



Migration der Oracle Datenbank

Enterprise applications

NetApp
February 10, 2026

Inhalt

Migration der Oracle Datenbank	1
Überblick	1
Skripte	1
Migrationsplanung	1
Überlegungen zur Planung	2
Datengröße	2
Anzahl der Datenbanken	2
Anforderungen neu architekurgerecht	3
Umstellungszeit	3
Rückweg	4
Probe	4
Verfahren	4
Überblick	4
Migration von Datendateien	13
Protokollversand	14
Host-Daten werden kopiert	50
Import fremder LUNs	89
Beispielskripts	111
Datenbank wird heruntergefahren	111
Starten der Datenbank	112
Konvertieren Sie das Dateisystem in schreibgeschützt	113
Ersetzen Sie das Dateisystem	114
Automatisierte Datenbankmigration	116
Dateispeicherorte anzeigen	118
Bereinigung der ASM-Migration	119
Namenskonvertierung von ASM in Dateisystem	121
Wiedergabe von Protokollen in der Datenbank	123
Wiedergabe von Protokollen in der Standby-Datenbank	123

Migration der Oracle Datenbank

Überblick

Die Nutzung der Leistungsfähigkeit einer neuen Storage-Plattform erfordert zwingend die Speicherung der Daten in dem neuen Storage-System. Mit ONTAP ist der Migrationsprozess denkbar einfach: Migrationen und Upgrades von ONTAP zu ONTAP, Import fremder LUNs und Verfahren für die direkte Nutzung des Host-Betriebssystems oder der Oracle Datenbanksoftware.



Diese Dokumentation ersetzt den bereits veröffentlichten technischen Bericht *TR-4534: Migration von Oracle-Datenbanken zu NetApp-Speichersystemen*

Im Falle eines neuen Datenbankprojekts ist dies kein Problem, da die Datenbank- und Anwendungsumgebungen eingerichtet sind. Die Migration stellt Unternehmen jedoch vor besondere Herausforderungen, was die Unterbrechung des Geschäftsbetriebs, den für den Abschluss der Migration erforderlichen Zeitaufwand, die erforderlichen Fachkompetenzen und die Risikominimierung angeht.

Skripte

In dieser Dokumentation sind Beispielskripte enthalten. Diese Skripte bieten Beispielmethode zur Automatisierung verschiedener Aspekte der Migration, um das Risiko von Benutzerfehlern zu verringern. Die Skripte können die Gesamtanforderungen an die für eine Migration verantwortlichen IT-Mitarbeiter verringern und den Gesamtprozess beschleunigen. Diese Skripte stammen alle aus Migrationsprojekten, die von NetApp Professional Services und NetApp-Partnern durchgeführt werden. Beispiele für deren Verwendung sind in dieser Dokumentation aufgeführt.

Migrationsplanung

Die Oracle-Datenmigration kann auf einer der drei Ebenen erfolgen: Der Datenbank, dem Host oder dem Storage Array.

Die Unterschiede liegen darin, welche Komponente der Gesamtlösung für das Verschieben von Daten verantwortlich ist: Die Datenbank, das Host-Betriebssystem oder das Speichersystem.

Die Abbildung unten zeigt ein Beispiel für die Migrationsebenen und den Datenfluss. Bei einer Migration auf Datenbankebene werden die Daten vom ursprünglichen Storage-System über die Host- und Datenbankschichten in die neue Umgebung verschoben. Die Migration auf Host-Ebene ist ähnlich, aber die Daten durchlaufen nicht die Applikationsebene und werden stattdessen mithilfe von Host-Prozessen an den neuen Speicherort geschrieben. Und schließlich ist bei der Migration auf Storage-Ebene ein Array wie ein NetApp FAS System für die Datenverschiebung verantwortlich.

[Fehler: Fehlendes Grafikbild]

Eine Migration auf Datenbankebene bezieht sich im Allgemeinen auf die Verwendung von Oracle-Protokollversand über eine Standby-Datenbank, um eine Migration auf der Oracle-Ebene durchzuführen. Migrationen auf Host-Ebene werden mithilfe der nativen Funktionen der Konfiguration des Host-Betriebssystems durchgeführt. Diese Konfiguration umfasst Dateikopiervorgänge mit Befehlen wie CP, tar und Oracle Recovery Manager (RMAN) oder mit einem Logical Volume Manager (LVM) zur Verlagerung der zugrunde liegenden Bytes eines Dateisystems. Oracle Automatic Storage Management (ASM) wird als

Funktion auf Hostebene kategorisiert, da sie unter der Ebene der Datenbankanwendung ausgeführt wird. ASM tritt an die Stelle des üblichen Logical Volume Managers auf einem Host. Und schließlich können Daten auf Storage-Array-Ebene migriert werden, d. h. unter der Ebene des Betriebssystems.

Überlegungen zur Planung

Die beste Option für die Migration hängt von einer Kombination verschiedener Faktoren ab: Vom Umfang der zu migrierenden Umgebung, der Notwendigkeit, Ausfallzeiten zu vermeiden, und dem für die Migration erforderlichen Gesamtaufwand. Große Datenbanken erfordern offensichtlich mehr Zeit und Aufwand für die Migration, aber die Komplexität einer solchen Migration ist minimal. Kleine Datenbanken können schnell migriert werden. Wenn jedoch Tausende migriert werden müssen, kann das Ausmaß des Aufwands zu Komplikationen führen. Und je größer die Datenbank ist, desto wahrscheinlicher ist sie, dass sie geschäftskritisch ist. Dies führt dazu, dass Ausfallzeiten minimiert werden müssen, während gleichzeitig ein Back-Out-Pfad beibehalten wird.

Im Folgenden werden einige Überlegungen zur Planung einer Migrationsstrategie erörtert.

Datengröße

Die Größe der zu migrierenden Datenbanken wirkt sich natürlich auf die Migrationsplanung aus. Die Größe wirkt sich jedoch nicht unbedingt auf die Umstellungszeit aus. Wenn große Datenmengen migriert werden müssen, kommt es in erster Linie auf die Bandbreite an. Kopiervorgänge werden in der Regel mit effizienten sequenziellen I/O-Vorgängen ausgeführt. Gehen Sie bei Kopiervorgängen von einer Auslastung der verfügbaren Netzwerkbandbreite von 50 % aus. Ein 8-GB-FC-Port kann theoretisch etwa 800 Mbit/s übertragen. Bei einer Auslastung von 50 % kann eine Datenbank mit einer Geschwindigkeit von etwa 400 Mbps kopiert werden. Somit kann eine 10-TB-Datenbank innerhalb von etwa sieben Stunden kopiert werden.

Eine Migration über größere Entfernungen erfordert in der Regel einen kreativen Ansatz, wie der Protokollversand-Prozess in erläutert ["Online-Datendatei verschieben"](#). IP-Netzwerke über große Entfernungen verfügen selten überall in der Nähe von LAN- oder SAN-Geschwindigkeiten über eine entsprechende Bandbreite. In einem Fall unterstützte NetApp die Fernmigration einer 220 TB großen Datenbank mit sehr hohen Archiv- Log-Generierungsraten. Der gewählte Ansatz für die Datenübertragung war der tägliche Versand von Bändern, da diese Methode die maximal mögliche Bandbreite bot.

Anzahl der Datenbanken

In vielen Fällen liegt das Problem beim Verschieben großer Datenmengen nicht in der Datengröße, sondern in der Komplexität der Konfiguration, die die Datenbank unterstützt. Wenn man einfach nur weiß, dass 50 TB Datenbanken migriert werden müssen, reicht das nicht aus. Dabei kann es sich um eine einzelne 50 TB große geschäftskritische Datenbank, eine Sammlung von 4, 000 älteren Datenbanken oder eine Kombination aus Produktions- und nicht produktiven Daten handeln. In manchen Fällen besteht ein Großteil der Daten aus Klone einer Quelldatenbank. Diese Klone müssen überhaupt nicht migriert werden, da sie einfach wiederhergestellt werden können. Dies gilt insbesondere dann, wenn die neue Architektur die Nutzung von NetApp FlexClone Volumes ermöglicht.

Für die Migrationsplanung müssen Sie verstehen, wie viele Datenbanken im Umfang enthalten sind und wie sie priorisiert werden müssen. Mit zunehmender Anzahl an Datenbanken ist die bevorzugte Migrationsoption in der Regel im Stack immer geringer. So kann zum Beispiel das Kopieren einer einzelnen Datenbank mit RMAN problemlos und bei einem kurzen Ausfall durchgeführt werden. Dies ist die Replizierung auf Host-Ebene.

Wenn es 50 Datenbanken gibt, kann es einfacher sein, die Einrichtung einer neuen Dateisystemstruktur zu vermeiden, um eine RMAN-Kopie zu erhalten und stattdessen die Daten an die Stelle zu verschieben. Dies kann durch hostbasierte LVM-Migration erfolgen, um Daten von alten LUNs auf neue LUNs zu verschieben. Dadurch wird die Verantwortung vom Datenbankadministratorteam (DBA) an das Betriebssystemteam

übertragen und die Daten werden in Bezug auf die Datenbank transparent migriert. Die Dateisystemkonfiguration ist unverändert.

Wenn schließlich 500 Datenbanken von 200 Servern migriert werden müssen, können speicherbasierte Optionen wie die ONTAP Funktion zum Importieren fremder LUNs (Foreign LUN Import, FLI) verwendet werden, um eine direkte Migration der LUNs durchzuführen.

Anforderungen neu architekturgerecht

In der Regel muss das Layout einer Datenbankdatei verändert werden, um die Funktionen des neuen Storage Array nutzen zu können. Dies ist jedoch nicht immer der Fall. Die Funktionen der EF-Series All-Flash-Arrays richten sich beispielsweise primär an SAN-Performance und SAN-Zuverlässigkeit. In den meisten Fällen können Datenbanken auf ein EF-Series Array migriert werden, ohne dass dabei besondere Überlegungen zum Datenlayout angestellt werden müssen. Die einzigen Anforderungen sind hohe IOPS, niedrige Latenz und robuste Zuverlässigkeit. Wenngleich es Best Practices für Faktoren wie RAID-Konfiguration oder Dynamic Disk Pools gibt, sind für EF-Series Projekte kaum nennenswerte Änderungen an der gesamten Storage-Architektur erforderlich, um diese Funktionen nutzen zu können.

Im Gegensatz dazu erfordert die Migration zu ONTAP in der Regel eine stärkere Berücksichtigung des Datenbanklayouts, um sicherzustellen, dass die endgültige Konfiguration den größtmöglichen Nutzen erzielt. ONTAP bietet für eine Datenbankumgebung bereits viele Funktionen ohne besondere Architekturanstrengungen. Am wichtigsten ist jedoch die Möglichkeit einer unterbrechungsfreien Migration auf neue Hardware, wenn die aktuelle Hardware das Ende ihres Lebenszyklus erreicht. Generell gilt: Eine Migration zu ONTAP ist die letzte Migration, die Sie durchführen müssen. Nachfolgende Hardware Upgrades werden durchgeführt und die Daten werden unterbrechungsfrei auf neue Medien migriert.

Bei einigen Planungen sind noch mehr Vorteile verfügbar. Die wichtigsten Überlegungen beziehen sich auf die Verwendung von Snapshots. Snapshots bilden die Grundlage für nahezu sofortige Backups, Restores und Klonvorgänge. Ein Beispiel für die Leistung von Snapshots ist der größte bekannte Einsatz bei einer einzigen Datenbank mit 996 TB, die auf ca. 250 LUNs auf 6 Controllern ausgeführt wird. Diese Datenbank kann innerhalb von 2 Minuten gesichert, innerhalb von 2 Minuten wiederhergestellt und innerhalb von 15 Minuten geklont werden. Zu den weiteren Vorteilen zählen die Möglichkeit, Daten im Cluster als Reaktion auf Workload-Änderungen zu verschieben, sowie die Anwendung von Quality-of-Service-Kontrollen (QoS), um in einer Umgebung mit mehreren Datenbanken eine gute und konsistente Performance zu erreichen.

Technologien wie QoS-Steuerung, Datenverlagerung, Snapshots und Klone arbeiten in nahezu jeder Konfiguration. Allerdings ist einige Überlegungen im Allgemeinen erforderlich, um die Vorteile zu maximieren. In einigen Fällen können Änderungen am Design von Datenbank-Storage-Layouts erforderlich sein, um die Investitionen in das neue Speicher-Array zu maximieren. Solche Designänderungen können sich auf die Migrationsstrategie auswirken, da Host- oder Storage-basierte Migrationen das ursprüngliche Datenlayout replizieren. Weitere Schritte sind möglicherweise erforderlich, um die Migration abzuschließen und ein für ONTAP optimiertes Daten-Layout zu liefern. Die in dargestellten Verfahren "[Oracle-Migrationsverfahren – Überblick](#)" Und später zeigen einige der Methoden, um nicht nur eine Datenbank zu migrieren, sondern sie mit minimalem Aufwand in das optimale finale Layout zu migrieren.

Umstellungszeit

Der maximal zulässige Service-Ausfall während der Umstellung sollte ermittelt werden. Es ist ein häufiger Fehler anzunehmen, dass der gesamte Migrationsprozess zu Störungen führt. Viele Aufgaben können vor Beginn von Serviceunterbrechungen durchgeführt werden. Viele Optionen ermöglichen den Abschluss der Migration ohne Unterbrechungen oder Ausfälle. Auch wenn Unterbrechungen unvermeidbar sind, müssen Sie dennoch den maximal zulässigen Serviceausfall definieren, da die Dauer der Umstellungszeit von Prozedur zu Prozedur variiert.

Das Kopieren einer 10-TB-Datenbank dauert beispielsweise in der Regel ungefähr sieben Stunden. Wenn das Unternehmen einen Ausfall von sieben Stunden zulassen muss, ist Dateikopien eine einfache und sichere Möglichkeit für die Migration. Wenn fünf Stunden nicht akzeptabel sind, lässt sich ein einfacher Protokollversand-Prozess wie (siehe) "[Oracle-Protokollversand](#)") Kann mit minimalem Aufwand eingerichtet werden, um die Umstellungszeit auf etwa 15 Minuten zu reduzieren. Während dieser Zeit kann ein Datenbankadministrator den Prozess abschließen. Wenn 15 Minuten nicht akzeptabel sind, kann der endgültige Umstellungsprozess durch Skripting automatisiert werden, um die Umstellungszeit auf wenige Minuten zu verkürzen. Sie können eine Migration jederzeit beschleunigen, doch dies kostet Zeit und Aufwand. Die Umstellungszeitziele sollten darauf basieren, was für das Unternehmen akzeptabel ist.

Rückweg

Keine Migration ist völlig risikolos. Auch wenn die Technik einwandfrei funktioniert, besteht immer die Möglichkeit eines Anwenderfehlers. Das mit einem ausgewählten Migrationspfad verbundene Risiko muss neben den Folgen einer fehlgeschlagenen Migration berücksichtigt werden. Die transparente Online-Storage-Migrationsfunktion von Oracle ASM ist beispielsweise eines der wichtigsten Merkmale, und diese Methode ist eine der zuverlässigsten. Mit dieser Methode werden die Daten jedoch irreversibel kopiert. In dem sehr unwahrscheinlichen Fall, dass ein Problem mit ASM auftritt, gibt es keinen einfachen Rückweg. Die einzige Option besteht darin, entweder die ursprüngliche Umgebung wiederherzustellen oder die Migration mit ASM zurück zu den ursprünglichen LUNs rückgängig zu machen. Das Risiko kann durch ein Backup vom Typ Snapshot auf dem ursprünglichen Storage-System minimiert, aber nicht sogar ganz beseitigt werden, vorausgesetzt, das System ist in der Lage, einen solchen Vorgang auszuführen.

Probe

Einige Migrationsverfahren müssen vor der Ausführung vollständig überprüft werden. Eine Migration und eine Generalprobe des Umstellungsprozesses ist eine häufige Anfrage bei geschäftskritischen Datenbanken, bei denen die Migration erfolgreich sein und die Downtime minimiert werden muss. Zudem gehören auch die Anwenderakzeptanztests häufig zu den Aufgaben nach der Migration, und das gesamte System kann erst nach Abschluss der Tests in die Produktionsumgebung zurückgeführt werden.

Wenn es Bedarf an Proben gibt, können verschiedene ONTAP Funktionen den Prozess wesentlich vereinfachen. Snapshots können insbesondere eine Testumgebung zurücksetzen und schnell mehrere platzsparende Kopien einer Datenbankumgebung erstellen.

Verfahren

Überblick

Für die Oracle-Migrationsdatenbank sind zahlreiche Verfahren verfügbar. Das richtige hängt von Ihren geschäftlichen Anforderungen ab.

In vielen Fällen haben Systemadministratoren und DBAs ihre eigenen bevorzugten Methoden, um physische Volume-Daten zu verschieben, zu spiegeln und zu demirrieren oder Oracle RMAN zum Kopieren von Daten zu nutzen.

