich vereinfachen, ohne vorhandene Clientanwendungen ändern zu müssen. Mit Auditing können Teams den Zugriff überwachen und möglicherweise verdächtige Aktivitäten nachverfolgen. Transparent Data Encryption schützt ruhende Daten auf Dateiebene, und Always Encrypted schützt ruhende und übertragene Daten. Da diese Fähigkeiten bei allen SQL Server-Editionen verfügbar sind, müssen Unternehmen ihre Bereitstellungsumgebung nicht mehr nach den gewünschten Funktionen auswählen, sondern können diese stattdessen nach den betrieblichen Anforderungen ermitteln.

SQL Server 2017 unter Linux ist kein portiertes oder neu geschriebenes System. SQL Server unter Windows und Linux verwenden eine gemeinsame Codebasis, die über eine Plattformabstraktionsschicht auf Betriebssystemfunktionen auf niedriger Ebene zugreift. Auch wenn noch nicht alle von SQL Server unter Windows abgedeckten Szenarien und Funktionen unter Linux unterstützt werden, ist SQL Server 2017 unter Linux bereit für transaktionale und Data Warehousing-Workloads und für die Teilnahme an Verfügbarkeitsgruppen. SQL Server Integration Services für ETL (Extrahieren, Transformieren, Laden) von Daten aus verschiedenen Datenquellen in ein SQL Server Data Warehouse wird unter Linux jetzt ebenfalls unterstützt. Der größte Teil der Database Engine-Workloads kann ohne Änderungen von Windows auf Linux verschoben werden. Weitere Informationen zu den SQL Server-Funktionen, die unter Linux noch nicht unterstützt werden, finden Sie in den [SQL Server unter Linux-Versionshinweisen](#).

Native Linux-Erfahrung mit Unterstützung für vorhandene SQL Server-Tools

Microsoft hat es sich zum Ziel gesetzt, bereits ab dem Installationsvorgang eine Linux-native Benutzererfahrung für SQL Server zu bieten. SQL Server 2017 verwendet die standardmäßige paketgestützte Installationsmethode für Linux unter Verwendung von yum für Fedora-basierte und apt-get für Debian-basierte Distributionen. Administratoren können SQL Server 2017-Instanzen unter Linux mit ihren vorhandenen Paket-Update/-Upgrade-Prozessen aktualisieren.

Der SQL Server-Dienst läuft nativ mit systemd, und die Leistung kann über das Dateisystem überwacht werden. Linux-Dateipfade werden in T-SQL-Anweisungen und -Skripten unterstützt, um Vorgänge wie die Definition/Änderung des Speicherorts von Dateien oder Datenbanksicherungsdateien durchzuführen. Hochverfügbares Clustering kann mit beliebigen hochverfügbaren Linux-Lösungen wie Pacemaker und Corosync verwaltet werden.

Die Volltextsuche ist jetzt für Linux verfügbar. Mit dieser Funktion können Sie Volltextabfragen für charakterbasierte Daten in SQL Server-Tabellen durchführen. Mit Volltextabfragen führen Sie Sprachsuchen in Textdateien von Volltextindizes durch, indem Wörter und Sätze basierend auf den Regeln einer Sprache, wie Englisch oder Japanisch, verarbeitet werden. Volltextabfragen können einfache Wörter und Sätze oder verschiedene Varianten eines Wortes oder Satzes enthalten. Bei einer Volltextabfrage werden alle Dokumente ausgegeben, die mindestens eine Übereinstimmung enthalten (auch *Treffer* genannt). Ein Treffer liegt dann vor, wenn ein Zieldokument alle in der Volltextabfrage aufgeführten Begriffe enthält und beliebige andere Suchkriterien erfüllt, wie zum Beispiel die Entfernung zwischen den übereinstimmenden Begriffen. Weitere Informationen zu dieser Funktion finden Sie unter [SQL Server-Volltextsuche unter Linux](#).

Microsoft bietet Tools wie den Data Migration Assistant an, um die Migration vorhandener Workloads nach SQL Server 2017 zu erleichtern.

Plattformabstraktion mit Containern

Container sind softwaredefinierte Bereiche, die Ähnlichkeiten zu virtuellen Computern aufweisen. Container verwenden keine Hardwarevirtualisierung, sondern das Hostbetriebssystem wird aus dem Bereich abstrahiert, sodass der Container nur die für die Anwendung erforderlichen Abhängigkeiten und die Anwendung selbst enthält. Ein Linux-basierter Container kann auf allen Linux-Geräten (physisch oder virtuell) durch die Ausführung von Docker bereitgestellt und in der Regel ohne Veränderungen des Hostbetriebssystems ausgeführt werden.

Dank der Unterstützung für Windows- und Linux-Container kann SQL Server in Containerorchestrierungslösungen wie Docker Swarm, Red Hat Open Shift, Mesosphere DC/OS und Kubernetes eingesetzt werden. Mit dem Management Pack für SQL Server unter Linux können Administratoren System Center Operations Manager zur umfassenden Überwachung einsetzen, von der Hardware bis hin zu Instanzen der Datenbankmodule und einzelnen Datenbanken.

Auf ähnliche Weise ermöglicht Docker CE für Mac Entwicklern den Einsatz von Linux-Containern unter macOS. SQL Server unterstützt zudem die Installation in Windows-Containern. Einer der Hauptvorteile bei der Verwendung von Containern im Entwicklungsprozess ist die Möglichkeit, in verschiedenen Umgebungen zu arbeiten. Mit Containern können Entwicklungsteams in Entwicklungs-/Testumgebungen arbeiten, deren Funktionen mit denen der Produktionsumgebungen übereinstimmen. Zudem können Container-Orchestrierungslösungen automatisch die Bereitstellung von Containern verwalten, wodurch die Automatisierung von Tests und der Bereitstellung unterstützt werden.