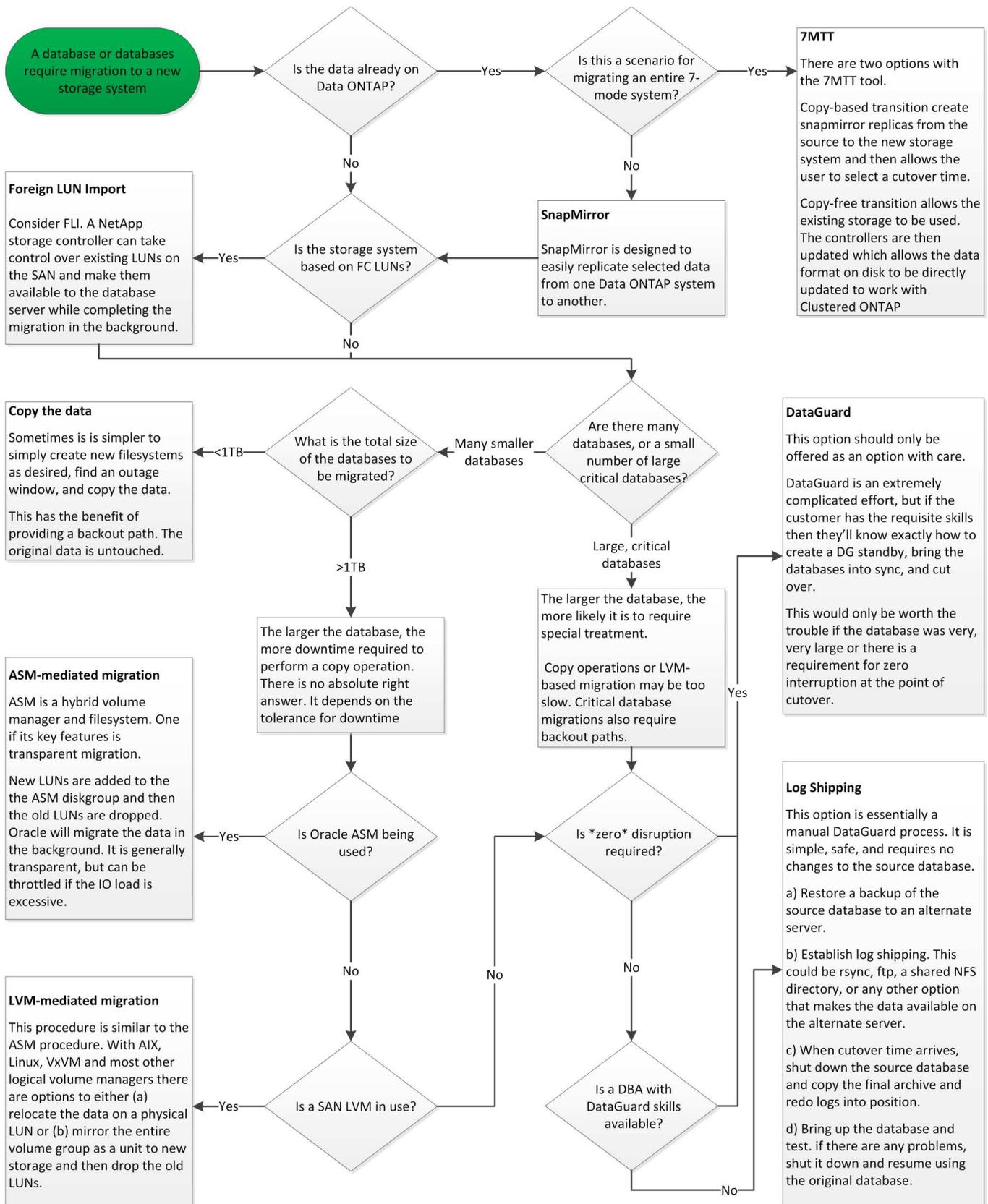
Diese Verfahren dienen in erster Linie als Orientierungshilfe für IT-Mitarbeiter, die mit einigen der verfügbaren Optionen nicht vertraut sind. Des Weiteren werden die Aufgaben, der zeitliche Bedarf und der Qualifikationsbedarf für jeden Migrationsansatz dargestellt. Dadurch können auch andere Parteien wie NetApp und Partner Professional Services oder IT-Management die Anforderungen an die einzelnen Verfahren voll einschätzen.

Es gibt keine einzigen Best Practices für die Erstellung einer Migrationsstrategie. Um einen Plan zu erstellen,

müssen zunächst die Verfügbarkeitsoptionen verstanden und anschließend die Methode ausgewählt werden, die den Anforderungen des Unternehmens am besten entspricht. Die folgende Abbildung zeigt die grundlegenden Überlegungen und typischen Schlussfolgerungen von Kunden, ist aber nicht universell auf alle Situationen anwendbar.

Ein Schritt wirft beispielsweise das Problem der Gesamtgröße der Datenbank auf. Der nächste Schritt hängt davon ab, ob die Datenbank mehr oder weniger als 1 TB umfasst. Die empfohlenen Schritte sind genau das – Empfehlungen auf der Basis typischer Kundenpraktiken. Die meisten Kunden würden nicht mit DataGuard eine kleine Datenbank kopieren, aber einige könnten. Die meisten Kunden würden aufgrund der erforderlichen Zeit nicht versuchen, eine 50 TB große Datenbank zu kopieren, aber einige haben möglicherweise ein ausreichend großes Wartungsfenster, um einen solchen Vorgang zu ermöglichen.

Das folgende Flussdiagramm zeigt die Arten von Überlegungen, welche Migrationspfade am besten geeignet sind. Sie können mit der rechten Maustaste auf das Bild klicken und es in einer neuen Registerkarte öffnen, um die Lesbarkeit zu verbessern.



Online-Datendatei verschieben

Bei Oracle 12cR1 und höher kann eine Datendatei verschoben werden, während die Datenbank online bleibt. Es funktioniert außerdem zwischen verschiedenen Dateisystemtypen. Eine Datendatei kann beispielsweise

von einem xfs-Dateisystem in ASM verschoben werden. Diese Methode wird im Allgemeinen nicht in der Größenordnung verwendet, da die Anzahl der erforderlichen individuellen Datendateiverschiebungsvorgänge erforderlich wäre. Es ist jedoch eine Option, die es sich lohnt, bei kleineren Datenbanken mit weniger Datendateien in Betracht zu ziehen.

Darüber hinaus ist das einfache Verschieben einer Datendatei eine gute Option für die Migration von Teilen vorhandener Datenbanken. Beispielsweise können weniger aktive Datendateien auf kostengünstigeren Storage verschoben werden, beispielsweise auf ein FabricPool Volume, mit dem ungenutzte Blöcke im Objektspeicher gespeichert werden können.

Migration auf Datenbankebene

Die Migration auf Datenbankebene bedeutet, dass die Datenbank Daten verschieben kann. Konkret bedeutet dies Protokollversand. Technologien wie RMAN und ASM sind Oracle Produkte. Im Rahmen der Migration arbeiten sie jedoch auf Hostebene, wo sie Dateien kopieren und Volumes managen.

Protokollversand

Die Grundlage für die Migration auf Datenbankebene ist das Oracle Archivprotokoll, das ein Protokoll der Änderungen an der Datenbank enthält. Meistens ist ein Archivprotokoll Bestandteil einer Backup- und Recovery-Strategie. Der Recovery-Prozess beginnt mit der Wiederherstellung einer Datenbank und dann mit der Wiedergabe eines oder mehrerer Archivprotokolle, um die Datenbank in den gewünschten Zustand zu bringen. Mit derselben Basistechnologie kann eine Migration mit nur minimaler bis keiner Unterbrechung des Betriebs durchgeführt werden. Noch wichtiger ist, dass diese Technologie die Migration ermöglicht und gleichzeitig die ursprüngliche Datenbank unberührt lässt. Dabei wird ein Back-Out-Pfad beibehalten.

Der Migrationsprozess beginnt mit der Wiederherstellung eines Datenbank-Backups auf einem sekundären Server. Dies kann auf unterschiedliche Weise erfolgen, doch die meisten Kunden verwenden ihre normale Backup-Applikation, um die Datendateien wiederherzustellen. Nachdem die Datendateien wiederhergestellt sind, legen Benutzer eine Methode für den Protokollversand fest. Das Ziel besteht darin, einen konstanten Feed von Archivprotokollen zu erstellen, die von der primären Datenbank generiert werden, und diese in der wiederhergestellten Datenbank wiederzugeben, um sie nahe am selben Status zu halten. Wenn die Umstellung ankommt, wird die Quelldatenbank vollständig heruntergefahren und die letzten Archivprotokolle sowie in einigen Fällen die Wiederherstellungsprotokolle kopiert und wiedergegeben. Es ist wichtig, dass die Wiederherstellungsprotokolle auch berücksichtigt werden, da sie einige der letzten abgeschlossenen Transaktionen enthalten können.

Nachdem diese Protokolle übertragen und wiedergegeben wurden, sind beide Datenbanken konsistent. Jetzt führen die meisten Kunden einige grundlegende Tests durch. Wenn während des Migrationsprozesses Fehler auftreten, sollte die Protokollwiedergabe Fehler melden und fehlschlagen. Es ist weiterhin ratsam, einige schnelle Tests basierend auf bekannten Abfragen oder applikationsgestützten Aktivitäten durchzuführen, um zu überprüfen, ob die Konfiguration optimal ist. Es ist auch üblich, eine abschließende Testtabelle zu erstellen, bevor die ursprüngliche Datenbank heruntergefahren wird, um zu überprüfen, ob sie in der migrierten Datenbank vorhanden ist. Dieser Schritt stellt sicher, dass während der endgültigen Protokollsynchronisierung keine Fehler gemacht wurden.

Eine einfache Log-Shipping-Migration kann Out-of-Band hinsichtlich der ursprünglichen Datenbank konfiguriert werden, was dies besonders für geschäftskritische Datenbanken nützlich macht. Für die Quelldatenbank sind keine Konfigurationsänderungen erforderlich, und die Wiederherstellung und Erstkonfiguration der Migrationsumgebung haben keine Auswirkungen auf den Produktionsbetrieb. Nachdem der Protokollversand konfiguriert wurde, werden einige I/O-Anforderungen an die Produktionsserver gestellt. Der Protokollversand besteht jedoch aus einfachen sequenziellen Lesevorgängen in den Archivprotokollen, was sich wahrscheinlich nicht auf die Performance der Produktionsdatenbank auswirken wird.

Der Protokollversand hat sich besonders für große Entfernungen bei Migrationen mit hohen Änderungsraten

bewährt. In einer Instanz wurde eine einzelne 220 TB Datenbank an einen etwa 500 Meilen entfernten neuen Standort migriert. Die Änderungsrate war extrem hoch und Sicherheitsbeschränkungen verhinderten die Nutzung einer Netzwerkverbindung. Der Protokollversand wurde durch Tape und Kurier durchgeführt. Eine Kopie der Quelldatenbank wurde zunächst mithilfe der unten beschriebenen Verfahren wiederhergestellt. Die Protokolle wurden dann wöchentlich per Kurier bis zum Zeitpunkt der Umstellung versendet, als die endgültigen Tapes zugestellt wurden und die Protokolle auf die Replikatdatenbank angewendet wurden.

Oracle DataGuard

In einigen Fällen ist eine vollständige DataGuard Umgebung gerechtfertigt. Es ist falsch, den Begriff DataGuard zu verwenden, um auf eine Protokollversandungs- oder Standby-Datenbankkonfiguration zu verweisen. Oracle DataGuard ist ein umfassendes Framework für das Management der Datenbankreplikation, es handelt sich jedoch nicht um eine Replizierungstechnologie. Der Hauptvorteil einer kompletten DataGuard-Umgebung bei einer Migration ist das transparente Umschalten von einer Datenbank zur anderen. DataGuard ermöglicht außerdem ein transparentes Switchover zurück zur Originaldatenbank, falls ein Problem erkannt wird, beispielsweise ein Problem mit der Performance oder der Netzwerkkonnektivität in der neuen Umgebung. Eine vollständig konfigurierte DataGuard-Umgebung erfordert nicht nur die Konfiguration der Datenbankschicht, sondern auch der Applikationen, damit Applikationen eine Änderung am primären Datenbankstandort erkennen können. Im Allgemeinen ist es nicht notwendig, eine Migration mit DataGuard durchzuführen, aber einige Kunden haben intern umfangreiche DataGuard-Kenntnisse und verlassen sich bei Migrationsaufgaben bereits auf diese.

Neuarchitektur

Wie bereits erläutert, erfordert die Nutzung der erweiterten Funktionen von Storage Arrays manchmal eine Änderung des Datenbank-Layouts. Darüber hinaus verändert eine Änderung des Storage-Protokolls, wie etwa das Wechsel von ASM zu einem NFS Filesystem, zwangsläufig das Filesystem-Layout.

Einer der Hauptvorteile von Protokollversandmethoden, einschließlich DataGuard, besteht darin, dass das Replizierungsziel nicht mit der Quelle übereinstimmen muss. Bei der Migration von ASM zu einem normalen Dateisystem oder umgekehrt gibt es keine Probleme mit der Verwendung eines Protokollversandansatzes. Das genaue Layout der Datendateien kann am Ziel geändert werden, um die Verwendung der Pluggable Database (PDB)-Technologie zu optimieren oder QoS-Kontrollen für bestimmte Dateien selektiv festzulegen. Mit anderen Worten: Ein Migrationsprozess auf der Basis des Protokollversand ermöglicht Ihnen eine einfache und sichere Optimierung des Datenbank-Storage-Layouts.

Server-Ressourcen

Eine Einschränkung für die Migration auf Datenbankebene besteht in der Notwendigkeit eines zweiten Servers. Dieser zweite Server kann auf zwei Arten verwendet werden:

1. Sie können den zweiten Server als permanentes neues Zuhause für die Datenbank verwenden.
2. Sie können den zweiten Server als temporären Staging-Server verwenden. Nachdem die Datenmigration zum neuen Storage-Array abgeschlossen und getestet wurde, werden die LUN- oder NFS-Dateisysteme vom Staging-Server getrennt und mit dem ursprünglichen Server verbunden.

Die erste Option ist die einfachste, aber in sehr großen Umgebungen, die sehr leistungsstarke Server erfordern, ist die Verwendung möglicherweise nicht möglich. Die zweite Option erfordert zusätzliche Arbeit, um die Dateisysteme wieder an den ursprünglichen Speicherort zu verschieben. Es kann sich um eine einfache Operation handeln, bei der NFS als Storage-Protokoll verwendet wird, da die File-Systeme vom Staging-Server abgehängt und dann wieder auf dem ursprünglichen Server gemountet werden können.

Blockbasierte Dateisysteme erfordern eine zusätzliche Arbeitsleistung für die Aktualisierung von FC-Zoning oder iSCSI-Initiatoren. Bei den meisten logischen Volume-Managern (einschließlich ASM) werden die LUNs automatisch erkannt und online geschaltet, nachdem sie auf dem ursprünglichen Server verfügbar gemacht

wurden. Einige Dateisystem- und LVM-Implementierungen erfordern jedoch möglicherweise mehr Arbeit für den Export und Import der Daten. Die genaue Vorgehensweise kann variieren, es ist jedoch im Allgemeinen einfach, ein einfaches, wiederholbares Verfahren einzurichten, um die Migration abzuschließen und die Daten auf dem ursprünglichen Server wiederherzustellen.

Es ist zwar möglich, einen Protokollversand einzurichten und eine Datenbank in einer einzigen Server-Umgebung zu replizieren, aber die neue Instanz muss eine andere Prozess-SID haben, um die Protokolle wiederzugeben. Es ist möglich, die Datenbank vorübergehend unter einem anderen Satz von Prozess-IDs mit einer anderen SID zu erstellen und später zu ändern. Dies kann jedoch zu vielen komplizierten Management-Aktivitäten und einem Risiko von Benutzerfehlern führen.

Migration auf Host-Ebene

Bei der Migration von Daten auf Hostebene müssen das Host-Betriebssystem und die zugehörigen Dienstprogramme zum Abschluss der Migration verwendet werden. Dieser Prozess umfasst alle Utilities zum Kopieren von Daten, darunter Oracle RMAN und Oracle ASM.

Kopieren von Daten

Der Wert einer einfachen Kopieroperation sollte nicht unterschätzt werden. Moderne Netzwerkinfrastrukturen können Daten in Gigabytes pro Sekunde verschieben und Dateikopiervorgänge basieren auf effizienten sequenziellen Lese- und Schreib-I/O. Im Vergleich zum Protokollversand lassen sich mehr Unterbrechungen durch Host-Kopien vermeiden, doch bei einer Migration handelt es sich nicht nur um die Datenverschiebung. Sie umfasst im Allgemeinen Änderungen am Netzwerk, den Neustartzeit der Datenbank und Tests nach der Migration.

Die tatsächlich zum Kopieren der Daten benötigte Zeit ist möglicherweise nicht signifikant. Darüber hinaus behält ein Kopiervorgang einen garantierten Back-out-Pfad bei, da die Originaldaten unverändert bleiben. Sollten während des Migrationsprozesses Probleme auftreten, können die ursprünglichen Dateisysteme mit den Originaldaten wieder aktiviert werden.

Ändern Der Plattform

Replatforming bezieht sich auf eine Änderung des CPU-Typs. Wenn eine Datenbank von einer herkömmlichen Solaris-, AIX- oder HP-UX-Plattform zu x86 Linux migriert wird, müssen die Daten aufgrund von Änderungen in der CPU-Architektur neu formatiert werden. SPARC, IA64 und POWER CPUs werden als Big-Endian-Prozessoren bezeichnet, während die x86- und x86_64-Architekturen als Little-Endian bezeichnet werden. Daher werden einige Daten in Oracle-Datendateien je nach verwendetem Prozessor unterschiedlich sortiert.

In der Vergangenheit haben Kunden Daten mithilfe von DataPump plattformübergreifend repliziert. DataPump ist ein Dienstprogramm, das einen speziellen Typ des logischen Datenexports erzeugt, der schneller in die Zieldatenbank importiert werden kann. Da es eine logische Kopie der Daten erstellt, lässt DataPump die Abhängigkeiten der Prozessorabhängigkeit hinter sich. DataPump wird von einigen Kunden weiterhin für das Replatforming verwendet, aber mit Oracle 11g ist eine schnellere Option verfügbar: Plattformübergreifende transportable Tablespaces. Mit diesem Vorschub kann ein Tablespace in ein anderes endian-Format konvertiert werden. Dies ist eine physische Transformation, die eine bessere Leistung bietet als ein DataPump-Export, der physische Bytes in logische Daten konvertieren und dann zurück in physische Bytes konvertieren muss.

Eine vollständige Diskussion über DataPump und transportable Tablespaces geht über den Umfang der NetApp-Dokumentation hinaus. NetApp hat jedoch einige Empfehlungen, die auf unseren Erfahrungen basieren, die Kunden bei der Migration zu einem neuen Storage Array-Protokoll mit einer neuen CPU-Architektur unterstützt haben:

- Wenn DataPump verwendet wird, sollte die für den Abschluss der Migration erforderliche Zeit in einer

Testumgebung gemessen werden. Kunden sind manchmal überrascht, wie lange sie für die Durchführung der Migration benötigen. Diese unerwartete zusätzliche Ausfallzeit kann zu Unterbrechungen führen.

- Viele Kunden glauben irrtümlicherweise, dass plattformübergreifende transportable Tablespaces keine Datenkonvertierung erfordern. Wenn eine CPU mit einem anderen Endian verwendet wird, wird ein RMAN verwendet `convert`. Der Betrieb muss zuvor an den Datendateien durchgeführt werden. Dies ist kein sofortiger Vorgang. In einigen Fällen kann der Konvertierungsprozess beschleunigt werden, indem mehrere Threads auf verschiedenen Dateien arbeiten, aber der Konvertierungsprozess kann nicht vermieden werden.

Migration über Manager eines logischen Volumes

LVMs nehmen eine Gruppe von einer oder mehreren LUNs und zerteilen sie in kleine Einheiten, die im Allgemeinen als Extents bezeichnet werden. Der Pool mit Erweiterungen wird dann als Quelle verwendet, um logische Volumes zu erstellen, die im Wesentlichen virtualisiert sind. Diese Virtualisierungsebene bietet auf verschiedene Weise einen Mehrwert:

- Logische Volumes können Extents verwenden, die von mehreren LUNs stammen. Wenn ein Filesystem auf einem logischen Volume erstellt wird, können alle Performance-Funktionen aller LUNs genutzt werden. Zudem wird die gleichmäßige Auslastung aller LUNs in der Volume-Gruppe gefördert, wodurch eine besser planbare Performance erzielt wird.
- Die Größe logischer Volumes kann durch Hinzufügen und in einigen Fällen durch Entfernen von Extents geändert werden. Die Größe eines Filesystems auf einem logischen Volume ist im Allgemeinen unterbrechungsfrei.
- Logische Volumes können unterbrechungsfrei migriert werden, indem die zugrunde liegenden Extents verschoben werden.

Migration mit einer LVM funktioniert auf zwei Arten: Ein Extent verschieben oder ein Extent spiegeln/demirrieren. Bei der LVM-Migration werden effiziente sequenzielle I/O große Blöcke eingesetzt, und es entstehen nur selten Performance-Probleme. Wenn dies zu einem Problem wird, gibt es in der Regel Optionen zur Drosselung der I/O-Rate. Dadurch erhöht sich die für den Abschluss der Migration erforderliche Zeit und gleichzeitig verringert sich die I/O-Last für Host- und Speichersysteme.

Spiegel und Demirror

Einige Volume-Manager, wie AIX LVM, erlauben dem Benutzer, die Anzahl der Kopien für jedes Extent festzulegen und zu steuern, welche Geräte die einzelnen Kopien hosten. Zur Migration wird ein vorhandenes logisches Volume erstellt, die zugrunde liegenden Extents zu den neuen Volumes gespiegelt, auf eine Synchronisierung der Kopien gewartet und anschließend die alte Kopie verworfen. Wenn ein Back- Out-Pfad gewünscht wird, kann vor dem Zeitpunkt, an dem die Spiegelungskopie abgelegt wird, ein Snapshot der Originaldaten erstellt werden. Alternativ kann der Server kurz heruntergefahren werden, um die ursprünglichen LUNs zu maskieren, bevor die enthaltenen Spiegelkopien erzwungen gelöscht werden. Dabei wird eine wiederherstellbare Kopie der Daten am ursprünglichen Speicherort aufbewahrt.

Extent-Migration

Fast alle Volume-Manager erlauben die Migration von Extents, und manchmal gibt es mehrere Optionen. Beispielsweise ermöglichen einige Volume Manager einem Administrator, die einzelnen Extents für ein bestimmtes logisches Volume von altem zu neuem Storage zu verschieben. Volume-Manager wie Linux LVM2 bieten die `pvmove` Befehl, der alle Extents auf dem angegebenen LUN-Gerät auf eine neue LUN verlagert. Nach der Evakuierung der alten LUN kann sie entfernt werden.



Das primäre Risiko für den Betrieb ist das Entfernen alter, nicht genutzter LUNs aus der Konfiguration. Beim Ändern des FC-Zoning und beim Entfernen veralteter LUN-Geräte ist besonders darauf zu achten.

Oracle Automatic Storage Management

Oracle ASM ist ein kombinierter logischer Volume-Manager und ein Dateisystem. Oracle ASM erstellt eine Sammlung von LUNs, unterteilt sie in kleine Zuweisungseinheiten und präsentiert sie als einzelnes Volume, das als ASM-Festplattengruppe bezeichnet wird. ASM bietet auch die Möglichkeit, die Laufwerksgruppe durch Festlegen des Redundanzniveaus zu spiegeln. Ein Volume kann nicht gespiegelt (externe Redundanz), gespiegelt (normale Redundanz) oder dreifach gespiegelt (hohe Redundanz) werden. Bei der Konfiguration der Redundanzstufe ist darauf zu achten, dass sie nach der Erstellung nicht mehr geändert werden kann.

ASM bietet auch Dateisystemfunktionen. Obwohl das Dateisystem nicht direkt vom Host aus sichtbar ist, kann die Oracle-Datenbank Dateien und Verzeichnisse auf einer ASM-Datenträgergruppe erstellen, verschieben und löschen. Außerdem kann die Struktur mit dem Dienstprogramm `asmcmd` navigiert werden.

Wie bei anderen LVM-Implementierungen optimiert Oracle ASM die I/O-Performance durch Striping und Lastausgleich der I/O-Vorgänge jeder Datei über alle verfügbaren LUNs. Zweitens können die zugrunde liegenden Extents verschoben werden, um sowohl die Größenänderung der ASM-Datenträgergruppe als auch die Migration zu ermöglichen. Oracle ASM automatisiert den Prozess durch den Rebalancing-Vorgang. Neue LUNs werden einer ASM-Festplattengruppe hinzugefügt und alte LUNs werden verworfen. Dies führt zu einer Extent-Verschiebung und einem nachfolgenden Drop der evakuierten LUN aus der Festplattengruppe. Dieser Prozess ist eine der bewährtesten Migrationsmethoden, und die Zuverlässigkeit von ASM bei der Bereitstellung einer transparenten Migration ist möglicherweise das wichtigste Merkmal.



Da die Spiegelungsebene von Oracle ASM fest festgelegt ist, kann sie nicht mit der Mirror- und Demirror-Methode der Migration verwendet werden.

Migration auf Storage-Ebene

Bei der Migration auf Storage-Ebene wird die Migration sowohl unter der Applikations- als auch unter der Betriebssystemebene durchgeführt. In der Vergangenheit bedeutete dies manchmal, spezialisierte Geräte zu verwenden, auf denen LUNs auf Netzwerkebene kopiert werden konnten. Diese Funktionen finden sich jedoch jetzt nativ in ONTAP.

SnapMirror

Mit der Datenreplizierungssoftware NetApp SnapMirror erfolgt die Migration von Datenbanken zwischen NetApp Systemen nahezu universell. Der Prozess beinhaltet die Einrichtung einer Spiegelbeziehung für die zu migrierenden Volumes, um sie zu synchronisieren und dann auf das Umstellungsfenster zu warten. Wenn sie eintrifft, wird die Quelldatenbank heruntergefahren, eine letzte Aktualisierung der Spiegelung durchgeführt und die Spiegelung wird unterbrochen. Die Replikatvolumes können dann verwendet werden, indem entweder ein enthaltenes NFS-Dateisystem-Verzeichnis gemountet oder die enthaltenen LUNs ermittelt und die Datenbank gestartet wird.

Das Verschieben von Volumes innerhalb eines einzigen ONTAP Clusters gilt nicht als Migration, sondern als Routine `volume move` Betrieb. SnapMirror wird als Datenreplizierungs-Engine im Cluster eingesetzt. Dieser Prozess ist vollständig automatisiert. Es gibt keine weiteren Migrationsschritte, die durchgeführt werden müssen, wenn Attribute des Volume, wie z. B. LUN-Zuordnung oder NFS-Exportberechtigungen, mit dem Volume selbst verschoben werden. Die Standortverlagerung hat keine Unterbrechung des Host-Betriebs. In manchen Fällen muss der Netzwerkzugriff aktualisiert werden, um sicherzustellen, dass auf die neu verlagerten Daten so effizient wie möglich zugegriffen wird. Diese Aufgaben sind aber auch unterbrechungsfrei.

Import fremder LUNs (FLI)

FLI ist eine Funktion, mit der ein Data ONTAP-System mit 8.3 oder höher eine vorhandene LUN von einem anderen Storage-Array migrieren kann. Das Verfahren ist einfach: Das ONTAP-System ist auf das bestehende Speicher-Array abgegrenzt, als ob es sich um einen anderen SAN-Host handelt. Data ONTAP übernimmt dann die Kontrolle über die gewünschten Legacy-LUNs und migriert die zugrunde liegenden Daten. Außerdem kommen bei der Migration von Daten im Importprozess die Effizienzeinstellungen des neuen Volume zum Einsatz, sodass Daten während des Migrationsprozesses inline komprimiert und dedupliziert werden können.

Die erste Implementierung von FLI in Data ONTAP 8.3 erlaubte nur Offline-Migration. Dies war ein extrem schneller Transfer, aber trotzdem bedeuteten die LUN-Daten, dass sie erst nach Abschluss der Migration verfügbar waren. Die Online-Migration wurde mit Data ONTAP 8.3 eingeführt. Diese Migration minimiert Unterbrechungen, da ONTAP während der Übertragung LUN-Daten bereitstellen kann. Während die Host-Zone neu aufgeteilt wird, um die LUNs über ONTAP zu verwenden, kommt es zu einer kurzen Unterbrechung. Sobald diese Änderungen jedoch vorgenommen werden, sind die Daten wieder verfügbar und bleiben während des gesamten Migrationsprozesses zugänglich.

Lese-I/O wird über ONTAP als Proxy übertragen, bis der Kopiervorgang abgeschlossen ist, während Schreib-I/O synchron sowohl auf die fremde als auch auf die ONTAP-LUN geschrieben wird. Die beiden LUN-Kopien werden auf diese Weise synchron gehalten, bis der Administrator eine vollständige Umstellung ausführt, die die fremde LUN freigibt und Schreibvorgänge nicht mehr repliziert.

FLI ist für den Einsatz mit FC konzipiert. Wenn jedoch ein Wechsel zu iSCSI gewünscht wird, kann die migrierte LUN nach Abschluss der Migration problemlos als iSCSI-LUN neu zugeordnet werden.

Zu den Merkmalen von FLI gehört die automatische Ausrichtungserkennung und -Einstellung. In diesem Kontext bezieht sich der Begriff „Alignment“ auf eine Partition auf einem LUN-Gerät. Für eine optimale Performance muss der I/O mit 4-KB-Blöcken abgestimmt werden. Wenn eine Partition auf einem Offset platziert wird, der kein Vielfaches von 4K ist, leidet die Performance.

Es gibt einen zweiten Aspekt der Ausrichtung, der nicht korrigiert werden kann, indem ein Partitionsoffset angepasst wird: Die Blockgröße des Dateisystems. Ein ZFS-Dateisystem beispielsweise hat in der Regel eine interne Blockgröße von 512 Byte. Andere Kunden, die AIX verwenden, haben gelegentlich jfs2-Dateisysteme mit einer 512- oder 1, 024-Byte-Blockgröße erstellt. Auch wenn das Filesystem an eine 4-KB-Grenze ausgerichtet ist, bleiben die in diesem Filesystem erstellten Dateien jedoch nicht und die Performance leidet.

FLI sollte unter diesen Umständen nicht verwendet werden. Obwohl nach der Migration auf die Daten zugegriffen werden kann, ergeben sich daraus Filesysteme mit erheblichen Performance-Einschränkungen. Grundsätzlich sollte jedes Filesystem, das einen zufälligen Überschreibvorgang auf ONTAP unterstützt, eine 4-KB-Blockgröße verwenden. Dies gilt insbesondere für Workloads wie Datenbankdateien und VDI-Implementierungen. Die Blockgröße kann mit den entsprechenden Host-Betriebssystembefehlen identifiziert werden.

Auf AIX kann beispielsweise die Blockgröße mit angezeigt werden `lsfs -q`. Mit Linux `xfs_info` Und `tune2fs` Kann für verwendet werden `xfs` Und `ext3/ext4`. Mit `zfs`, Der Befehl lautet `zdb -C`.

Der Parameter, der die Blockgröße steuert, ist `ashift` Und im Allgemeinen ist der Standardwert 9, was 2^9 oder 512 Byte bedeutet. Für eine optimale Leistung, die `ashift` Wert muss 12 ($2^{12}=4K$) sein. Dieser Wert wird zum Zeitpunkt der Erstellung des `zpool` gesetzt und kann nicht geändert werden, was bedeutet, dass Data `zpool`s mit einem `ashift` Andere als 12 sollten durch Kopieren der Daten in einen neu erstellten `zpool` migriert werden.

Oracle ASM hat keine grundlegende Blockgröße. Die einzige Voraussetzung ist, dass die Partition, auf der die ASM-Festplatte erstellt wird, ordnungsgemäß ausgerichtet sein muss.

7-Mode Transition Tool

Bei dem 7-Mode Transition Tool (7MTT) handelt es sich um ein Automatisierungstool zur Migration großer 7-Mode Konfigurationen zu ONTAP. Die meisten Datenbankkunden finden andere Methoden einfacher, zum Teil, da sie in der Regel ihre Umgebungen einer Datenbank nach Datenbank migrieren, anstatt den gesamten Storage-Platzbedarf zu verschieben. Zudem sind Datenbanken häufig nur ein Teil einer größeren Storage-Umgebung. Daher werden Datenbanken oft einzeln migriert und die restliche Umgebung kann mit 7MTT verschoben werden.

Es gibt eine kleine aber beträchtliche Anzahl von Kunden, die Storage-Systeme haben, die komplizierten Datenbankumgebungen gewidmet sind. Diese Umgebungen können viele Volumes, Snapshots und zahlreiche Konfigurationsdetails wie Exportberechtigungen, LUN-Initiatorgruppen, Benutzerberechtigungen und die Konfiguration des Lightweight Directory Access Protocol enthalten. In diesen Fällen können die Automatisierungsfunktionen von 7MTT die Migration vereinfachen.

7MTT kann in einem der beiden Modi ausgeführt werden:

- **Copy-Based Transition (CBT).** 7MTT mit CBT richtet SnapMirror Volumes aus einem bestehenden 7-Mode System in der neuen Umgebung ein. Nachdem die Daten synchronisiert sind, orchestriert 7MTT den Umstellungsprozess.
- **Copy-Free Transition (CFT).** 7MTT mit CFT basiert auf der in-Place Konvertierung vorhandener 7-Mode Platten-Shelfs. Es werden keine Daten kopiert und die vorhandenen Festplatten-Shelfs können wieder verwendet werden. Die vorhandene Konfiguration für Datensicherung und Storage-Effizienz bleibt erhalten.