Umsetzen von DevOps-Methoden mit Containern

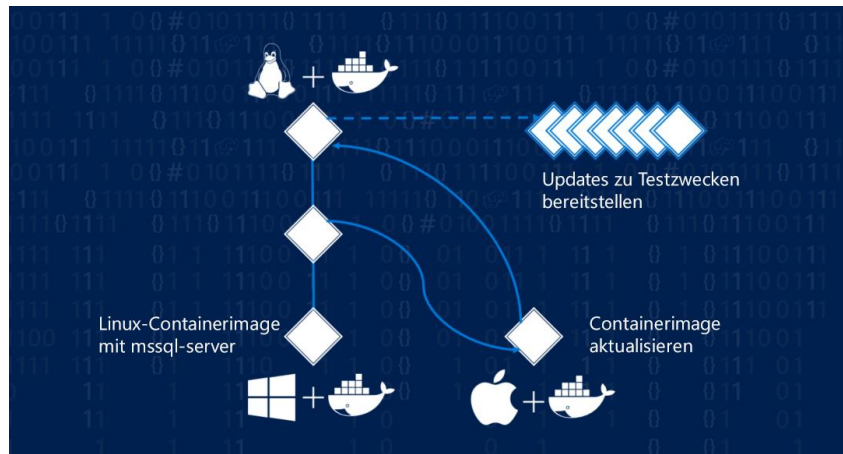
DevOps bietet Kunden durch Menschen, Prozesse und Tools hervorragende Anwendungen. Ein schlanker Ansatz der Produktentwicklung (zum Beispiel durch Aufteilen der Arbeit in kleinere Batches und Implementierung von Kunden-Feedback) verspricht eine höhere IT-Leistung und weniger Probleme bei der Bereitstellung.⁴ Verglichen mit den Herausforderungen eines traditionellen Bereitstellungsansatzes oder sogar eines agilen Ansatzes hat sich DevOps durch die Bereitstellung von Hochleistungslösungen auf dem Markt bewährt. Traditionelle Hardware- und VM-Installationen von SQL Server hatten bei der Anpassung in dieses Framework einige Anfangsschwierigkeiten. Da SQL Server jetzt auch in Containern verfügbar ist, stehen viele DevOps-Methoden für die datengestützte Anwendungsentwicklung zur Verfügung, darunter auch die Möglichkeit, ein Containerimage zu erstellen, das in jeder Umgebung verwendet werden kann.

Dev/Test

Dev/Test basiert auf dem Prinzip von identischen Umgebungen für Entwicklung, Test und Produktion. Auf diese Weise wird die Gefahr minimiert, dass nach der Bereitstellung in der Produktionsumgebung konfigurations- oder abhängigkeitsbasierte Probleme auftreten. Mit Containern werden die Anwendungsabhängigkeiten innerhalb des Containers verpackt. Container können auf jedem Host betrieben werden, der Container unterstützt, beispielsweise Red Hat Enterprise Linux oder Windows Server 2016. Dies bedeutet, dass in containerbasierten Lösungen das Host-Betriebssystem nicht mit dem Host-Betriebssystem in der Produktionsumgebung übereinstimmen muss.

⁴ Puppet and Dora, 2016 State of DevOps Report ([Link](#)).

Ein Entwicklungsteam kann beispielsweise in einer Entwicklungsumgebung arbeiten, möglicherweise unter Linux oder Mac, und den Container mit Docker lokal ausführen, inklusive des SQL Server-Datenbankmoduls und der für die Anwendung benötigten Datenbanken. Als Testumgebung kann ein Cluster aus Linux-Hosts mit Hunderten von Instanzen für automatische Tests eingesetzt werden. Die Produktionsumgebung ist in der Praxis identisch und verwendet orchestrierte Container in der Private/Public/Hybrid Cloud-Infrastruktur. Die SQL Server-Datenbankkonfiguration, das Datenbankschema und vorab ausgefüllte Daten werden als Teil des Containerimages mit jeder Containerinstanz repliziert, um Konfigurationsunterschiede als mögliche Fehlerursache zu eliminieren.



Kontinuierliche Integration und Bereitstellung

Kontinuierliche Integration und Bereitstellung (Continuous Integration and Continuous Delivery, CI/CD) sind DevOps-Methoden, bei denen Updates und Korrekturen fortlaufend integriert und neue Versionen innerhalb von Minuten anstatt von Tagen bereitgestellt werden. Auf diese Weise lassen sich Produkte schneller, hochwertiger, zuverlässiger, effizienter und kundenfreundlicher auf den Markt bringen. Mit SQL Server 2017 in Containern können Containerorchestrierungslösungen wie Docker Swarm oder Red Hat Open Shift Container-Updates schnell in automatisierten Testumgebungen und anschließend in der Produktionsumgebung verteilen.

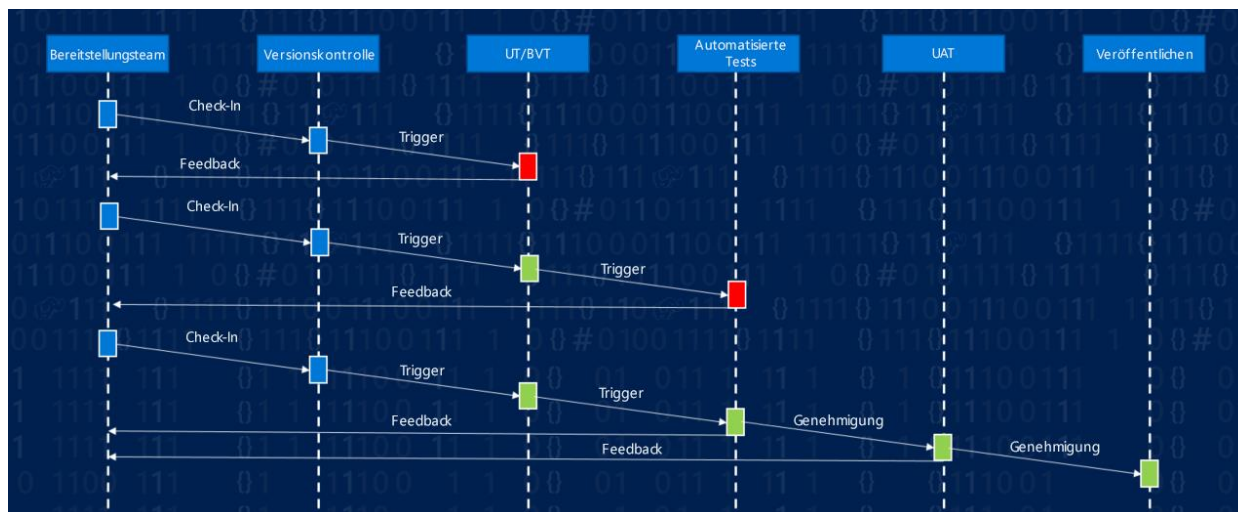


Abbildung: CI/CD-Ablaufdiagramm

CI/CD bedeutet eine Abkehr von herkömmlichen Softwareversionen und ermöglicht die Veröffentlichung neuer Versionen im Tagestakt oder sogar noch häufiger. Dafür ist ein hoher Automatisierungsgrad erforderlich, um nach einer Änderung die Bereitstellung in Umgebungen auszulösen, Tests durchzuführen und mit der nächsten Entwicklungsstufe fortzufahren. Die Vorteile überwiegen die Kosten, und die Teams können hochwertige Produkte noch zuverlässiger, effizienter und schneller ausliefern, was wiederum die Kundenzufriedenheit verbessert und die häufigere Veröffentlichung neuer Inhalte und Funktionen ermöglicht.