Der primäre Unterschied zwischen diesen beiden Optionen ist der Copy-Free Transition. Er ist ein „Big-Bang“-Ansatz, bei dem alle mit dem ursprünglichen 7-Mode HA-Paar verbundenen Platten-Shelfs in die neue Umgebung verschoben werden müssen. Eine Untergruppe von Shelfs lässt sich nicht verschieben. Durch den Copy-basierten Ansatz können ausgewählte Volumes verschoben werden. Es besteht auch die Möglichkeit, dass ein längeres Umstellungsfenster mit Copy-Free Transition möglich ist, da für die Neuerstellung von Festplatten-Shelfs und die Konvertierung von Metadaten eine Verbindung erforderlich ist. Je nach Praxiserfahrung empfiehlt NetApp, für die Verlagerung und Neuverkabelung von Festplatten-Shelfs eine Stunde und für die Metadatenkonvertierung zwischen 15 Minuten und 2 Stunden zu verwenden.

Migration von Datendateien

Einzelne Oracle Datendateien können mit einem einzigen Befehl verschoben werden.

Mit dem folgenden Befehl wird beispielsweise die Datendatei IOPST.dbf aus dem Dateisystem verschoben /oradata2 Zu Dateisystem /oradata3.

```
SQL> alter database move datafile '/oradata2/NTAP/IOPS002.dbf' to  
'/oradata3/NTAP/IOPS002.dbf';  
Database altered.
```

Das Verschieben einer Datendatei mit dieser Methode kann langsam sein, sollte jedoch normalerweise nicht genügend I/O produzieren, um die täglichen Datenbank-Workloads zu beeinträchtigen. Im Gegensatz dazu kann die Migration über die ASM-Ausbalancierung viel schneller ablaufen, doch dies geschieht auf Kosten der Verlangsamung der gesamten Datenbank, während die Daten verschoben werden.

Die zum Verschieben von Datendateien erforderliche Zeit kann einfach gemessen werden, indem eine Test-Datendatei erstellt und dann verschoben wird. Die verstrichene Zeit für den Vorgang wird in den Sitzungsdaten mit den folgenden Kosten aufgezeichnet:

```

SQL> set linesize 300;
SQL> select elapsed_seconds||':'||message from v$session_longops;
ELAPSED_SECONDS||':'||MESSAGE
-----
-----
351:Online data file move: data file 8: 22548578304 out of 22548578304
bytes done
SQL> select bytes / 1024 / 1024 /1024 as GB from dba_data_files where
FILE_ID = 8;
          GB
-----
          21

```

In diesem Beispiel handelte es sich bei der verschobenen Datei um die Datendatei 8, deren Größe 21 GB betrug und die Migration ca. 6 Minuten in Anspruch nahm. Der erforderliche Zeitaufwand hängt natürlich von den Funktionen des Storage-Systems, des Storage-Netzwerks und der gesamten Datenbankaktivität zum Zeitpunkt der Migration ab.

Protokollversand

Ziel bei einer Migration mit Protokollversand ist, eine Kopie der ursprünglichen Datendateien an einem neuen Standort zu erstellen und anschließend eine Methode für den Versand von Änderungen in die neue Umgebung zu definieren.

Nach der Einrichtung können Protokollversand und -Wiedergabe automatisiert werden, um die Replikatdatenbank weitgehend mit der Quelle synchron zu halten. So kann beispielsweise ein Cron-Job so geplant werden, dass (a) die letzten Protokolle an den neuen Speicherort kopiert und (b) alle 15 Minuten erneut wiedergegeben werden. Dadurch sind zum Zeitpunkt der Umstellung nur minimale Unterbrechungen möglich, da maximal 15 Minuten Archiv-Logs wieder eingespielt werden müssen.

Das unten abgebildete Verfahren ist außerdem im Wesentlichen eine Datenbankklonoperation. Die gezeigte Logik ähnelt der Engine in NetApp SnapManager für Oracle (SMO) und dem NetApp SnapCenter Oracle Plugin. Einige Kunden haben das in Skripten oder WFA Workflows angezeigte Verfahren für individuelle Klonvorgänge verwendet. Dieses Verfahren ist zwar mehr manuell als SMO oder SnapCenter, es wird jedoch immer noch ohne Skripte erstellt und die Datenmanagement-APIs in ONTAP vereinfachen den Prozess weiter.

Protokollversand – Dateisystem an Dateisystem

Dieses Beispiel zeigt die Migration einer Datenbank namens WAFFLE von einem gewöhnlichen Dateisystem zu einem anderen gewöhnlichen Dateisystem auf einem anderen Server. Es veranschaulicht auch die Verwendung von SnapMirror zum Erstellen einer schnellen Kopie der Datendateien, aber dies ist kein integraler Bestandteil des gesamten Verfahrens.

Erstellen Sie eine Datenbanksicherung

Der erste Schritt besteht darin, ein Datenbank-Backup zu erstellen. Insbesondere erfordert dieses Verfahren eine Reihe von Datendateien, die für die Wiedergabe des Archivprotokolls verwendet werden können.

Umgebung

In diesem Beispiel befindet sich die Quelldatenbank auf einem ONTAP-System. Die einfachste Methode, um ein Backup einer Datenbank zu erstellen, ist die Verwendung eines Snapshots. Die Datenbank wird für einige Sekunden in den Hot Backup-Modus versetzt, während ein `snapshot create` Der Vorgang wird auf dem Volume ausgeführt, auf dem die Datendateien gehostet werden.

```
SQL> alter database begin backup;  
Database altered.
```

```
Cluster01::*> snapshot create -vserver vserver1 -volume jfsc1_oradata  
hotbackup  
Cluster01::*>
```

```
SQL> alter database end backup;  
Database altered.
```

Das Ergebnis ist ein Snapshot auf der Festplatte, der aufgerufen wird `hotbackup` Die ein Image der Datendateien enthält, während sie sich im Hot Backup-Modus befinden. Wenn die Daten in diesem Snapshot mit den entsprechenden Archivprotokollen konsistent sind, können sie als Grundlage für eine Wiederherstellung oder einen Klon verwendet werden. In diesem Fall wird sie auf den neuen Server repliziert.

Wiederherstellung in neuer Umgebung

Das Backup muss nun in der neuen Umgebung wiederhergestellt werden. Dies kann auf verschiedene Arten erfolgen, z. B. Oracle RMAN, Wiederherstellung über eine Backup-Applikation wie NetBackup oder ein einfacher Kopiervorgang von Datendateien, die im Hot-Backup-Modus platziert wurden.

In diesem Beispiel wird SnapMirror verwendet, um das Snapshot Hot-Backup an einen neuen Speicherort zu replizieren.

1. Erstellen Sie ein neues Volume für den Empfang der Snapshot-Daten. Initialisieren Sie die Spiegelung von `jfsc1_oradata` Bis `vol_oradata`.

```
Cluster01::*> volume create -vserver vserver1 -volume vol_oradata  
-aggregate data_01 -size 20g -state online -type DP -snapshot-policy  
none -policy jfsc3  
[Job 833] Job succeeded: Successful
```

```
Cluster01::*> snapmirror initialize -source-path vserver1:jfsc1_oradata
-destination-path vserver1:vol_oradata
Operation is queued: snapmirror initialize of destination
"vserver1:vol_oradata".
Cluster01::*> volume mount -vserver vserver1 -volume vol_oradata
-junction-path /vol_oradata
Cluster01::*>
```

2. Nachdem der Status von SnapMirror festgelegt wurde und Sie angeben, dass die Synchronisierung abgeschlossen ist, aktualisieren Sie die Spiegelung speziell auf der Grundlage des gewünschten Snapshots.

```
Cluster01::*> snapmirror show -destination-path vserver1:vol_oradata
-fields state
source-path          destination-path      state
-----
vserver1:jfsc1_oradata vserver1:vol_oradata SnapMirrored
```

```
Cluster01::*> snapmirror update -destination-path vserver1:vol_oradata
-source-snapshot hotbackup
Operation is queued: snapmirror update of destination
"vserver1:vol_oradata".
```

3. Die erfolgreiche Synchronisierung kann durch Anzeigen des überprüften newest-snapshot Feld auf der Spiegelungslautstärke.

```
Cluster01::*> snapmirror show -destination-path vserver1:vol_oradata
-fields newest-snapshot
source-path          destination-path      newest-snapshot
-----
vserver1:jfsc1_oradata vserver1:vol_oradata hotbackup
```

4. Der Spiegel kann dann gebrochen werden.

```
Cluster01:::> snapmirror break -destination-path vserver1:vol_oradata
Operation succeeded: snapmirror break for destination
"vserver1:vol_oradata".
Cluster01:::>
```

5. Mounten Sie das neue Dateisystem. bei blockbasierten Dateisystemen variieren die genauen Verfahren je nach der verwendeten LVM. FC-Zoning oder iSCSI-Verbindungen müssen konfiguriert werden. Nachdem die Verbindung zu den LUNs hergestellt wurde, Befehle wie Linux `pvs`scan Möglicherweise muss ermittelt

werden, welche Volume-Gruppen oder LUNs ordnungsgemäß konfiguriert werden müssen, damit sie von ASM erkannt werden können.

In diesem Beispiel wird ein einfaches NFS-Dateisystem verwendet. Dieses Dateisystem kann direkt gemountet werden.

```
fas8060-nfs1:/vol_oradata      19922944    1639360    18283584    9%
/oradata
fas8060-nfs1:/vol_logs        9961472     128        9961344     1%
/logs
```

Erstellen Sie eine Vorlage für die Erstellung von Steuerdateien

Als nächstes müssen Sie eine controlfile-Vorlage erstellen. Der `backup controlfile to trace` Befehl erstellt Textbefehle, um eine Steuerdatei neu zu erstellen. Diese Funktion kann unter bestimmten Umständen hilfreich sein, um eine Datenbank aus einem Backup wiederherzustellen. Sie wird häufig bei Skripten verwendet, die Aufgaben wie das Klonen von Datenbanken ausführen.

1. Die Ausgabe des folgenden Befehls wird verwendet, um die Steuerdateien für die migrierte Datenbank neu zu erstellen.

```
SQL> alter database backup controlfile to trace as '/tmp/waffle.ctrl';
Database altered.
```

2. Nachdem die Steuerdateien erstellt wurden, kopieren Sie die Datei auf den neuen Server.

```
[oracle@jpsc3 tmp]$ scp oracle@jpsc1:/tmp/waffle.ctrl /tmp/
oracle@jpsc1's password:
waffle.ctrl                                100% 5199
5.1KB/s   00:00
```

Parameterdatei sichern

In der neuen Umgebung ist auch eine Parameterdatei erforderlich. Die einfachste Methode ist, aus dem aktuellen spfile oder pfile ein pfile zu erstellen. In diesem Beispiel verwendet die Quelldatenbank ein spfile.

```
SQL> create pfile='/tmp/waffle.tmp.pfile' from spfile;
File created.
```

Oratab-Eintrag erstellen

Die Erstellung eines Oratab-Eintrags ist für das ordnungsgemäße Funktionieren von Dienstprogrammen wie oraenv erforderlich. Führen Sie den folgenden Schritt aus, um einen Oratab-Eintrag zu erstellen.

```
WAFFLE:/orabin/product/12.1.0/dbhome_1:N
```

Verzeichnisstruktur vorbereiten

Wenn die benötigten Verzeichnisse noch nicht vorhanden waren, müssen Sie sie erstellen, oder der Datenbankstartvorgang schlägt fehl. Um die Verzeichnisstruktur vorzubereiten, müssen Sie die folgenden Mindestanforderungen erfüllen.

```
[oracle@jpsc3 ~]$ . oraenv
ORACLE_SID = [oracle] ? WAFFLE
The Oracle base has been set to /orabin
[oracle@jpsc3 ~]$ cd $ORACLE_BASE
[oracle@jpsc3 orabin]$ cd admin
[oracle@jpsc3 admin]$ mkdir WAFFLE
[oracle@jpsc3 admin]$ cd WAFFLE
[oracle@jpsc3 WAFFLE]$ mkdir adump dpdump pfile scripts xdb_wallet
```

Aktualisierung der Parameterdatei

1. Um die Parameterdatei auf den neuen Server zu kopieren, führen Sie die folgenden Befehle aus. Der Standardspeicherort ist der `$ORACLE_HOME/dbs` Verzeichnis. In diesem Fall kann die `pfile` überall platziert werden. Sie wird nur als Zwischenschritt im Migrationsprozess genutzt.

```
[oracle@jpsc3 admin]$ scp oracle@jpsc1:/tmp/waffle.tmp.pfile
$ORACLE_HOME/dbs/waffle.tmp.pfile
oracle@jpsc1's password:
waffle.pfile                                100%  916
0.9KB/s   00:00
```

1. Bearbeiten Sie die Datei nach Bedarf. Wenn sich beispielsweise der Speicherort des Archivprotokolls geändert hat, muss das `pfile` entsprechend dem neuen Speicherort geändert werden. In diesem Beispiel werden nur die Steuerdateien verschoben, zum Teil, um sie zwischen Protokoll- und Datendateisystemen zu verteilen.

```

[root@jfscl tmp]# cat waffle.pfile
WAFFLE.__data_transfer_cache_size=0
WAFFLE.__db_cache_size=507510784
WAFFLE.__java_pool_size=4194304
WAFFLE.__large_pool_size=20971520
WAFFLE.__oracle_base='/orabin'#ORACLE_BASE set from environment
WAFFLE.__pga_aggregate_target=268435456
WAFFLE.__sga_target=805306368
WAFFLE.__shared_io_pool_size=29360128
WAFFLE.__shared_pool_size=234881024
WAFFLE.__streams_pool_size=0
*.audit_file_dest='/orabin/admin/WAFFLE/adump'
*.audit_trail='db'
*.compatible='12.1.0.2.0'
*.control_files='/oradata//WAFFLE/control01.ctl','/oradata//WAFFLE/control02.ctl'
*.control_files='/oradata/WAFFLE/control01.ctl','/logs/WAFFLE/control02.ctl'
*.db_block_size=8192
*.db_domain=''
*.db_name='WAFFLE'
*.diagnostic_dest='/orabin'
*.dispatchers='(PROTOCOL=TCP) (SERVICE=WAFFLEXDB)'
*.log_archive_dest_1='LOCATION=/logs/WAFFLE/arch'
*.log_archive_format='%t_%s_%r.dbf'
*.open_cursors=300
*.pga_aggregate_target=256m
*.processes=300
*.remote_login_passwordfile='EXCLUSIVE'
*.sga_target=768m
*.undo_tablespace='UNDOTBS1'

```

2. Nachdem die Bearbeitungen abgeschlossen sind, erstellen Sie auf Basis dieses pfile ein spfile.

```

SQL> create spfile from pfile='waffle.tmp.pfile';
File created.

```

Erstellen Sie Steuerdateien neu

In einem vorherigen Schritt wird die Ausgabe von `analyze backup controlfile to trace` auf den neuen Server kopiert. Der spezifische Teil des erforderlichen Ausgangs ist der `controlfile recreation` Befehl. Diese Informationen finden Sie in der Datei unter dem markierten Abschnitt `Set #1. NORESETLOGS`. Es beginnt mit der Linie `create controlfile reuse database` Und sollte das Wort `noresetlogs` enthalten. Er endet mit dem Semikolon (;).

1. In diesem Beispiel liest die Datei wie folgt.

```
CREATE CONTROLFILE REUSE DATABASE "WAFFLE" NORESETLOGS ARCHIVELOG
  MAXLOGFILES 16
  MAXLOGMEMBERS 3
  MAXDATAFILES 100
  MAXINSTANCES 8
  MAXLOGHISTORY 292
LOGFILE
  GROUP 1 '/logs/WAFFLE/redo/redo01.log' SIZE 50M BLOCKSIZE 512,
  GROUP 2 '/logs/WAFFLE/redo/redo02.log' SIZE 50M BLOCKSIZE 512,
  GROUP 3 '/logs/WAFFLE/redo/redo03.log' SIZE 50M BLOCKSIZE 512
-- STANDBY LOGFILE
DATAFILE
  '/oradata/WAFFLE/system01.dbf',
  '/oradata/WAFFLE/sysaux01.dbf',
  '/oradata/WAFFLE/undotbs01.dbf',
  '/oradata/WAFFLE/users01.dbf'
CHARACTER SET WE8MSWIN1252
;
```

2. Bearbeiten Sie dieses Skript wie gewünscht, um den neuen Speicherort der verschiedenen Dateien anzuzeigen. Beispielsweise können bestimmte Datendateien, von denen bekannt ist, dass sie eine hohe I/O-Last unterstützen, auf ein Filesystem auf einer hochperformanten Storage-Ebene umgeleitet werden. In anderen Fällen könnten die Änderungen lediglich aus Administratorgründen vorgenommen werden, wie z. B. die Isolierung der Datendateien einer bestimmten PDB in dedizierten Volumes.
3. In diesem Beispiel ist der DATAFILE Stanza bleibt unverändert, aber die Redo-Logs werden an einen neuen Speicherort in verschoben /redo Statt Speicherplatz für Archivprotokolle freizugeben /logs.

```
CREATE CONTROLFILE REUSE DATABASE "WAFFLE" NORESETLOGS  ARCHIVELOG
  MAXLOGFILES 16
  MAXLOGMEMBERS 3
  MAXDATAFILES 100
  MAXINSTANCES 8
  MAXLOGHISTORY 292
LOGFILE
  GROUP 1 '/redo/redo01.log'  SIZE 50M BLOCKSIZE 512,
  GROUP 2 '/redo/redo02.log'  SIZE 50M BLOCKSIZE 512,
  GROUP 3 '/redo/redo03.log'  SIZE 50M BLOCKSIZE 512
-- STANDBY LOGFILE
DATAFILE
  '/oradata/WAFFLE/system01.dbf',
  '/oradata/WAFFLE/sysaux01.dbf',
  '/oradata/WAFFLE/undotbs01.dbf',
  '/oradata/WAFFLE/users01.dbf'
CHARACTER SET WE8MSWIN1252
;
```

```

SQL> startup nomount;
ORACLE instance started.
Total System Global Area  805306368 bytes
Fixed Size                  2929552 bytes
Variable Size              331353200 bytes
Database Buffers          465567744 bytes
Redo Buffers                5455872 bytes
SQL> CREATE CONTROLFILE REUSE DATABASE "WAFFLE" NORESETLOGS  ARCHIVELOG
 2     MAXLOGFILES 16
 3     MAXLOGMEMBERS 3
 4     MAXDATAFILES 100
 5     MAXINSTANCES 8
 6     MAXLOGHISTORY 292
 7 LOGFILE
 8   GROUP 1 '/redo/redo01.log'  SIZE 50M BLOCKSIZE 512,
 9   GROUP 2 '/redo/redo02.log'  SIZE 50M BLOCKSIZE 512,
10   GROUP 3 '/redo/redo03.log'  SIZE 50M BLOCKSIZE 512
11 -- STANDBY LOGFILE
12 DATAFILE
13   '/oradata/WAFFLE/system01.dbf',
14   '/oradata/WAFFLE/sysaux01.dbf',
15   '/oradata/WAFFLE/undotbs01.dbf',
16   '/oradata/WAFFLE/users01.dbf'
17 CHARACTER SET WE8MSWIN1252
18 ;
Control file created.
SQL>

```

Wenn Dateien falsch platziert oder Parameter falsch konfiguriert sind, werden Fehler generiert, die angeben, was repariert werden muss. Die Datenbank ist gemountet, aber noch nicht geöffnet und kann nicht geöffnet werden, da die verwendeten Datendateien noch als Hot Backup-Modus markiert sind. Um die Datenbankkonsistenz zu gewährleisten, müssen zunächst Archivprotokolle angewendet werden.

Erste Protokollreplizierung

Es ist mindestens ein Protokollantwort erforderlich, um die Datendateien konsistent zu gestalten. Es stehen zahlreiche Optionen zur Wiedergabe von Protokollen zur Verfügung. In einigen Fällen kann der ursprüngliche Speicherort des Archivprotokolls auf dem ursprünglichen Server über NFS freigegeben werden, und die Protokollantwort kann direkt erfolgen. In anderen Fällen müssen die Archivprotokolle kopiert werden.

Zum Beispiel, eine einfache `scp` Der Vorgang kann alle aktuellen Protokolle vom Quellserver auf den Migrationsserver kopieren:

```

[oracle@jfsc3 arch]$ scp jfsc1:/logs/WAFFLE/arch/* ./
oracle@jfsc1's password:
1_22_912662036.dbf          100%   47MB
47.0MB/s   00:01
1_23_912662036.dbf          100%   40MB
40.4MB/s   00:00
1_24_912662036.dbf          100%   45MB
45.4MB/s   00:00
1_25_912662036.dbf          100%   41MB
40.9MB/s   00:01
1_26_912662036.dbf          100%   39MB
39.4MB/s   00:00
1_27_912662036.dbf          100%   39MB
38.7MB/s   00:00
1_28_912662036.dbf          100%   40MB
40.1MB/s   00:01
1_29_912662036.dbf          100%   17MB
16.9MB/s   00:00
1_30_912662036.dbf          100%   636KB
636.0KB/s   00:00

```

Erste Protokollwiedergabe

Nachdem sich die Dateien im Archiv-Log-Speicherort befinden, können sie mit dem Befehl wiedergegeben werden `recover database until cancel` Gefolgt von der Antwort `AUTO` Um alle verfügbaren Protokolle automatisch wiederzugeben.

```

SQL> recover database until cancel;
ORA-00279: change 382713 generated at 05/24/2016 09:00:54 needed for
thread 1
ORA-00289: suggestion : /logs/WAFFLE/arch/1_23_912662036.dbf
ORA-00280: change 382713 for thread 1 is in sequence #23
Specify log: {<RET>=suggested | filename | AUTO | CANCEL}
AUTO
ORA-00279: change 405712 generated at 05/24/2016 15:01:05 needed for
thread 1
ORA-00289: suggestion : /logs/WAFFLE/arch/1_24_912662036.dbf
ORA-00280: change 405712 for thread 1 is in sequence #24
ORA-00278: log file '/logs/WAFFLE/arch/1_23_912662036.dbf' no longer
needed for
this recovery
...
ORA-00279: change 713874 generated at 05/26/2016 04:26:43 needed for
thread 1
ORA-00289: suggestion : /logs/WAFFLE/arch/1_31_912662036.dbf
ORA-00280: change 713874 for thread 1 is in sequence #31
ORA-00278: log file '/logs/WAFFLE/arch/1_30_912662036.dbf' no longer
needed for
this recovery
ORA-00308: cannot open archived log '/logs/WAFFLE/arch/1_31_912662036.dbf'
ORA-27037: unable to obtain file status
Linux-x86_64 Error: 2: No such file or directory
Additional information: 3

```

Die endgültige Antwort des Archivprotokolls meldet einen Fehler. Dies ist jedoch normal. Das Protokoll zeigt das an `sqlplus` Ich habe eine bestimmte Protokolldatei gesucht und sie nicht gefunden. Der Grund dafür ist höchstwahrscheinlich, dass die Protokolldatei noch nicht existiert.

Wenn die Quelldatenbank vor dem Kopieren von Archivprotokollen heruntergefahren werden kann, muss dieser Schritt nur einmal durchgeführt werden. Die Archivprotokolle werden kopiert und eingespielt. Anschließend kann der Prozess direkt zum Umstellungsprozess fortgesetzt werden, der die kritischen Wiederherstellungsprotokolle repliziert.

Inkrementelle Protokollreplikation und -Wiedergabe

In den meisten Fällen erfolgt die Migration nicht sofort. Es kann Tage oder sogar Wochen bis zum Abschluss des Migrationsprozesses dauern. Das bedeutet, dass die Protokolle kontinuierlich an die Replikatdatenbank gesendet und erneut eingespielt werden müssen. Bei Ankunft der Umstellung müssen daher nur minimale Daten übertragen und erneut eingespielt werden.

Dies kann auf viele Arten per Skript gesteuert werden, aber eine der beliebtesten Methoden ist die Verwendung von `rsync`, einem gemeinsamen Dateireplikationsdienstprogramm. Die sicherste Methode, dieses Dienstprogramm zu verwenden, ist es als Daemon zu konfigurieren. Beispiel: Der `rsyncd.conf` Die folgende Datei zeigt, wie eine Ressource mit dem Namen erstellt wird `waffle.arch` Der Zugriff erfolgt mit Oracle-Benutzeranmeldeinformationen und ist zugeordnet `/logs/WAFFLE/arch`. Am wichtigsten ist jedoch, dass die

Ressource schreibgeschützt ist, wodurch die Produktionsdaten gelesen, aber nicht verändert werden können.

```
[root@jfscl arch]# cat /etc/rsyncd.conf
[waffle.arch]
  uid=oracle
  gid=dba
  path=/logs/WAFFLE/arch
  read only = true
[root@jfscl arch]# rsync --daemon
```

Mit dem folgenden Befehl wird das Archivprotokollziel des neuen Servers mit der rsync-Ressource synchronisiert `waffle.arch` Auf dem ursprünglichen Server. Der `t` Argument in `rsync -ptg` Führt dazu, dass die Dateiliste anhand des Zeitstempels verglichen wird und nur neue Dateien kopiert werden. Dieser Prozess bietet eine inkrementelle Aktualisierung des neuen Servers. Dieser Befehl kann auch in cron so geplant werden, dass er regelmäßig ausgeführt wird.

```

[oracle@jfsc3 arch]$ rsync -potg --stats --progress jfsc1::waffle.arch/*
/logs/WAFFLE/arch/
1_31_912662036.dbf
   650240 100% 124.02MB/s   0:00:00 (xfer#1, to-check=8/18)
1_32_912662036.dbf
   4873728 100% 110.67MB/s   0:00:00 (xfer#2, to-check=7/18)
1_33_912662036.dbf
   4088832 100%  50.64MB/s   0:00:00 (xfer#3, to-check=6/18)
1_34_912662036.dbf
   8196096 100%  54.66MB/s   0:00:00 (xfer#4, to-check=5/18)
1_35_912662036.dbf
  19376128 100%  57.75MB/s   0:00:00 (xfer#5, to-check=4/18)
1_36_912662036.dbf
    71680 100% 201.15kB/s   0:00:00 (xfer#6, to-check=3/18)
1_37_912662036.dbf
  1144320 100%   3.06MB/s   0:00:00 (xfer#7, to-check=2/18)
1_38_912662036.dbf
  35757568 100%  63.74MB/s   0:00:00 (xfer#8, to-check=1/18)
1_39_912662036.dbf
   984576 100%   1.63MB/s   0:00:00 (xfer#9, to-check=0/18)
Number of files: 18
Number of files transferred: 9
Total file size: 399653376 bytes
Total transferred file size: 75143168 bytes
Literal data: 75143168 bytes
Matched data: 0 bytes
File list size: 474
File list generation time: 0.001 seconds
File list transfer time: 0.000 seconds
Total bytes sent: 204
Total bytes received: 75153219
sent 204 bytes  received 75153219 bytes  150306846.00 bytes/sec
total size is 399653376  speedup is 5.32

```

Nachdem die Protokolle empfangen wurden, müssen sie erneut abgespielt werden. Frühere Beispiele zeigen die Verwendung von sqlplus zum manuellen Ausführen `recover database until cancel` Ein Prozess, der leicht automatisiert werden kann. Das hier abgebildete Beispiel verwendet das in beschriebene Skript "[Protokolle in der Datenbank wiedergeben](#)". Die Skripte akzeptieren ein Argument, das die Datenbank angibt, die einen Wiedergabevorgang erfordert. Damit kann dasselbe Skript bei einer Migration mit mehreren Datenbanken verwendet werden.

```

[oracle@jfsc3 logs]$ ./replay.logs.pl WAFFLE
ORACLE_SID = [WAFFLE] ? The Oracle base remains unchanged with value
/orabin
SQL*Plus: Release 12.1.0.2.0 Production on Thu May 26 10:47:16 2016
Copyright (c) 1982, 2014, Oracle. All rights reserved.
Connected to:
Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 - 64bit
Production
With the Partitioning, OLAP, Advanced Analytics and Real Application
Testing options
SQL> ORA-00279: change 713874 generated at 05/26/2016 04:26:43 needed for
thread 1
ORA-00289: suggestion : /logs/WAFFLE/arch/1_31_912662036.dbf
ORA-00280: change 713874 for thread 1 is in sequence #31
Specify log: {<RET>=suggested | filename | AUTO | CANCEL}
ORA-00279: change 814256 generated at 05/26/2016 04:52:30 needed for
thread 1
ORA-00289: suggestion : /logs/WAFFLE/arch/1_32_912662036.dbf
ORA-00280: change 814256 for thread 1 is in sequence #32
ORA-00278: log file '/logs/WAFFLE/arch/1_31_912662036.dbf' no longer
needed for
this recovery
ORA-00279: change 814780 generated at 05/26/2016 04:53:04 needed for
thread 1
ORA-00289: suggestion : /logs/WAFFLE/arch/1_33_912662036.dbf
ORA-00280: change 814780 for thread 1 is in sequence #33
ORA-00278: log file '/logs/WAFFLE/arch/1_32_912662036.dbf' no longer
needed for
this recovery
...
ORA-00279: change 1120099 generated at 05/26/2016 09:59:21 needed for
thread 1
ORA-00289: suggestion : /logs/WAFFLE/arch/1_40_912662036.dbf
ORA-00280: change 1120099 for thread 1 is in sequence #40
ORA-00278: log file '/logs/WAFFLE/arch/1_39_912662036.dbf' no longer
needed for
this recovery
ORA-00308: cannot open archived log '/logs/WAFFLE/arch/1_40_912662036.dbf'
ORA-27037: unable to obtain file status
Linux-x86_64 Error: 2: No such file or directory
Additional information: 3
SQL> Disconnected from Oracle Database 12c Enterprise Edition Release
12.1.0.2.0 - 64bit Production
With the Partitioning, OLAP, Advanced Analytics and Real Application
Testing options

```

Umstellung

Wenn Sie bereit sind, in die neue Umgebung zu schneiden, müssen Sie eine abschließende Synchronisierung durchführen, die sowohl Archivprotokolle als auch Redo-Protokolle enthält. Wenn der ursprüngliche Speicherort des Wiederherstellungsprotokolls nicht bereits bekannt ist, kann er wie folgt identifiziert werden:

```
SQL> select member from v$logfile;
MEMBER
-----
-----
/logs/WAFFLE/redo/redo01.log
/logs/WAFFLE/redo/redo02.log
/logs/WAFFLE/redo/redo03.log
```

1. Fahren Sie die Quelldatenbank herunter.
2. Führen Sie eine abschließende Synchronisierung der Archivprotokolle auf dem neuen Server mit der gewünschten Methode durch.
3. Die Wiederherstellungsprotokolle der Quelle müssen auf den neuen Server kopiert werden. In diesem Beispiel wurden die Wiederherstellungsprotokolle in ein neues Verzeichnis unter verschoben `/redo`.

```
[oracle@jfsc3 logs]$ scp jfsc1:/logs/WAFFLE/redo/* /redo/
oracle@jfsc1's password:
redo01.log
100%  50MB  50.0MB/s   00:01
redo02.log
100%  50MB  50.0MB/s   00:00
redo03.log
100%  50MB  50.0MB/s   00:00
```

4. In dieser Phase enthält die neue Datenbankumgebung alle Dateien, die als Quelle erforderlich sind. Die Archivprotokolle müssen ein letztes Mal wiedergegeben werden.

```

SQL> recover database until cancel;
ORA-00279: change 1120099 generated at 05/26/2016 09:59:21 needed for
thread 1
ORA-00289: suggestion : /logs/WAFFLE/arch/1_40_912662036.dbf
ORA-00280: change 1120099 for thread 1 is in sequence #40
Specify log: {<RET>=suggested | filename | AUTO | CANCEL}
AUTO
ORA-00308: cannot open archived log
'/logs/WAFFLE/arch/1_40_912662036.dbf'
ORA-27037: unable to obtain file status
Linux-x86_64 Error: 2: No such file or directory
Additional information: 3
ORA-00308: cannot open archived log
'/logs/WAFFLE/arch/1_40_912662036.dbf'
ORA-27037: unable to obtain file status
Linux-x86_64 Error: 2: No such file or directory
Additional information: 3

```

5. Nach Abschluss müssen die Wiederherstellungsprotokolle erneut wiedergegeben werden. Wenn die Meldung angezeigt wird `Media recovery complete` Wird zurückgegeben, der Prozess ist erfolgreich und die Datenbanken sind synchronisiert und können geöffnet werden.

```

SQL> recover database;
Media recovery complete.
SQL> alter database open;
Database altered.

```

Protokollversand: ASM an Dateisystem

In diesem Beispiel wird die Verwendung von Oracle RMAN zur Migration einer Datenbank demonstriert. Es ähnelt dem vorherigen Beispiel des Dateisystems zum Protokollversand des Dateisystems, aber die Dateien auf ASM sind für den Host nicht sichtbar. Die einzigen Optionen für die Migration von Daten auf ASM-Geräten sind entweder die Verlagerung der ASM-LUN oder die Durchführung der Kopiervorgänge mithilfe von Oracle RMAN.

Auch wenn RMAN für das Kopieren von Dateien aus Oracle ASM erforderlich ist, ist die Verwendung von RMAN nicht auf ASM beschränkt. Mit RMAN können beliebige Storage-Typen zu beliebigen anderen Storage-Typen migriert werden.

Dieses Beispiel zeigt die Verlagerung einer Datenbank namens PANCAKE aus dem ASM-Speicher in ein normales Dateisystem, das sich auf einem anderen Server in Pfaden befindet `/oradata` Und `/logs`.

Erstellen Sie eine Datenbanksicherung

Im ersten Schritt wird ein Backup der Datenbank erstellt, die auf einen alternativen Server migriert werden soll. Da die Quelle Oracle ASM verwendet, muss RMAN verwendet werden. Ein einfaches RMAN-Backup kann wie folgt durchgeführt werden. Diese Methode erstellt ein getaggttes Backup, das später im Verfahren von RMAN

leicht identifiziert werden kann.