SQL Server 2017 zur Ausführung in Containern ist konform mit den DevOps-Prinzipien und ermöglicht verschiedene Schlüsselpraktiken, die bei der Bereitstellung von unternehmenskritischen intelligenten Apps in der Produktion eine wesentliche Rolle spielen, kosteneffizient sind sowie eine hohe Qualität und Kundenzufriedenheit bieten. Von der Parität der Umgebung mit Dev/Test bis zu Hochgeschwindigkeits-Bereitstellungen durch CI/CD ist SQL Server 2017 für Linux und Container eine leistungsstarke Ergänzung der DevOps-Tools in Unternehmen.

Neue Funktionen in SQL Server 2017 unter Linux

Seit SQL Server 2017 mit der Unterstützung für Linux und Docker bietet SQL Server als Plattform eine echte Auswahl: Entwicklungssprachen, Datentypen, lokal oder Cloud, verschiedene Betriebssysteme, allesamt mit Unterstützung für Big Data und Advanced Analytics. Eine vollständige Liste der neuen Funktionen und Verbesserungen in bestimmten Funktionsbereichen finden Sie unter [Neuigkeiten in SQL Server 2017](#).

Featuregruppe der adaptiven Abfrageverarbeitung

In SQL Server 2017 werden mit der adaptiven Abfrageverarbeitung neue Funktionen eingeführt, mit denen der SQL Server-Abfrageprozessor Planentscheidungen basierend auf Laufzeiteigenschaften ändern kann.

Während der Abfrageverarbeitung und -optimierung ist der Vorgang zur Einschätzung der Kardinalität (CE) verantwortlich für die Einschätzung der Anzahl von Reihen, die in jedem Schritt eines Ausführungsplans verarbeitet werden. Ungenaue Einschätzungen können in einer längeren Antwortzeit für die Anfrage, einer sehr hohen Ressourcennutzung (CPU, Speicher, E/A), einem geringeren Durchsatz und einer niedrigeren Parallelität resultieren. Zur Verbesserung der CE-Technik führt SQL Server 2017 eine neue Featuregruppe, die adaptive Abfrageverarbeitung (AQP), ein. Mit AQP können hartnäckige CE-Probleme leichter behoben werden. Funktionen der AQP-Featuregruppe:

Überlappende Ausführung: Realisiert problematische Schätzungen von Tabellenwertfunktionen mit mehreren Anweisungen (MSTVF), die an nachgeschaltete Vorgänge weitergegeben werden, korrigiert ungenaue Schätzungen und ermöglicht dem Abfrageoptimierer die Überarbeitung von Planentscheidungen anhand von exakten Schätzungen. Die erste Version der überlappenden Ausführung befasst sich mit der Einschätzung der Kardinalität für MSTVFs.

Adaptive Joins im Batchmodus: Ermöglicht die Verzögerung der Auswahl zwischen einem Hash-Join oder einem Join geschachtelter Schleifen und einer Columnstore-Tabelle, bis die erste Join-Eingabe analysiert wurde. Adaptive Join wertet die Eingabe aus und führt den effizientesten der beiden Join-Algorithmen zur Ausführungszeit aus.

Feedback für die Speicherzuweisung im Batchmodus: Überwacht den tatsächlich für eine Abfrage benötigten Speicher, und ermöglicht beim nächsten Aufruf einer identischen Anweisung eine exaktere Speicherzuweisung. Auf diese Weise werden übermäßige Speicherzuweisungen vermieden, um die Parallelität zu reduzieren, sowie zu klein dimensionierte Speicherzuweisungen, die dazu führen können, dass die Festplatte als Datenüberlauf verwendet wird.

Automatische Plankorrektur

Neue Funktionen von SQL Server 2017 erkennen Regressionen bei der Planauswahl und empfehlen Maßnahmen zur Problembeseitigung. Diese Funktionen für die automatische Plankorrektur helfen dabei, die Leistung von Datenabfragen auch dann beizubehalten, wenn Anwendungsänderungen auftreten.

Automatische Optimierung: Diese Datenbankfunktion liefert Einblicke in potenzielle Abfrageleistungsprobleme, empfiehlt Lösungen und korrigiert identifizierte Probleme automatisch.

Erzwingen des letzten funktionierenden Plans: Die Benutzer sollten das System regelmäßig auf Abfragen mit rückläufiger Leistung überwachen, um unerwartete Leistungsprobleme zu vermeiden. Wenn ein Plan rückläufig ist, sollte ein vorheriger funktionierender Plan gesucht und anstelle des aktuellen Plans verwendet werden. Mit dieser Funktion können Sie die Leistung einer ausgeführten Abfrage mit dem erzwungenen Plan überwachen und überprüfen, ob der Plan wie erwartet funktioniert.

Automatische Regressionserkennung: Das Datenbankmodul erkennt potenzielle Regressionen bei der Planauswahl und empfiehlt Maßnahmen in der Ansicht "sys.dm_db_tuning_recommendations". Diese Ansicht zeigt Informationen über das Problem, die Bedeutung des Problems und Details, wie beispielsweise die ermittelte Abfrage, die ID des rückläufigen Plans, die ID des Plans, der als Grundlage für den Vergleich verwendet wird, und die Transact-SQL-Erklärung, die ausgeführt werden kann, um das Problem zu beheben.

Automatische Planoptimierung: Das Datenbankmodul kann automatisch zum letzten funktionierenden Plan wechseln, wenn eine Regression erkannt wird.