Der erste Befehl definiert den Zieltyp für das Backup und den zu verwendenden Speicherort. Die zweite initiiert nur die Sicherung der Datendateien.

```

RMAN> configure channel device type disk format '/rman/pancake/%U';
using target database control file instead of recovery catalog
old RMAN configuration parameters:
CONFIGURE CHANNEL DEVICE TYPE DISK FORMAT    '/rman/pancake/%U';
new RMAN configuration parameters:
CONFIGURE CHANNEL DEVICE TYPE DISK FORMAT    '/rman/pancake/%U';
new RMAN configuration parameters are successfully stored
RMAN> backup database tag 'ONTAP_MIGRATION';
Starting backup at 24-MAY-16
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=251 device type=DISK
channel ORA_DISK_1: starting full datafile backup set
channel ORA_DISK_1: specifying datafile(s) in backup set
input datafile file number=00001 name=+ASM0/PANCAKE/system01.dbf
input datafile file number=00002 name=+ASM0/PANCAKE/sysaux01.dbf
input datafile file number=00003 name=+ASM0/PANCAKE/undotbs101.dbf
input datafile file number=00004 name=+ASM0/PANCAKE/users01.dbf
channel ORA_DISK_1: starting piece 1 at 24-MAY-16
channel ORA_DISK_1: finished piece 1 at 24-MAY-16
piece handle=/rman/pancake/lgr6c161_1_1 tag=ONTAP_MIGRATION comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:03
channel ORA_DISK_1: starting full datafile backup set
channel ORA_DISK_1: specifying datafile(s) in backup set
including current control file in backup set
including current SPFILE in backup set
channel ORA_DISK_1: starting piece 1 at 24-MAY-16
channel ORA_DISK_1: finished piece 1 at 24-MAY-16
piece handle=/rman/pancake/lhr6c164_1_1 tag=ONTAP_MIGRATION comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:01
Finished backup at 24-MAY-16

```

Sicherungscontrolfile

Im weiteren Verlauf des Verfahrens wird eine Sicherungscontrolfile benötigt duplicate database Betrieb.

```

RMAN> backup current controlfile format '/rman/pancake/ctrl.bkp';
Starting backup at 24-MAY-16
using channel ORA_DISK_1
channel ORA_DISK_1: starting full datafile backup set
channel ORA_DISK_1: specifying datafile(s) in backup set
including current control file in backup set
channel ORA_DISK_1: starting piece 1 at 24-MAY-16
channel ORA_DISK_1: finished piece 1 at 24-MAY-16
piece handle=/rman/pancake/ctrl.bkp tag=TAG20160524T032651 comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:01
Finished backup at 24-MAY-16

```

Parameterdatei sichern

In der neuen Umgebung ist auch eine Parameterdatei erforderlich. Die einfachste Methode ist, aus dem aktuellen spfile oder pfile ein pfile zu erstellen. In diesem Beispiel verwendet die Quelldatenbank ein spfile.

```

RMAN> create pfile='/rman/pancake/pfile' from spfile;
Statement processed

```

Skript zum Umbenennen der ASM-Datei

Mehrere aktuell in den Steuerdateien definierte Dateispeicherorte ändern sich, wenn die Datenbank verschoben wird. Mit dem folgenden Skript wird ein RMAN-Skript erstellt, um den Prozess zu vereinfachen. Dieses Beispiel zeigt eine Datenbank mit einer sehr kleinen Anzahl von Datendateien, aber in der Regel enthalten Datenbanken Hunderte oder gar Tausende von Datendateien.

Dieses Skript finden Sie in ["Namenskonvertierung von ASM in Dateisystem"](#) Und es tut zwei Dinge.

Zuerst erstellt es einen Parameter, um die Speicherort des Wiederherstellungsprotokolls neu zu definieren `log_file_name_convert`. Es handelt sich im Wesentlichen um eine Liste von abwechselnden Feldern. Das erste Feld ist der Speicherort eines aktuellen Wiederherstellungsprotokolls und das zweite Feld ist der Speicherort auf dem neuen Server. Das Muster wird dann wiederholt.

Die zweite Funktion ist die Bereitstellung einer Vorlage für die Umbenennung von Datendateien. Das Skript führt eine Schleife durch die Datendateien durch, ruft den Namen und die Dateinummer ab und formatiert sie als RMAN-Skript. Dann macht es das gleiche mit den temporären Dateien. Das Ergebnis ist ein einfaches rman-Skript, das nach Bedarf bearbeitet werden kann, um sicherzustellen, dass die Dateien an dem gewünschten Speicherort wiederhergestellt werden.

```

SQL> @/rman/mk.rename.scripts.sql
Parameters for log file conversion:
*.log_file_name_convert = '+ASM0/PANCAKE/redo01.log',
'/NEW_PATH/redo01.log', '+ASM0/PANCAKE/redo02.log',
'/NEW_PATH/redo02.log', '+ASM0/PANCAKE/redo03.log', '/NEW_PATH/redo03.log'
rman duplication script:
run
{
set newname for datafile 1 to '+ASM0/PANCAKE/system01.dbf';
set newname for datafile 2 to '+ASM0/PANCAKE/sysaux01.dbf';
set newname for datafile 3 to '+ASM0/PANCAKE/undotbs101.dbf';
set newname for datafile 4 to '+ASM0/PANCAKE/users01.dbf';
set newname for tempfile 1 to '+ASM0/PANCAKE/temp01.dbf';
duplicate target database for standby backup location INSERT_PATH_HERE;
}
PL/SQL procedure successfully completed.

```

Erfassen Sie die Ausgabe dieses Bildschirms. Der `log_file_name_convert` Parameter wird wie unten beschrieben in `pfile` platziert. Die RMAN-Datendatei umbenennen und das doppelte Skript müssen entsprechend bearbeitet werden, um die Datendateien an den gewünschten Speicherorten zu platzieren. In diesem Beispiel werden sie alle in `/oradata/pancake` platziert.

```

run
{
set newname for datafile 1 to '/oradata/pancake/pancake.dbf';
set newname for datafile 2 to '/oradata/pancake/sysaux.dbf';
set newname for datafile 3 to '/oradata/pancake/undotbs1.dbf';
set newname for datafile 4 to '/oradata/pancake/users.dbf';
set newname for tempfile 1 to '/oradata/pancake/temp.dbf';
duplicate target database for standby backup location '/rman/pancake';
}

```

Verzeichnisstruktur vorbereiten

Die Skripte sind fast fertig zur Ausführung, aber zuerst muss die Verzeichnisstruktur vorhanden sein. Wenn die benötigten Verzeichnisse nicht bereits vorhanden sind, müssen sie erstellt werden, oder der Datenbankstartvorgang schlägt fehl. Das folgende Beispiel gibt die Mindestanforderungen wieder.

```

[oracle@jpsc2 ~]$ mkdir /oradata/pancake
[oracle@jpsc2 ~]$ mkdir /logs/pancake
[oracle@jpsc2 ~]$ cd /orabin/admin
[oracle@jpsc2 admin]$ mkdir PANCAKE
[oracle@jpsc2 admin]$ cd PANCAKE
[oracle@jpsc2 PANCAKE]$ mkdir adump dpdump pfile scripts xdb_wallet

```

Oratab-Eintrag erstellen

Der folgende Befehl ist für Dienstprogramme wie oraenv erforderlich, um ordnungsgemäß zu funktionieren.

```
PANCAKE:/orabin/product/12.1.0/dbhome_1:N
```

Parameteraktualisierungen

Die gespeicherte pfile muss aktualisiert werden, um alle Pfadänderungen auf dem neuen Server widerzuspiegeln. Die Änderungen des Datendateipfads werden durch das RMAN-Duplizierungsskript geändert, und fast alle Datenbanken erfordern Änderungen am `control_files` Und `log_archive_dest` Parameter. Es können auch Prüfdateipositionen vorhanden sein, die geändert werden müssen, und Parameter wie `db_create_file_dest` Ist außerhalb von ASM möglicherweise nicht relevant. Ein erfahrener DBA sollte die vorgeschlagenen Änderungen sorgfältig prüfen, bevor er fortfahren kann.

In diesem Beispiel sind die wichtigsten Änderungen die Speicherorte der Steuerdatei, das Protokollarchivziel und das Hinzufügen des `log_file_name_convert` Parameter.

```

PANCAKE.__data_transfer_cache_size=0
PANCAKE.__db_cache_size=545259520
PANCAKE.__java_pool_size=4194304
PANCAKE.__large_pool_size=25165824
PANCAKE.__oracle_base='/orabin'#ORACLE_BASE set from environment
PANCAKE.__pga_aggregate_target=268435456
PANCAKE.__sga_target=805306368
PANCAKE.__shared_io_pool_size=29360128
PANCAKE.__shared_pool_size=192937984
PANCAKE.__streams_pool_size=0
*.audit_file_dest='/orabin/admin/PANCAKE/adump'
*.audit_trail='db'
*.compatible='12.1.0.2.0'
*.control_files='+ASM0/PANCAKE/control01.ctl','+ASM0/PANCAKE/control02.ctl'
*.control_files='/oradata/pancake/control01.ctl','/logs/pancake/control02.ctl'
*.db_block_size=8192
*.db_domain=''
*.db_name='PANCAKE'
*.diagnostic_dest='/orabin'
*.dispatchers='(PROTOCOL=TCP) (SERVICE=PANCAKEXDB)'
*.log_archive_dest_1='LOCATION=+ASM1'
*.log_archive_dest_1='LOCATION=/logs/pancake'
*.log_archive_format='%t_%s_%r.dbf'
'/logs/path/redo02.log'
*.log_file_name_convert = '+ASM0/PANCAKE/redo01.log',
'/logs/pancake/redo01.log', '+ASM0/PANCAKE/redo02.log',
'/logs/pancake/redo02.log', '+ASM0/PANCAKE/redo03.log',
'/logs/pancake/redo03.log'
*.open_cursors=300
*.pga_aggregate_target=256m
*.processes=300
*.remote_login_passwordfile='EXCLUSIVE'
*.sga_target=768m
*.undo_tablespace='UNDOTBS1'

```

Nachdem die neuen Parameter bestätigt wurden, müssen die Parameter wirksam werden. Es gibt mehrere Optionen, aber die meisten Kunden erstellen ein Spfile basierend auf dem Text pfile.

```
bash-4.1$ sqlplus / as sysdba
SQL*Plus: Release 12.1.0.2.0 Production on Fri Jan 8 11:17:40 2016
Copyright (c) 1982, 2014, Oracle. All rights reserved.
Connected to an idle instance.
SQL> create spfile from pfile='/rman/pancake/pfile';
File created.
```

Startbezeichnung

Der letzte Schritt vor dem Replizieren der Datenbank ist, die Datenbankprozesse zu laden, aber nicht die Dateien zu mounten. In diesem Schritt können Probleme mit dem spfile offensichtlich werden. Wenn der `startup nomount` Befehl schlägt aufgrund eines Parameterfehlers fehl, es ist einfach herunterzufahren, die `pfile`-Vorlage zu korrigieren, sie als `spfile` neu zu laden und es erneut zu versuchen.

```
SQL> startup nomount;
ORACLE instance started.
Total System Global Area 805306368 bytes
Fixed Size 2929552 bytes
Variable Size 373296240 bytes
Database Buffers 423624704 bytes
Redo Buffers 5455872 bytes
```

Duplizieren Sie die Datenbank

Die Wiederherstellung des vorherigen RMAN-Backups am neuen Speicherort nimmt mehr Zeit in Anspruch als andere Schritte in diesem Prozess. Die Datenbank muss ohne Änderung der Datenbank-ID (DBID) oder Zurücksetzen der Protokolle dupliziert werden. Dadurch wird verhindert, dass Protokolle angewendet werden, was ein erforderlicher Schritt zur vollständigen Synchronisierung der Kopien ist.

Stellen Sie mit RMAN als AUX eine Verbindung zur Datenbank her, und geben Sie den Befehl Duplicate Database aus, indem Sie das in einem vorherigen Schritt erstellte Skript verwenden.

```
[oracle@jfsc2 pancake]$ rman auxiliary /
Recovery Manager: Release 12.1.0.2.0 - Production on Tue May 24 03:04:56
2016
Copyright (c) 1982, 2014, Oracle and/or its affiliates. All rights
reserved.
connected to auxiliary database: PANCAKE (not mounted)
RMAN> run
2> {
3> set newname for datafile 1 to '/oradata/pancake/pancake.dbf';
4> set newname for datafile 2 to '/oradata/pancake/sysaux.dbf';
5> set newname for datafile 3 to '/oradata/pancake/undotbs1.dbf';
6> set newname for datafile 4 to '/oradata/pancake/users.dbf';
7> set newname for tempfile 1 to '/oradata/pancake/temp.dbf';
8> duplicate target database for standby backup location '/rman/pancake';
```

```

9> }
executing command: SET NEWNAME
Starting Duplicate Db at 24-MAY-16
contents of Memory Script:
{
  restore clone standby controlfile from  '/rman/pancake/ctrl.bkp';
}
executing Memory Script
Starting restore at 24-MAY-16
allocated channel: ORA_AUX_DISK_1
channel ORA_AUX_DISK_1: SID=243 device type=DISK
channel ORA_AUX_DISK_1: restoring control file
channel ORA_AUX_DISK_1: restore complete, elapsed time: 00:00:01
output file name=/oradata/pancake/control01.ctl
output file name=/logs/pancake/control02.ctl
Finished restore at 24-MAY-16
contents of Memory Script:
{
  sql clone 'alter database mount standby database';
}
executing Memory Script
sql statement: alter database mount standby database
released channel: ORA_AUX_DISK_1
allocated channel: ORA_AUX_DISK_1
channel ORA_AUX_DISK_1: SID=243 device type=DISK
contents of Memory Script:
{
  set newname for tempfile 1 to
"/oradata/pancake/temp.dbf";
  switch clone tempfile all;
  set newname for datafile 1 to
"/oradata/pancake/pancake.dbf";
  set newname for datafile 2 to
"/oradata/pancake/sysaux.dbf";
  set newname for datafile 3 to
"/oradata/pancake/undotbs1.dbf";
  set newname for datafile 4 to
"/oradata/pancake/users.dbf";
  restore
  clone database
;
}

```

```
executing Memory Script
executing command: SET NEWNAME
renamed tempfile 1 to /oradata/pancake/temp.dbf in control file
executing command: SET NEWNAME
executing command: SET NEWNAME
executing command: SET NEWNAME
executing command: SET NEWNAME
Starting restore at 24-MAY-16
using channel ORA_AUX_DISK_1
channel ORA_AUX_DISK_1: starting datafile backup set restore
channel ORA_AUX_DISK_1: specifying datafile(s) to restore from backup set
channel ORA_AUX_DISK_1: restoring datafile 00001 to
/oradata/pancake/pancake.dbf
channel ORA_AUX_DISK_1: restoring datafile 00002 to
/oradata/pancake/sysaux.dbf
channel ORA_AUX_DISK_1: restoring datafile 00003 to
/oradata/pancake/undotbs1.dbf
channel ORA_AUX_DISK_1: restoring datafile 00004 to
/oradata/pancake/users.dbf
channel ORA_AUX_DISK_1: reading from backup piece
/rman/pancake/1gr6c161_1_1
channel ORA_AUX_DISK_1: piece handle=/rman/pancake/1gr6c161_1_1
tag=ONTAP_MIGRATION
channel ORA_AUX_DISK_1: restored backup piece 1
channel ORA_AUX_DISK_1: restore complete, elapsed time: 00:00:07
Finished restore at 24-MAY-16
contents of Memory Script:
{
  switch clone datafile all;
}
executing Memory Script
datafile 1 switched to datafile copy
input datafile copy RECID=5 STAMP=912655725 file
name=/oradata/pancake/pancake.dbf
datafile 2 switched to datafile copy
input datafile copy RECID=6 STAMP=912655725 file
name=/oradata/pancake/sysaux.dbf
datafile 3 switched to datafile copy
input datafile copy RECID=7 STAMP=912655725 file
name=/oradata/pancake/undotbs1.dbf
datafile 4 switched to datafile copy
input datafile copy RECID=8 STAMP=912655725 file
name=/oradata/pancake/users.dbf
Finished Duplicate Db at 24-MAY-16
```

Erste Protokollreplizierung

Sie müssen die Änderungen nun von der Quelldatenbank an einen neuen Speicherort senden. Dies kann eine Kombination von Schritten erfordern. Die einfachste Methode wäre, RMAN auf der Quelldatenbank zu haben, um Archivprotokolle auf eine freigegebene Netzwerkverbindung zu schreiben. Wenn ein freigegebener Speicherort nicht verfügbar ist, verwenden Sie RMAN zum Schreiben auf ein lokales Dateisystem und anschließend `rcp` oder `rsync` zum Kopieren der Dateien.

In diesem Beispiel ist der `/rman` Verzeichnis ist eine NFS-Freigabe, die sowohl für die ursprüngliche als auch für die migrierte Datenbank verfügbar ist.

Ein wichtiges Thema ist hier die `disk format` Klausel. Das Festplattenformat des Backups ist `%h_%e_%a.dbf`, Das bedeutet, dass Sie das Format der Thread-Nummer, Sequenznummer und Aktivierungs-ID für die Datenbank verwenden müssen. Obwohl die Buchstaben unterschiedlich sind, entspricht dies der `log_archive_format='%t_%s_%r.dbf` Parameter in `pfile`. Mit diesem Parameter werden auch Archivprotokolle im Format Thread-Nummer, Sequenznummer und Aktivierungs-ID angegeben. Das Endergebnis ist, dass die Protokolldatei-Backups auf der Quelle eine Benennungskonvention verwenden, die von der Datenbank erwartet wird. Dadurch werden z. B. Operationen wie die `recover database` Viel einfacher, weil `sqlplus` richtig vorwegnimmt die Namen der Archiv-Protokolle wiedergegeben werden.

```

RMAN> configure channel device type disk format
'/rman/pancake/logship/%h_%e_%a.dbf';
old RMAN configuration parameters:
CONFIGURE CHANNEL DEVICE TYPE DISK FORMAT
'/rman/pancake/arch/%h_%e_%a.dbf';
new RMAN configuration parameters:
CONFIGURE CHANNEL DEVICE TYPE DISK FORMAT
'/rman/pancake/logship/%h_%e_%a.dbf';
new RMAN configuration parameters are successfully stored
released channel: ORA_DISK_1
RMAN> backup as copy archivelog from time 'sysdate-2';
Starting backup at 24-MAY-16
current log archived
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=373 device type=DISK
channel ORA_DISK_1: starting archived log copy
input archived log thread=1 sequence=54 RECID=70 STAMP=912658508
output file name=/rman/pancake/logship/1_54_912576125.dbf RECID=123
STAMP=912659482
channel ORA_DISK_1: archived log copy complete, elapsed time: 00:00:01
channel ORA_DISK_1: starting archived log copy
input archived log thread=1 sequence=41 RECID=29 STAMP=912654101
output file name=/rman/pancake/logship/1_41_912576125.dbf RECID=124
STAMP=912659483
channel ORA_DISK_1: archived log copy complete, elapsed time: 00:00:01
...
channel ORA_DISK_1: starting archived log copy
input archived log thread=1 sequence=45 RECID=33 STAMP=912654688
output file name=/rman/pancake/logship/1_45_912576125.dbf RECID=152
STAMP=912659514
channel ORA_DISK_1: archived log copy complete, elapsed time: 00:00:01
channel ORA_DISK_1: starting archived log copy
input archived log thread=1 sequence=47 RECID=36 STAMP=912654809
output file name=/rman/pancake/logship/1_47_912576125.dbf RECID=153
STAMP=912659515
channel ORA_DISK_1: archived log copy complete, elapsed time: 00:00:01
Finished backup at 24-MAY-16

```

Erste Protokollwiedergabe

Nachdem sich die Dateien im Archiv-Log-Speicherort befinden, können sie mit dem Befehl wiedergegeben werden `recover database until cancel` Gefolgt von der Antwort `AUTO` Um alle verfügbaren Protokolle automatisch wiederzugeben. Die Parameterdatei leitet derzeit Archivprotokolle an `/logs/archive`, Aber dies stimmt nicht mit dem Speicherort überein, an dem RMAN zum Speichern von Protokollen verwendet wurde. Der Speicherort kann vor der Wiederherstellung der Datenbank wie folgt vorübergehend umgeleitet werden.

```

SQL> alter system set log_archive_dest_1='LOCATION=/rman/pancake/logship'
scope=memory;
System altered.
SQL> recover standby database until cancel;
ORA-00279: change 560224 generated at 05/24/2016 03:25:53 needed for
thread 1
ORA-00289: suggestion : /rman/pancake/logship/1_49_912576125.dbf
ORA-00280: change 560224 for thread 1 is in sequence #49
Specify log: {<RET>=suggested | filename | AUTO | CANCEL}
AUTO
ORA-00279: change 560353 generated at 05/24/2016 03:29:17 needed for
thread 1
ORA-00289: suggestion : /rman/pancake/logship/1_50_912576125.dbf
ORA-00280: change 560353 for thread 1 is in sequence #50
ORA-00278: log file '/rman/pancake/logship/1_49_912576125.dbf' no longer
needed
for this recovery
...
ORA-00279: change 560591 generated at 05/24/2016 03:33:56 needed for
thread 1
ORA-00289: suggestion : /rman/pancake/logship/1_54_912576125.dbf
ORA-00280: change 560591 for thread 1 is in sequence #54
ORA-00278: log file '/rman/pancake/logship/1_53_912576125.dbf' no longer
needed
for this recovery
ORA-00308: cannot open archived log
'/rman/pancake/logship/1_54_912576125.dbf'
ORA-27037: unable to obtain file status
Linux-x86_64 Error: 2: No such file or directory
Additional information: 3

```

Die endgültige Antwort des Archivprotokolls meldet einen Fehler. Dies ist jedoch normal. Der Fehler zeigt an, dass sqlplus eine bestimmte Protokolldatei gesucht und nicht gefunden hat. Der Grund dafür ist sehr wahrscheinlich, dass die Protokolldatei noch nicht existiert.

Wenn die Quelldatenbank vor dem Kopieren von Archivprotokollen heruntergefahren werden kann, muss dieser Schritt nur einmal durchgeführt werden. Die Archivprotokolle werden kopiert und eingespielt. Anschließend kann der Prozess direkt zum Umstellungsprozess fortgesetzt werden, der die kritischen Wiederherstellungsprotokolle repliziert.

Inkrementelle Protokollreplikation und -Wiedergabe

In den meisten Fällen erfolgt die Migration nicht sofort. Es kann Tage oder sogar Wochen bis zum Abschluss des Migrationsprozesses dauern. Das bedeutet, dass die Protokolle kontinuierlich an die Replikatdatenbank gesendet und wieder eingespielt werden müssen. So ist sichergestellt, dass bei der Umstellung nur minimale Daten übertragen und eingespielt werden müssen.

Dieser Prozess kann einfach per Skript ausgeführt werden. Beispielsweise kann der folgende Befehl für die

ursprüngliche Datenbank geplant werden, um sicherzustellen, dass der für den Protokollversand verwendete Speicherort fortlaufend aktualisiert wird.

```
[oracle@jfscl pancake]$ cat copylogs.rman
configure channel device type disk format
'/rman/pancake/logship/%h_%e_%a.dbf';
backup as copy archivelog from time 'sysdate-2';
```

```
[oracle@jfscl pancake]$ rman target / cmdfile=copylogs.rman
Recovery Manager: Release 12.1.0.2.0 - Production on Tue May 24 04:36:19
2016
Copyright (c) 1982, 2014, Oracle and/or its affiliates. All rights
reserved.
connected to target database: PANCAKE (DBID=3574534589)
RMAN> configure channel device type disk format
'/rman/pancake/logship/%h_%e_%a.dbf';
2> backup as copy archivelog from time 'sysdate-2';
3>
4>
using target database control file instead of recovery catalog
old RMAN configuration parameters:
CONFIGURE CHANNEL DEVICE TYPE DISK FORMAT
'/rman/pancake/logship/%h_%e_%a.dbf';
new RMAN configuration parameters:
CONFIGURE CHANNEL DEVICE TYPE DISK FORMAT
'/rman/pancake/logship/%h_%e_%a.dbf';
new RMAN configuration parameters are successfully stored
Starting backup at 24-MAY-16
current log archived
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=369 device type=DISK
channel ORA_DISK_1: starting archived log copy
input archived log thread=1 sequence=54 RECID=123 STAMP=912659482
RMAN-03009: failure of backup command on ORA_DISK_1 channel at 05/24/2016
04:36:22
ORA-19635: input and output file names are identical:
/rman/pancake/logship/1_54_912576125.dbf
continuing other job steps, job failed will not be re-run
channel ORA_DISK_1: starting archived log copy
input archived log thread=1 sequence=41 RECID=124 STAMP=912659483
RMAN-03009: failure of backup command on ORA_DISK_1 channel at 05/24/2016
04:36:23
ORA-19635: input and output file names are identical:
/rman/pancake/logship/1_41_912576125.dbf
continuing other job steps, job failed will not be re-run
```

```

...
channel ORA_DISK_1: starting archived log copy
input archived log thread=1 sequence=45 RECID=152 STAMP=912659514
RMAN-03009: failure of backup command on ORA_DISK_1 channel at 05/24/2016
04:36:55
ORA-19635: input and output file names are identical:
/rman/pancake/logship/1_45_912576125.dbf
continuing other job steps, job failed will not be re-run
channel ORA_DISK_1: starting archived log copy
input archived log thread=1 sequence=47 RECID=153 STAMP=912659515
RMAN-00571: =====
RMAN-00569: ===== ERROR MESSAGE STACK FOLLOWS =====
RMAN-00571: =====
RMAN-03009: failure of backup command on ORA_DISK_1 channel at 05/24/2016
04:36:57
ORA-19635: input and output file names are identical:
/rman/pancake/logship/1_47_912576125.dbf
Recovery Manager complete.

```

Nachdem die Protokolle empfangen wurden, müssen sie erneut abgespielt werden. Frühere Beispiele zeigten die Verwendung von sqlplus zum manuellen Ausführen `recover database until cancel`, Die leicht automatisiert werden kann. Das hier abgebildete Beispiel verwendet das in beschriebene Skript "[Wiedergabe von Protokollen in der Standby-Datenbank](#)". Das Skript akzeptiert ein Argument, das die Datenbank angibt, für die eine Wiedergabeoperation erforderlich ist. Bei diesem Prozess kann dasselbe Skript für eine Migration mit mehreren Datenbanken verwendet werden.

```
[root@jpsc2 pancake]# ./replaylogs.pl PANCAKE
ORACLE_SID = [oracle] ? The Oracle base has been set to /orabin
SQL*Plus: Release 12.1.0.2.0 Production on Tue May 24 04:47:10 2016
Copyright (c) 1982, 2014, Oracle. All rights reserved.
Connected to:
Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 - 64bit
Production
With the Partitioning, OLAP, Advanced Analytics and Real Application
Testing options
SQL> ORA-00279: change 560591 generated at 05/24/2016 03:33:56 needed for
thread 1
ORA-00289: suggestion : /rman/pancake/logship/1_54_912576125.dbf
ORA-00280: change 560591 for thread 1 is in sequence #54
Specify log: {<RET>=suggested | filename | AUTO | CANCEL}
ORA-00279: change 562219 generated at 05/24/2016 04:15:08 needed for
thread 1
ORA-00289: suggestion : /rman/pancake/logship/1_55_912576125.dbf
ORA-00280: change 562219 for thread 1 is in sequence #55
ORA-00278: log file '/rman/pancake/logship/1_54_912576125.dbf' no longer
needed for this recovery
ORA-00279: change 562370 generated at 05/24/2016 04:19:18 needed for
thread 1
ORA-00289: suggestion : /rman/pancake/logship/1_56_912576125.dbf
ORA-00280: change 562370 for thread 1 is in sequence #56
ORA-00278: log file '/rman/pancake/logship/1_55_912576125.dbf' no longer
needed for this recovery
...
ORA-00279: change 563137 generated at 05/24/2016 04:36:20 needed for
thread 1
ORA-00289: suggestion : /rman/pancake/logship/1_65_912576125.dbf
ORA-00280: change 563137 for thread 1 is in sequence #65
ORA-00278: log file '/rman/pancake/logship/1_64_912576125.dbf' no longer
needed for this recovery
ORA-00308: cannot open archived log
'/rman/pancake/logship/1_65_912576125.dbf'
ORA-27037: unable to obtain file status
Linux-x86_64 Error: 2: No such file or directory
Additional information: 3
SQL> Disconnected from Oracle Database 12c Enterprise Edition Release
12.1.0.2.0 - 64bit Production
With the Partitioning, OLAP, Advanced Analytics and Real Application
Testing options
```

Umstellung

Wenn Sie bereit sind, in die neue Umgebung zu schneiden, müssen Sie eine abschließende Synchronisierung durchführen. Bei der Arbeit mit normalen Dateisystemen ist es leicht sicherzustellen, dass die migrierte Datenbank zu 100 % mit dem Original synchronisiert wird, da die ursprünglichen Wiederherstellungsprotokolle kopiert und wiedergegeben werden. Es gibt keinen guten Weg, dies mit ASM zu tun. Nur die Archivprotokolle können einfach wiederaufgenommen werden. Um sicherzustellen, dass keine Daten verloren gehen, muss das endgültige Herunterfahren der ursprünglichen Datenbank sorgfältig durchgeführt werden.

1. Zunächst muss die Datenbank stillgelegt werden, um sicherzustellen, dass keine Änderungen vorgenommen werden. Diese Stilllegung kann die Deaktivierung geplanter Vorgänge, das Herunterfahren von Listnern und/oder das Herunterfahren von Anwendungen umfassen.
2. Nach diesem Schritt erstellen die meisten DBAs eine Dummy-Tabelle, die als Marker für das Herunterfahren dient.
3. Erzwingen Sie eine Protokollarchivierung, um sicherzustellen, dass die Erstellung der Dummy-Tabelle in den Archivprotokollen aufgezeichnet wird. Führen Sie dazu die folgenden Befehle aus:

```
SQL> create table cutovercheck as select * from dba_users;
Table created.
SQL> alter system archive log current;
System altered.
SQL> shutdown immediate;
Database closed.
Database dismounted.
ORACLE instance shut down.
```

4. Führen Sie die folgenden Befehle aus, um die letzten Archivprotokolle zu kopieren. Die Datenbank muss verfügbar, aber nicht geöffnet sein.

```
SQL> startup mount;
ORACLE instance started.
Total System Global Area 805306368 bytes
Fixed Size 2929552 bytes
Variable Size 331353200 bytes
Database Buffers 465567744 bytes
Redo Buffers 5455872 bytes
Database mounted.
```

5. Um die Archivprotokolle zu kopieren, führen Sie die folgenden Befehle aus:

```

RMAN> configure channel device type disk format
'/rman/pancake/logship/%h_%e_%a.dbf';
2> backup as copy archivelog from time 'sysdate-2';
3>
4>
using target database control file instead of recovery catalog
old RMAN configuration parameters:
CONFIGURE CHANNEL DEVICE TYPE DISK FORMAT
'/rman/pancake/logship/%h_%e_%a.dbf';
new RMAN configuration parameters:
CONFIGURE CHANNEL DEVICE TYPE DISK FORMAT
'/rman/pancake/logship/%h_%e_%a.dbf';
new RMAN configuration parameters are successfully stored
Starting backup at 24-MAY-16
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=8 device type=DISK
channel ORA_DISK_1: starting archived log copy
input archived log thread=1 sequence=54 RECID=123 STAMP=912659482
RMAN-03009: failure of backup command on ORA_DISK_1 channel at
05/24/2016 04:58:24
ORA-19635: input and output file names are identical:
/rman/pancake/logship/1_54_912576125.dbf
continuing other job steps, job failed will not be re-run
...
channel ORA_DISK_1: starting archived log copy
input archived log thread=1 sequence=45 RECID=152 STAMP=912659514
RMAN-03009: failure of backup command on ORA_DISK_1 channel at
05/24/2016 04:58:58
ORA-19635: input and output file names are identical:
/rman/pancake/logship/1_45_912576125.dbf
continuing other job steps, job failed will not be re-run
channel ORA_DISK_1: starting archived log copy
input archived log thread=1 sequence=47 RECID=153 STAMP=912659515
RMAN-00571: =====
RMAN-00569: ===== ERROR MESSAGE STACK FOLLOWS =====
RMAN-00571: =====
RMAN-03009: failure of backup command on ORA_DISK_1 channel at
05/24/2016 04:59:00
ORA-19635: input and output file names are identical:
/rman/pancake/logship/1_47_912576125.dbf

```

6. Geben Sie abschließend die restlichen Archivprotokolle auf dem neuen Server wieder.

```

[root@jpsc2 pancake]# ./replaylogs.pl PANCAKE
ORACLE_SID = [oracle] ? The Oracle base has been set to /orabin
SQL*Plus: Release 12.1.0.2.0 Production on Tue May 24 05:00:53 2016
Copyright (c) 1982, 2014, Oracle. All rights reserved.
Connected to:
Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 - 64bit
Production
With the Partitioning, OLAP, Advanced Analytics and Real Application
Testing options
SQL> ORA-00279: change 563137 generated at 05/24/2016 04:36:20 needed
for thread 1
ORA-00289: suggestion : /rman/pancake/logship/1_65_912576125.dbf
ORA-00280: change 563137 for thread 1 is in sequence #65
Specify log: {<RET>=suggested | filename | AUTO | CANCEL}
ORA-00279: change 563629 generated at 05/24/2016 04:55:20 needed for
thread 1
ORA-00289: suggestion : /rman/pancake/logship/1_66_912576125.dbf
ORA-00280: change 563629 for thread 1 is in sequence #66
ORA-00278: log file '/rman/pancake/logship/1_65_912576125.dbf' no longer
needed
for this recovery
ORA-00308: cannot open archived log
'/rman/pancake/logship/1_66_912576125.dbf'
ORA-27037: unable to obtain file status
Linux-x86_64 Error: 2: No such file or directory
Additional information: 3
SQL> Disconnected from Oracle Database 12c Enterprise Edition Release
12.1.0.2.0 - 64bit Production
With the Partitioning, OLAP, Advanced Analytics and Real Application
Testing options

```

7. In dieser Phase sollten Sie alle Daten replizieren. Die Datenbank kann von einer Standby-Datenbank in eine aktive Betriebsdatenbank konvertiert und dann geöffnet werden.

```

SQL> alter database activate standby database;
Database altered.
SQL> alter database open;
Database altered.