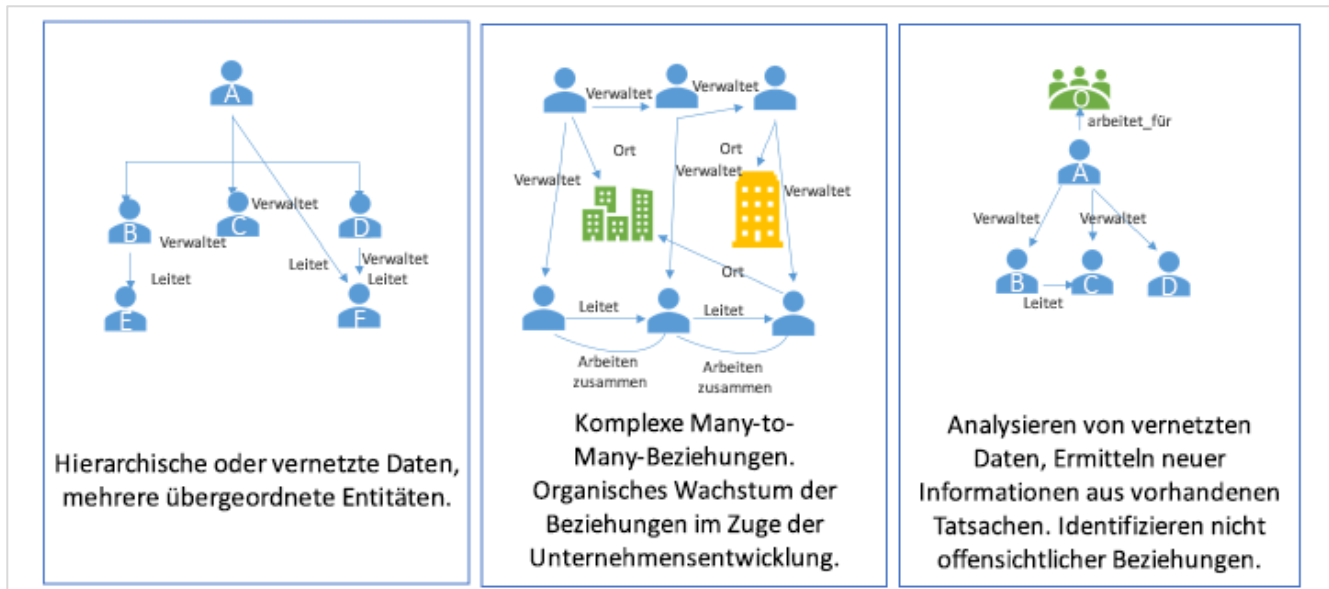
Neuerstellung eines fortsetzbaren Onlineindexes

Einen umfangreichen Index online zu planen, zu warten und zu verwalten, kann eine große Herausforderung sein. Mit zunehmender Größe von Indizes steigt auch der Aufwand für deren Wartung. Die Neuorganisation und der Wiederaufbau von Indizes kann besonders umständlich sein. Die Neuerstellung eines fortsetzbaren Onlineindexes in SQL Server 2017 ermöglicht das Fortsetzen einer Indexneuerstellung nach einem Fehler, z. B. durch einen Failover auf ein Replikat oder durch unzureichenden Festplattenplatz. Durch die Neuerstellung eines fortsetzbaren Onlineindexes kann der Vorgang der Neuerstellung eines Onlineindexes angehalten und später fortgesetzt werden. Es ist zum Beispiel möglich, dass Sie vorübergehend Systemressourcen freimachen müssen, um eine besonders wichtige Aufgabe auszuführen, oder die Neuerstellung des Indexes zu einem Zeitpunkt fortsetzen müssen, wenn das verfügbare Wartungsfenster für eine große Tabelle zu knapp bemessen ist. Die Neuerstellung eines fortsetzbaren Onlineindexes benötigt zudem nicht besonders viel Speicherplatz, sodass Sie während des Vorgangs eine Protokollkürzung vornehmen können.

Die Neuerstellung eines nicht-fortsetzbaren Onlineindexes ist bereits seit SQL Server 2005 eine Funktion der SQL Server Enterprise Edition.

Graphenverarbeitung mit SQL Server 2017

Neben der Verwaltung großer Datenmengen müssen Kunden auch andere Aufgaben bewältigen. Sie müssen ihre vorhandenen Daten effizienter analysieren, um die Beziehungen und Muster zu verstehen. Das Abfragen von Daten aus einem relationalen Schema mit traditionellen SQL-Abfragen kann eine komplexe Aufgabe sein. SQL Graph wurde zusammen mit SQL Server 2017 eingeführt, um die Modellierung und Analyse von Beziehungen zu vereinfachen, indem Benutzer die Beziehungen flexibler und agiler bearbeiten können.



Eine Graphdatenbank besteht aus einer Gruppe von Knoten (oder Scheitelpunkten) und Kanten (oder Beziehungen). Eine Graphdatenbank ist nützlich für die Repräsentation von Daten, die viele und häufig auch komplexe Beziehungen enthalten. Mit dem in SQL Server 2017 integrierten SQL Graph erhält SQL Server die Fähigkeit, Graphen zu verarbeiten, sodass Benutzer verschiedene Teile verbundener Daten zusammenführen können, um wertvolle Einblicke zu erhalten und die betriebliche Flexibilität zu verbessern. Dies ist besonders gut für Anwendungen geeignet, bei denen Beziehungen eine wichtige Rolle spielen, wie zum Beispiel Betrugsermittlung, Risikomanagement, soziale Netzwerke, Empfehlungsmodule, prädiktive Analysen und IoT Suites.

Diagrammobjekte erstellen

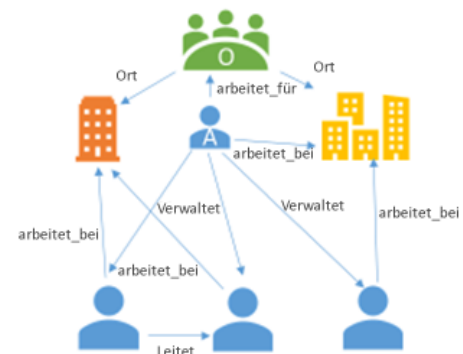
- Knoten und Kanten erstellen
- Mit Knoten und Kanten verbundene Eigenschaften

```
CREATE TABLE Person (ID INTEGER PRIMARY KEY,
name VARCHAR(100)) AS NODE;
```

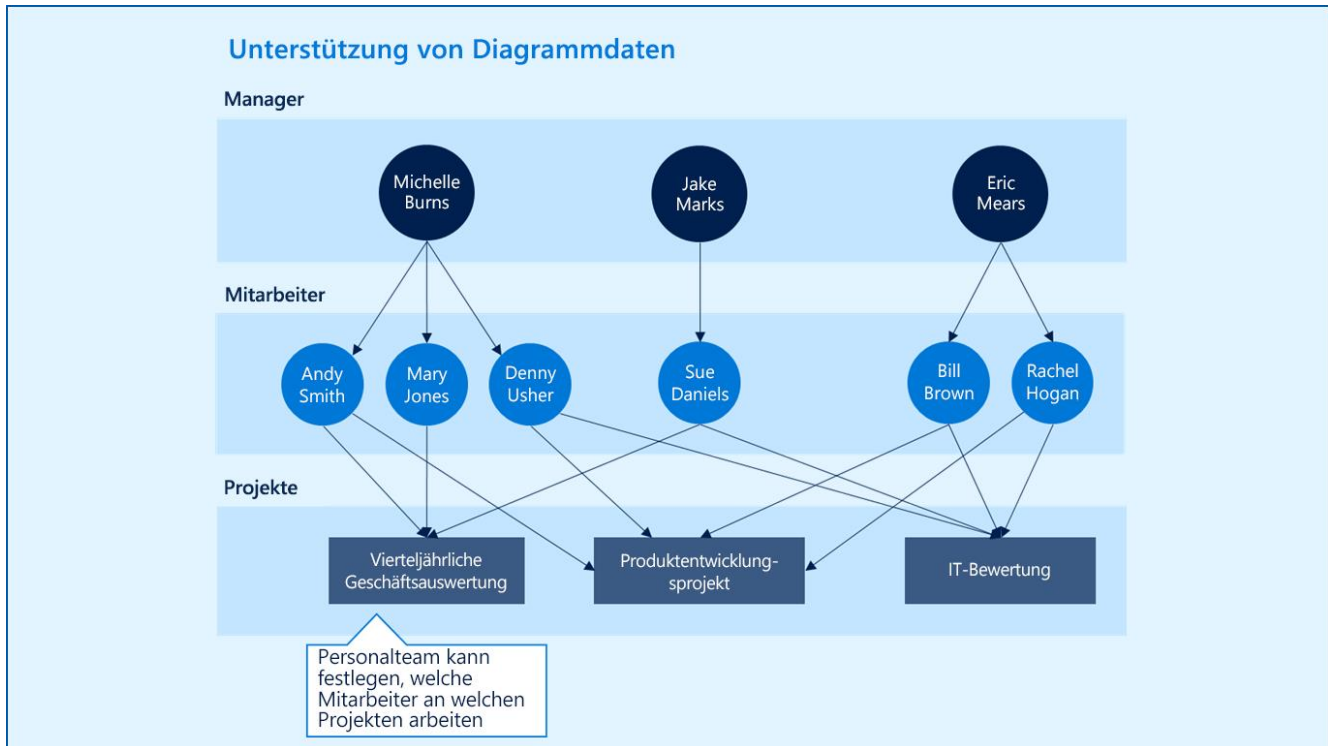
```
CREATE TABLE Restaurant (ID INTEGER PRIMARY KEY,
name VARCHAR(100)) AS NODE;
```

```
CREATE TABLE likes AS EDGE;
```

```
CREATE TABLE friends (StartDate date) AS EDGE;
```



CRUD-Vorgänge mit SQL Graph erstellen effektiv Knoten zur Repräsentation verschiedener Entitäten und Kanten zur Repräsentation von Beziehungen zwischen zwei beliebigen Knoten. Beiden Knoten und Kanten können Eigenschaften zugeordnet sein. Zudem kann SQL Graph in einem Graph mithilfe eines Musterabgleichs eine Navigation über mehrere Hops durchführen (keine Joins). Die SQL-Spracherweiterungen für die Graphen unterstützen Join-freie Abfragen mit Musterabgleich für die Navigation über mehrere Hops.



Unternehmenskritische Hochverfügbarkeit mit Linux

Die Hochverfügbarkeitslösungen stellen unternehmenskritische Betriebszeiten, schnelle Failover, verbesserte Verwaltungsfähigkeit und eine bessere Nutzung der Hardwareressourcen bereit.

Hohe Verfügbarkeit auf Instanzebene mit AlwaysOn-Failoverclusterinstanzen

Eine AlwaysOn-Failoverclusterinstanz (FCI) bietet Redundanz auf Instanzebene, sodass eine SQL Server-Instanz während geplanter und ungeplanter Betriebssystemausfälle durch Hardware- und Softwarefehler oder Systemwartung weiterhin verfügbar bleibt. FCIs werden unter Windows und Linux unterstützt.

Eine FCI umfasst zwei oder mehr Clusterknoten mit Zugriff auf gemeinsam genutzten Clusterspeicher (entweder SAN oder Direct Attached Storage). Es ist immer nur ein Knoten aktiv. Sekundäre Knoten sind verfügbar, aber passiv und können während eines Ausfalls die Rolle des aktiven Knotens übernehmen. Eine FCI benötigt einen Clustermanager für die Mobilisierung der Clusterressourcen. Unter Linux wird Pacemaker als Clustermanager unterstützt. Weitere Informationen zum Leistungsumfang dieser Funktion finden Sie unter [AlwaysOn-Failoverclusterinstanzen \(SQL Server\)](#).

Hochverfügbarkeit auf Datenbankebene mit AlwaysOn-Verfügbarkeitsgruppen

Eine Verfügbarkeitsgruppe unterstützt eine replizierte Umgebung für einen diskreten Benutzerdatenbanksatz, die als Verfügbarkeitsdatenbanken bezeichnet werden. Sie können eine Verfügbarkeitsgruppe für Hochverfügbarkeit (HA) oder Leseskalierung erstellen. Eine HA-Verfügbarkeitsgruppe ist eine Gruppe von Datenbanken, bei denen ein Failover gemeinsam erfolgt. Eine Leseskalierungsverfügbarkeitsgruppe ist eine Gruppe von Datenbanken, die in andere Instanzen von SQL Server für einen geschützten Workload kopiert werden. Eine Verfügbarkeitsgruppe unterstützt einen Satz primärer Datenbanken und einen bis acht Sätze entsprechender sekundärer Datenbanken. AlwaysOn-Verfügbarkeitsgruppen bieten dieselbe Hochverfügbarkeit und Notfallwiederherstellung wie Oracle Real Application Clusters, jedoch mit weniger Servern, und sind außerdem im Core-Lizenzpreis für SQL Server enthalten.

SQL Server 2017 führt die folgenden verbesserten Funktionen zur Gewährleistung einer hohen Verfügbarkeit bei der Ausführung unternehmenskritischer Workloads ein.

Flexibilität für HA-Architekturen

SQL Server 2017 unterstützt zwei verschiedene Architekturen für Verfügbarkeitsgruppen: AlwaysOn und Leseskalierung.

AlwaysOn-Verfügbarkeitsgruppen bieten Hochverfügbarkeit, Notfallwiederherstellung und Leseskalierungsausgleich. Diese Verfügbarkeitsgruppen benötigen Pacemaker für Linux-Cluster.

Leseskalierungsverfügbarkeitsgruppen bieten schreibgeschützte Replikate von Arbeitslasten, jedoch keine Hochverfügbarkeit. Bei dieser Architektur wird kein Clustermanager benötigt. Der Vorteil hierbei sind die sekundären Repliken in gemischten Betriebssystemumgebungen. Leseskalierungsverfügbarkeitsgruppen wurden mit SQL Server 2017 neu eingeführt.

AlwaysOn-Verfügbarkeitsgruppen für Linux

AlwaysOn-Verfügbarkeitsgruppen wurden zur Linux-Edition hinzugefügt, sodass Kunden bei der Ausführung kritischer Workloads die Widerstandsfähigkeit der Datenbanksoftware testen können. Diese Funktion ist jetzt in allen von SQL Server 2017 unterstützten Linux-Distributionen verfügbar, d. h. Red Hat Enterprise Linux, Ubuntu und SUSE Linux Enterprise Server. Alle Funktionen, die Verfügbarkeitsgruppen zu einer flexiblen, integrierten und effizienten HADR-Lösung machen, sind unter Linux verfügbar. Dazu zählen auch Failover auf mehreren Datenbanken, mehrfach synchrone und asynchrone Sekundärspeicher, manueller oder automatischer Failover, aktive Sekundärspeicher für Lese- und Backup-Workloads usw. Weitere Informationen zum Leistungsumfang dieser Funktion finden Sie unter [AlwaysOn-Verfügbarkeitsgruppen für SQL Server unter Linux](#).

Mitgliedsinstanzen einer einzigen verteilten AlwaysOn-Verfügbarkeitsgruppe können unter Windows, Linux oder einer Mischung der beiden Betriebssysteme ausgeführt werden. Auf diese Weise können Unternehmen, die ihre SQL Server nach Linux migrieren möchten, ihre Arbeitslasten und Anwendungen vor der Umstellung mühelos testen.

Lastenausgleich von lesbaren sekundären Replikaten

Sekundäre Replikate unterstützen den schreibgeschützten Zugriff auf alle sekundären Datenbanken. In der Regel sind diese Replikate synchron mit dem primären Replikat, einschließlich Volltextindizes und dauerhaften speicheroptimierten Tabellen. Dies bedeutet, dass sekundäre Replikate ähnliche Services wie Data Mart bieten, indem sie mit einer geringen Latenzzeit schreibgeschützten Zugriff auf Produktionsdaten bieten. Der Lastenausgleich für das Routing schreibgeschützter Abfragen kann ebenfalls auf dem Verfügbarkeitsgruppen-Listener durchgeführt werden, sodass Unternehmen bestimmen können, wie schreibgeschützte Workloads zu den sekundären Repliken geroutet werden. Weitere Informationen finden Sie unter [Konfigurieren des schreibgeschützten Routing für eine Verfügbarkeitsgruppe](#).

Verfügbare Sicherheits- und Leistungsfunktionen in SQL Server 2017 unter Linux

SQL Server 2017 bietet Entwicklern und Unternehmen eine einheitliche Programmieroberfläche für verschiedene SQL Server-Editionen. Auf diese Weise können Kunden und Partner erweiterte Anwendungen entwickeln, die wachstumsgesteuert über verschiedene Editionen und in der Cloud wachsen. Entwickler und Anwendungspartner können eine einzige Programmieroberfläche für die Erstellung oder Erweiterung intelligenter Anwendungen verwenden und eine Edition verwenden, die optimal mit den jeweiligen Anwendungsanforderungen skaliert. Dazu gehören Funktionen wie In-Memory-OLTP, In-Memory-Columnstore, operative Echtzeitanalysen, Kompression und Partitionierung. Sicherheitsfunktionen wie Always Encrypted, Sicherheit auf Zeilenebene, dynamische Datenmaskierung, Auditing und Transparent Data Encryption sind unter Linux ebenfalls verfügbar. Die folgenden Features sind nur ein Teil der verfügbaren Funktionen und Möglichkeiten in der Linux-Version von SQL Server 2017. Die [technische SQL Server-Dokumentation](#) ist eine hervorragende Ressource für weitere Informationen über alle verfügbaren Features, Funktionen und Dienste in SQL Server.

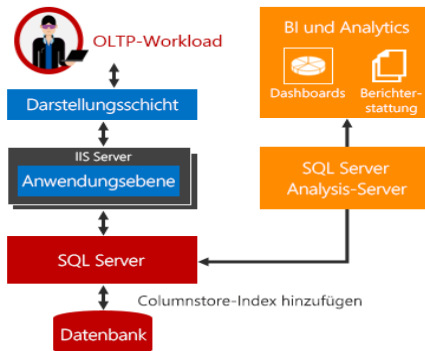
In-Memory-Onlinetransaktionsverarbeitung (OLTP)

In-Memory-OLTP ist ein speicheroptimiertes Datenbankmodul, das in das SQL Server-Modul integriert und für die Transaktionsverarbeitung optimiert ist. In-Memory-OLTP kann die Leistung von OLTP-Datenbankanwendungen spürbar verbessern. Diese Funktion verbessert den Durchsatz und reduziert die Latenz bei der Transaktionsverarbeitung, und verbessert die Leistung von vorübergehenden Datenszenarien wie temporäre Tabellen und ETL.

Weitere Informationen finden Sie unter [In-Memory OLTP \(Arbeitsspeicheroptimierung\)](#).

Columnstore

In SQL Server 2016 wurde die operative Echtzeitanalyse eingeführt, um Analyse- und OLTP-Workloads gleichzeitig in derselben Datenbanktabelle ausführen zu können. Sie können nicht nur Echtzeitanalysen durchführen, sondern sind auch nicht mehr auf ETL und ein Data Warehouse für Ihre operativen Workloads angewiesen.



In der Vergangenheit verwendeten Unternehmen meistens separate Systeme für operative und Analyseworkloads. In solchen Systemen werden ETL-Aufträge (Extrahieren, Transformieren, Laden) eingesetzt, um die Daten regelmäßig aus dem operativen Speicher in den Analysespeicher zu verschieben. Die Analysedaten werden in der Regel in einem Data Warehouse oder Data Mart gespeichert, die für die Ausführung von Analyseabfragen reserviert sind. Diese Lösung galt lange Zeit als Standard, hat jedoch drei wichtige Nachteile:

Komplexität. Die Implementierung von ETL erfordert einen beachtlichen Programmieraufwand, insbesondere beim Laden der bearbeiteten Zeilen. Es ist nicht einfach, herauszufinden, welche Zeilen geändert wurden.

Kosten. Für die Implementierung von ETL müssen Hardware und Softwarelizenzen eingekauft werden.

Datenlatenz. ETL ist immer mit einer Zeitverzögerung bei der Analyse verbunden. Wenn der ETL-Auftrag beispielsweise am Ende eines jeden Werktags ausgeführt wird, sind die Daten aus den Analyseabfragen immer mindestens einen Tag alt. Für viele Unternehmen ist diese Verzögerung inakzeptabel, da die Daten in Echtzeit analysiert werden müssen. Für die Betrugserkennung müssen operative Daten beispielsweise in Echtzeit analysiert werden.

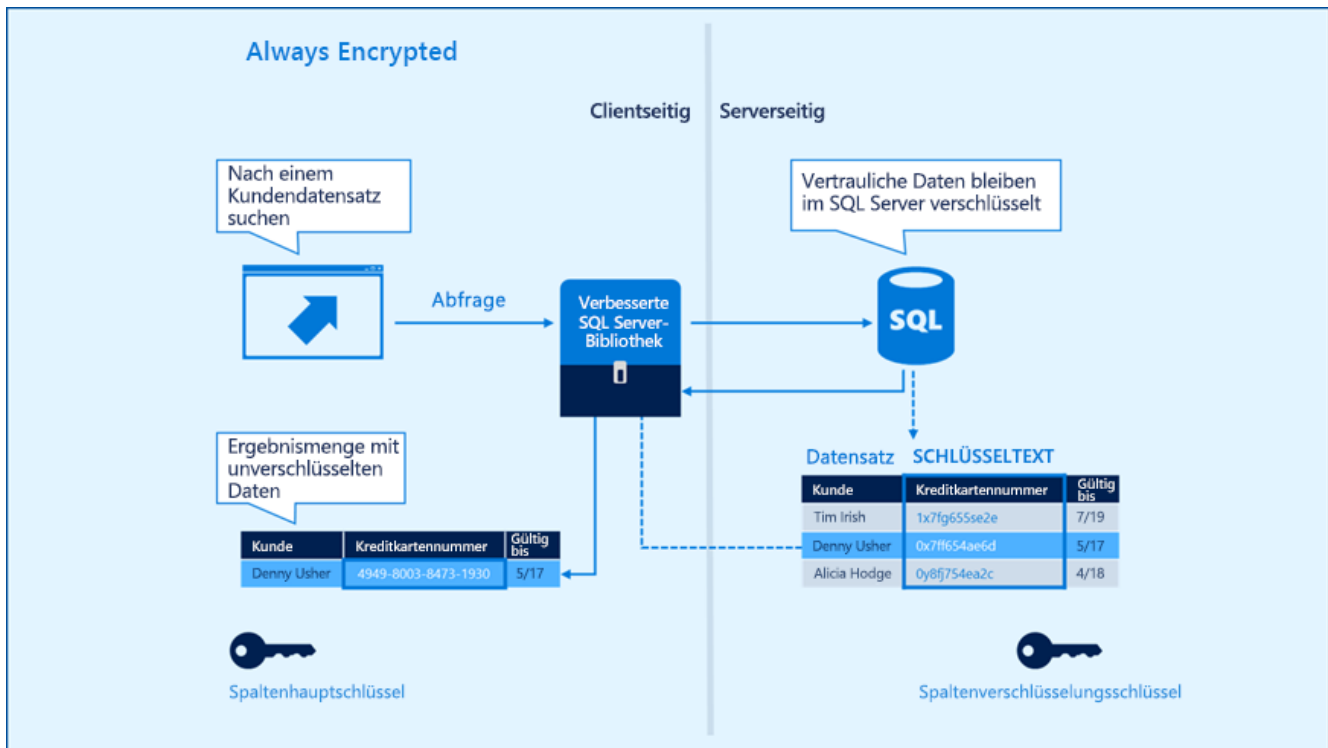
Die operative Echtzeitanalyse bietet eine Lösung für diese Probleme. Wenn Analyse- und OLTP-Arbeitslasten auf derselben zugrunde liegenden Tabelle ausgeführt werden, treten keine Verzögerungen auf. In Szenarien mit Echtzeitanalysen können Kosten und Komplexität deutlich reduziert werden, da kein ETL mehr benötigt wird und kein separates Data Warehouse eingekauft und gewartet werden muss.

Weitere Informationen finden Sie unter [Beschreibung von Columnstore-Indizes](#).

Always Encrypted

Mit Always Encrypted können Sie sensible Informationen wie Kreditkartennummern oder persönliche Identifikationsdaten (z. B. US-Sozialversicherungsnummern) sicher in Azure SQL Database- oder SQL Server-Datenbanken speichern. Mit Always Encrypted können Kunden vertrauliche Daten in ihren Anwendungen verschlüsseln, ohne die Verschlüsselungsschlüssel jemals an das Datenbankmodul (SQL Database oder SQL Server) weiterzugeben. Auf diese Weise ermöglicht Always Encrypted eine Trennung zwischen denjenigen, die die Daten besitzen und abrufen können, und denjenigen, die die Daten verwalten, jedoch keinen Zugriff haben sollten. Always Encrypted garantiert, dass lokale Datenbankadministratoren, Clouddatenbankbetreiber und andere nicht

autorisierte Benutzer mit umfassenden Privilegien nicht auf die verschlüsselten Daten zugreifen können, und ermöglicht den Kunden so die sichere Speicherung vertraulicher Daten außerhalb ihrer direkten Kontrolle. Auf diese Weise können Unternehmen ihre ruhenden und übertragenen Daten für die Speicherung auf Azure verschlüsseln, die Administration lokaler Datenbanken an externe Parteien auslagern und die Sicherheitsanforderungen für ihre eigenen DBA-Mitarbeiter reduzieren. Weitere Informationen finden Sie unter [Always Encrypted](#).



Transparent Data Encryption

Transparent Data Encryption (TDE) verschlüsselt SQL Server-, Azure SQL Database- und Azure SQL Data Warehouse-Daten. Dies wird auch als Verschlüsselung ruhender Daten bezeichnet. Unternehmen können verschiedene Maßnahmen zur Sicherung der Datenbank ergreifen, z. B. ein sicheres System entwickeln, vertrauliche Bestände verschlüsseln und eine Firewall um die Datenbankserver herum errichten. Wenn jedoch die physischen Medien wie z. B. Laufwerke oder Sicherungsbänder gestohlen werden, können Angreifer die Datenbank wiederherstellen oder anfügen und auf die Daten zugreifen. Dies kann verhindert werden, indem die vertraulichen Daten in der Datenbank verschlüsselt und die Verschlüsselungsschlüssel mit einem Zertifikat gesichert werden. Auf diese Weise können nur Personen im Besitz des Schlüssels auf die Daten zugreifen, aber diese Lösung muss im Voraus geplant werden.

TDE führt Echtzeit-E/A-Verschlüsselung und -Entschlüsselung der Daten- und Protokolldateien durch. Bei der Verschlüsselung kommt ein Schlüssel zur Datenbankverschlüsselung (DEK) zum Einsatz, der zur besseren Verfügbarkeit bei der Wiederherstellung im Boot-Datensatz der Datenbank gespeichert ist. Der DEK ist ein symmetrischer Schlüssel, der mit einem in der Master-Datenbank auf dem Server abgelegten Zertifikat gesichert wird, oder ein asymmetrischer Schlüssel, der von einem Verwaltungsmodul für Verschlüsselungsschlüssel geschützt wird. Weitere Informationen finden Sie unter [Transparent Data Encryption \(TDE\)](#).

Sicherheit auf Zeilenebene

Mit der Sicherheit auf Zeilenebene (Row-Level Security, RLS) können Kunden den Zugriff auf die Zeilen in einer Datenbanktabelle anhand der Eigenschaften des Benutzers steuern, der die Abfrage ausführt (z. B. anhand der Gruppenmitgliedschaft oder des Ausführungskontexts).

Die Sicherheit auf Zeilenebene vereinfacht das Design und die Entwicklung der Sicherheit in einer Anwendung. Mit der Sicherheit auf Zeilenebene können Unternehmen außerdem Einschränkungen für den Zugriff auf Datenzeilen implementieren. Ein Unternehmen kann beispielsweise sicherstellen, dass die Mitarbeiter nur auf Daten zugreifen können, die für ihre Abteilung relevant sind, oder den Datenzugriff eines Kunden auf die Daten beschränken, die für dessen Unternehmen relevant sind.

Die Zugriffsbeschränkungslogik befindet sich in der Datenbankebene, und nicht separat von den Daten in einer anderen Anwendungsebene. Das Datenbanksystem wendet die Zugriffsbeschränkungen immer dann an, wenn versucht wird, von einer anderen Ebene auf die Daten zuzugreifen. Auf diese Weise ist das Sicherheitssystem zuverlässiger und robuster, da die Angriffsfläche reduziert wird. Weitere Informationen finden Sie unter [Sicherheit auf Zeilenebene](#).

Dynamische Datenmaskierung

Die dynamische Datenmaskierung (Dynamic Data Masking, DDM) maskiert Daten für unberechtigte Benutzer, um die Risiken für vertrauliche Daten zu reduzieren. Diese Funktion vereinfacht das Design und die Entwicklung der Sicherheit in einer Anwendung deutlich.

Mit der dynamischen Datenmaskierung können Unternehmen festlegen, welcher Teil ihrer vertraulichen Daten offengelegt wird, um den unbefugten Zugriff auf diese Daten zu verhindern. Außerdem hat diese Funktion nur minimale Auswirkungen auf die Anwendungsebene. Die dynamische Datenmaskierung kann in der Datenbank konfiguriert werden, um vertrauliche Daten in den Result Sets von Abfragen für bestimmte Datenbankfelder auszublenden, ohne die Daten in der Datenbank zu ändern. Da die Maskierungsregeln in den Abfrageergebnissen angewendet werden, kann die dynamische Datenmaskierung problemlos mit vorhandenen Anwendungen eingesetzt werden. In vielen Anwendungen können vertrauliche Daten ohne Änderungen an vorhandenen Abfragen maskiert werden. Weitere Informationen finden Sie unter [dynamische Datenmaskierung](#).

Auditing

Mit SQL Server Audit können Sie Ereignisse nachverfolgen und protokollieren, die in einer Instanz des Datenbankmoduls oder in einer einzelnen Datenbank auftreten. Serverüberwachungen enthalten bestimmte Spezifikationen für Ereignisse auf der Serverebene, und Datenbanküberwachungen enthalten Spezifikationen für Ereignisse auf der Datenbankebene. Die überwachten Ereignisse können in die Ereignisprotokolle oder in Überwachungsdateien geschrieben werden.

Je nach gesetzlichen oder standardisierten Anforderungen für Ihre Installation bietet SQL Server verschiedene Überwachungsebenen an. SQL Server Audit enthält die benötigten Tools und Prozesse zum Aktivieren, Speichern und Abrufen der Überwachungen für verschiedene Server- und Datenbankobjekte. Weitere Informationen finden Sie unter [SQL Server Audit](#).

Erste Schritte mit SQL Server 2017 unter Linux

SQL Server 2017 wird unter Red Hat Enterprise Linux (RHEL), SUSE Enterprise Server (SLES) und Ubuntu unterstützt. SQL Server 2017 ist auch als Docker-Image erhältlich, das mit dem Docker-Modul unter Linux oder unter Docker für Windows/Mac ausgeführt werden kann. Das folgende Beispiel verwendet Red Hat Enterprise Linux. Eine ausführliche Anleitung zur Installation von SQL Server 2017 und der benötigten Tools für die unterstützten Linux-Distributionen und Container finden Sie unter [Installationsleitfaden für SQL Server on Linux](#).

Die Installation von SQL Server 2017 erfolgt schnell und einfach. Sie können den Paketmanager für Ihr jeweiliges BS verwenden. Zunächst müssen Sie die Konfigurationsdatei für das Repository herunterladen. Außerdem benötigen Sie einen Benutzer mit **sudo**-Zugriff.

```
sudo curl -o /etc/yum.repos.d/mssql-server.repo  
https://packages.microsoft.com/config/rhel/7/mssql-server.repo
```

Führen Sie die folgenden Befehle aus, um SQL Server zu installieren:

```
sudo yum update  
sudo yum install -y mssql-server
```

Führen Sie nach Abschluss der Paketinstallation den Befehl **mssql-conf setup** aus, und folgen Sie den Anweisungen, um das Kennwort für den Systemadministrator (SA) festzulegen und Ihre Edition auszuwählen.

```
sudo /opt/mssql/bin/mssql-conf setup
```

Schlussfolgerung

SQL Server 2017 ist ein Durchbruch für SQL Server, und bringt die branchenführenden Leistungs- und Sicherheitsfunktionen von SQL Server zu Linux und Docker-Containern. Mit den neuen Features wie der adaptiven Abfrageverarbeitung und der automatischen Plankorrektur können Unternehmen ihre Datenverarbeitungskapazitäten weiter optimieren, und mit SQL Graph können Beziehungen in einer Graphenstruktur abgebildet und abgefragt werden, anstatt wie bisher mit einem herkömmlichen relationalen Schema. SQL Server 2017 bietet Unternehmen totale Freiheit bei der Erstellung oder Erweiterung ihrer Datenökosysteme über unterschiedliche Betriebssysteme und Programmiersprachen hinweg.

Handlungsempfehlungen

Weitere Informationen zu SQL Server finden Sie unter <https://docs.microsoft.com/de-de/sql/index>

Laden Sie eine Testversion unter <https://www.microsoft.com/de-de/sql-server/sql-server-downloads> herunter

Feedback

War dieses Whitepaper für Sie hilfreich? Wir würden uns über Ihr Feedback freuen. Bitte bewerten Sie das Whitepaper auf einer Skala von 1 (schlecht) bis 5 (hervorragend), und nennen Sie uns die Gründe für diese Bewertung. Genauer gesagt:

- Bewerten Sie das Whitepaper als gut, weil es relevante Beispiele oder hilfreiche Screenshots enthält, der Text gut verständlich formuliert ist usw.?
- Bewerten Sie das Whitepaper als schlecht, weil die Beispiele für Ihre Belange nicht relevant sind, die Screenshots unscharf sind, der Schreibstil zu wünschen übrig lässt usw.?

Ihr Feedback hilft uns, die Qualität der von uns veröffentlichten Whitepaper zu verbessern.

Bitte senden Sie Ihr Feedback an: sqlsrvwpfeedback@microsoft.com