```

8. Bestätigen Sie das Vorhandensein der Dummy-Tabelle und legen Sie sie dann ab.

```

SQL> desc cutovercheck
Name                                                    Null?    Type
-----
-----
USERNAME                                                NOT NULL VARCHAR2 (128)
USER_ID                                                  NOT NULL NUMBER
PASSWORD                                                VARCHAR2 (4000)
ACCOUNT_STATUS                                          NOT NULL VARCHAR2 (32)
LOCK_DATE                                               DATE
EXPIRY_DATE                                             DATE
DEFAULT_TABLESPACE                                     NOT NULL VARCHAR2 (30)
TEMPORARY_TABLESPACE                                  NOT NULL VARCHAR2 (30)
CREATED                                                 NOT NULL DATE
PROFILE                                                 NOT NULL VARCHAR2 (128)
INITIAL_RSRC_CONSUMER_GROUP                            VARCHAR2 (128)
EXTERNAL_NAME                                           VARCHAR2 (4000)
PASSWORD_VERSIONS                                       VARCHAR2 (12)
EDITIONS_ENABLED                                       VARCHAR2 (1)
AUTHENTICATION_TYPE                                    VARCHAR2 (8)
PROXY_ONLY_CONNECT                                    VARCHAR2 (1)
COMMON                                                  VARCHAR2 (3)
LAST_LOGIN                                              TIMESTAMP (9) WITH
TIME_ZONE
ORACLE_MAINTAINED                                       VARCHAR2 (1)
SQL> drop table cutovercheck;
Table dropped.

```

Unterbrechungsfreie Migration von Wiederherstellungsprotokollen

Es gibt Zeiten, in denen eine Datenbank insgesamt korrekt organisiert ist, mit Ausnahme der Wiederherstellungsprotokolle. Dies kann aus vielen Gründen geschehen, von denen die häufigste im Zusammenhang mit Snapshots steht. Produkte wie SnapManager für Oracle, SnapCenter und das Storage Management Framework NetApp Snap Creator ermöglichen eine nahezu sofortige Wiederherstellung einer Datenbank, jedoch nur, wenn Sie den Zustand der Daten-File-Volumes zurücksetzen. Wenn Redo-Logs Speicherplatz mit den Datendateien teilen, kann Reversion nicht sicher ausgeführt werden, da es zur Zerstörung der Redo-Protokolle führen würde, was wahrscheinlich Datenverlust bedeutet. Daher müssen die Redo-Logs verschoben werden.

Dieses Verfahren ist einfach und unterbrechungsfrei.

Aktuelle Konfiguration des Wiederherstellungsprotokolls

1. Ermitteln Sie die Anzahl der Redo-Log-Gruppen und deren jeweilige Gruppennummern.

```

SQL> select group#||' '||member from v$logfile;
GROUP#||' '||MEMBER
-----
-----
1 /redo0/NTAP/redo01a.log
1 /redo1/NTAP/redo01b.log
2 /redo0/NTAP/redo02a.log
2 /redo1/NTAP/redo02b.log
3 /redo0/NTAP/redo03a.log
3 /redo1/NTAP/redo03b.log
rows selected.

```

2. Geben Sie die Größe der Wiederherstellungsprotokolle ein.

```

SQL> select group#||' '||bytes from v$log;
GROUP#||' '||BYTES
-----
-----
1 524288000
2 524288000
3 524288000

```

Erstellen Sie neue Protokolle

1. Erstellen Sie für jedes Redo-Protokoll eine neue Gruppe mit einer passenden Größe und Anzahl von Mitgliedern.

```

SQL> alter database add logfile ('/newredo0/redo01a.log',
'/newredo1/redo01b.log') size 500M;
Database altered.
SQL> alter database add logfile ('/newredo0/redo02a.log',
'/newredo1/redo02b.log') size 500M;
Database altered.
SQL> alter database add logfile ('/newredo0/redo03a.log',
'/newredo1/redo03b.log') size 500M;
Database altered.
SQL>

```

2. Überprüfen Sie die neue Konfiguration.

```

SQL> select group#||' '||member from v$logfile;
GROUP#||' '||MEMBER
-----
-----
1 /redo0/NTAP/redo01a.log
1 /redo1/NTAP/redo01b.log
2 /redo0/NTAP/redo02a.log
2 /redo1/NTAP/redo02b.log
3 /redo0/NTAP/redo03a.log
3 /redo1/NTAP/redo03b.log
4 /newredo0/redo01a.log
4 /newredo1/redo01b.log
5 /newredo0/redo02a.log
5 /newredo1/redo02b.log
6 /newredo0/redo03a.log
6 /newredo1/redo03b.log
12 rows selected.

```

Alte Protokolle ablegen

1. Löschen Sie die alten Protokolle (Gruppen 1, 2 und 3).

```

SQL> alter database drop logfile group 1;
Database altered.
SQL> alter database drop logfile group 2;
Database altered.
SQL> alter database drop logfile group 3;
Database altered.

```

2. Wenn ein Fehler auftritt, der verhindert, dass Sie ein aktives Protokoll ablegen, erzwingen Sie einen Wechsel zum nächsten Protokoll, um die Sperre freizugeben und einen globalen Kontrollpunkt zu erzwingen. Siehe folgendes Beispiel für diesen Prozess. Der Versuch, die Logfile-Gruppe 2, die sich am alten Speicherort befand, zu löschen, wurde abgelehnt, da noch aktive Daten in dieser Logdatei vorhanden waren.

```

SQL> alter database drop logfile group 2;
alter database drop logfile group 2
*
ERROR at line 1:
ORA-01623: log 2 is current log for instance NTAP (thread 1) - cannot
drop
ORA-00312: online log 2 thread 1: '/redo0/NTAP/redo02a.log'
ORA-00312: online log 2 thread 1: '/redo1/NTAP/redo02b.log'

```

3. Eine Protokollarchivierung, gefolgt von einem Kontrollpunkt, ermöglicht es Ihnen, die Protokolldatei zu löschen.

```
SQL> alter system archive log current;
System altered.
SQL> alter system checkpoint;
System altered.
SQL> alter database drop logfile group 2;
Database altered.
```

4. Löschen Sie anschließend die Protokolle aus dem Dateisystem. Sie sollten diesen Vorgang mit äußerster Sorgfalt durchführen.

Host-Daten werden kopiert

Wie bei der Migration auf Datenbankebene bietet auch die Migration auf Hostebene einen vom Storage-Anbieter unabhängigen Ansatz.

Mit anderen Worten, manchmal "einfach die Dateien kopieren" ist die beste Option.

Obwohl dieser Low-Tech-Ansatz zu einfach erscheint, bietet er doch erhebliche Vorteile, da keine spezielle Software erforderlich ist und die Originaldaten während des Prozesses sicher unberührt bleiben. Die primäre Einschränkung besteht darin, dass eine Datenmigration auf Dateikopien einen unterbrechungsfreien Prozess darstellt, da die Datenbank vor Beginn des Kopiervorgangs heruntergefahren werden muss. Es gibt keine gute Möglichkeit, Änderungen innerhalb einer Datei zu synchronisieren, so dass die Dateien vollständig stillgelegt werden müssen, bevor das Kopieren beginnt.

Wenn das für einen Kopiervorgang erforderliche Herunterfahren nicht wünschenswert ist, ist die nächstbeste Host-basierte Option die Nutzung eines Logical Volume Managers (LVM). Es gibt viele LVM-Optionen, einschließlich Oracle ASM, alle mit ähnlichen Funktionen, aber auch mit einigen Einschränkungen, die berücksichtigt werden müssen. In den meisten Fällen lässt sich die Migration ohne Ausfallzeit und Unterbrechung durchführen.

Dateisystem wird kopiert

Der Nutzen einer einfachen Kopieroperation sollte nicht unterschätzt werden. Dieser Vorgang erfordert während des Kopierprozesses Ausfallzeiten, ist jedoch äußerst zuverlässig und erfordert keine besondere Expertise in Bezug auf Betriebssysteme, Datenbanken oder Speichersysteme. Darüber hinaus ist es sehr sicher, weil es die ursprünglichen Daten nicht beeinträchtigt. In der Regel ändert ein Systemadministrator die Quelldateisysteme, die schreibgeschützt gemountet werden, und startet dann einen Server neu, um zu gewährleisten, dass die aktuellen Daten nicht beschädigt werden können. Der Kopiervorgang kann mithilfe eines Skripts durchgeführt werden, um sicherzustellen, dass er so schnell wie möglich ohne das Risiko eines Benutzerfehlers ausgeführt wird. Da der I/O-Typ eine einfache, sequenzielle Datenübertragung ist, ist er eine äußerst effiziente Bandbreitennutzung.

Das folgende Beispiel zeigt eine Möglichkeit für eine sichere und schnelle Migration.

Umgebung

Die zu migrierende Umgebung ist wie folgt:

- Aktuelle Dateisysteme

```
ontap-nfs1:/host1_oradata      52428800  16196928  36231872  31%  
/oradata  
ontap-nfs1:/host1_logs        49807360   548032   49259328  2% /logs
```

- Neue Filesysteme

```
ontap-nfs1:/host1_logs_new    49807360   128   49807232  1%  
/new/logs  
ontap-nfs1:/host1_oradata_new 49807360   128   49807232  1%  
/new/oradata
```

Überblick

Die Datenbank kann von einem Datenbankadministrator migriert werden, indem er die Datenbank herunterfährt und die Dateien kopiert. Der Prozess kann jedoch problemlos Skripte erstellen, wenn viele Datenbanken migriert werden müssen oder die Ausfallzeit minimiert werden muss. Die Verwendung von Skripten verringert zudem das Risiko von Benutzerfehlern.

Die Beispielskripte automatisieren die folgenden Vorgänge:

- Die Datenbank wird heruntergefahren
- Konvertieren der vorhandenen Dateisysteme in einen schreibgeschützten Zustand
- Kopieren aller Daten von der Quelle auf Zieldateisysteme, wobei alle Dateiberechtigungen erhalten bleiben
- Heben Sie das Mounten der alten und neuen Dateisysteme auf
- Erneutes Mounten der neuen Dateisysteme in denselben Pfaden wie die vorherigen Dateisysteme

Verfahren

1. Fahren Sie die Datenbank herunter.

```

[root@host1 current]# ./dbshut.pl NTAP
ORACLE_SID = [oracle] ? The Oracle base has been set to /orabin
SQL*Plus: Release 12.1.0.2.0 Production on Thu Dec 3 15:58:48 2015
Copyright (c) 1982, 2014, Oracle. All rights reserved.
Connected to:
Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 - 64bit
Production
With the Partitioning, OLAP, Advanced Analytics and Real Application
Testing options
SQL> Database closed.
Database dismounted.
ORACLE instance shut down.
SQL> Disconnected from Oracle Database 12c Enterprise Edition Release
12.1.0.2.0 - 64bit Production
With the Partitioning, OLAP, Advanced Analytics and Real Application
Testing options
NTAP shut down

```

2. Konvertieren Sie die Dateisysteme in schreibgeschützt. Dies ist mit einem Skript schneller möglich, wie in dargestellt "[Dateisystem in schreibgeschützt konvertieren](#)".

```

[root@host1 current]# ./mk.fs.readonly.pl /oradata
/oradata unmounted
/oradata mounted read-only
[root@host1 current]# ./mk.fs.readonly.pl /logs
/logs unmounted
/logs mounted read-only

```

3. Vergewissern Sie sich, dass die Dateisysteme jetzt schreibgeschützt sind.

```

ontap-nfs1:/host1_oradata on /oradata type nfs
(ro,bg,vers=3,rsz=65536,wsz=65536,addr=172.20.101.10)
ontap-nfs1:/host1_logs on /logs type nfs
(ro,bg,vers=3,rsz=65536,wsz=65536,addr=172.20.101.10)

```

4. Synchronisieren Sie Dateisysteminhalte mit dem `rsync` Befehl.

```

[root@host1 current]# rsync -rlpogt --stats --progress
--exclude=.snapshot /oradata/ /new/oradata/
sending incremental file list
./
NTAP/
NTAP/IOPS.dbf

```

```

10737426432 100% 153.50MB/s 0:01:06 (xfer#1, to-check=10/13)
NTAP/iops.dbf.zip
    22823573 100% 12.09MB/s 0:00:01 (xfer#2, to-check=9/13)
...
NTAP/undotbs02.dbf
    1073750016 100% 131.60MB/s 0:00:07 (xfer#10, to-check=1/13)
NTAP/users01.dbf
    5251072 100% 3.95MB/s 0:00:01 (xfer#11, to-check=0/13)
Number of files: 13
Number of files transferred: 11
Total file size: 18570092218 bytes
Total transferred file size: 18570092218 bytes
Literal data: 18570092218 bytes
Matched data: 0 bytes
File list size: 277
File list generation time: 0.001 seconds
File list transfer time: 0.000 seconds
Total bytes sent: 18572359828
Total bytes received: 228
sent 18572359828 bytes received 228 bytes 162204017.96 bytes/sec
total size is 18570092218 speedup is 1.00
[root@host1 current]# rsync -rlpogt --stats --progress
--exclude=.snapshot /logs/ /new/logs/
sending incremental file list
./
NTAP/
NTAP/1_22_897068759.dbf
    45523968 100% 95.98MB/s 0:00:00 (xfer#1, to-check=15/18)
NTAP/1_23_897068759.dbf
    40601088 100% 49.45MB/s 0:00:00 (xfer#2, to-check=14/18)
...
NTAP/redo/redo02.log
    52429312 100% 44.68MB/s 0:00:01 (xfer#12, to-check=1/18)
NTAP/redo/redo03.log
    52429312 100% 68.03MB/s 0:00:00 (xfer#13, to-check=0/18)
Number of files: 18
Number of files transferred: 13
Total file size: 527032832 bytes
Total transferred file size: 527032832 bytes
Literal data: 527032832 bytes
Matched data: 0 bytes
File list size: 413
File list generation time: 0.001 seconds
File list transfer time: 0.000 seconds
Total bytes sent: 527098156
Total bytes received: 278

```

```
sent 527098156 bytes  received 278 bytes  95836078.91 bytes/sec
total size is 527032832  speedup is 1.00
```

5. Heben Sie die Bereitstellung der alten Dateisysteme auf, und verschieben Sie die kopierten Daten. Dies ist mit einem Skript schneller möglich, wie in dargestellt "[Ersetzen Sie Das Dateisystem](#)".

```
[root@host1 current]# ./swap.fs.pl /logs,/new/logs
/new/logs unmounted
/logs unmounted
Updated /logs mounted
[root@host1 current]# ./swap.fs.pl /oradata,/new/oradata
/new/oradata unmounted
/oradata unmounted
Updated /oradata mounted
```

6. Vergewissern Sie sich, dass die neuen Dateisysteme in der Position sind.

```
ontap-nfs1:/host1_logs_new on /logs type nfs
(rw,bg,vers=3,rsz=65536,wsz=65536,addr=172.20.101.10)
ontap-nfs1:/host1_oradata_new on /oradata type nfs
(rw,bg,vers=3,rsz=65536,wsz=65536,addr=172.20.101.10)
```

7. Starten Sie die Datenbank.

```
[root@host1 current]# ./dbstart.pl NTAP
ORACLE_SID = [oracle] ? The Oracle base has been set to /orabin
SQL*Plus: Release 12.1.0.2.0 Production on Thu Dec 3 16:10:07 2015
Copyright (c) 1982, 2014, Oracle. All rights reserved.
Connected to an idle instance.
SQL> ORACLE instance started.
Total System Global Area 805306368 bytes
Fixed Size 2929552 bytes
Variable Size 390073456 bytes
Database Buffers 406847488 bytes
Redo Buffers 5455872 bytes
Database mounted.
Database opened.
SQL> Disconnected from Oracle Database 12c Enterprise Edition Release
12.1.0.2.0 - 64bit Production
With the Partitioning, OLAP, Advanced Analytics and Real Application
Testing options
NTAP started
```

Vollständig automatisierte Umstellung

Dieses Beispielskript akzeptiert Argumente der Datenbank-SID gefolgt von gemeinsam getrennten Paaren von Dateisystemen. Für das oben abgebildete Beispiel wird der Befehl wie folgt ausgegeben:

```
[root@host1 current]# ./migrate.oracle.fs.pl NTAP /logs,/new/logs  
/oradata,/new/oradata
```

Wenn das Beispielskript ausgeführt wird, wird die folgende Sequenz ausgeführt. Er wird beendet, wenn in einem beliebigen Schritt ein Fehler auftritt:

1. Fahren Sie die Datenbank herunter.
2. Konvertieren Sie die aktuellen Dateisysteme in den schreibgeschützten Status.
3. Verwenden Sie jedes durch Kommas getrennte Paar von Dateisystemargumenten, und synchronisieren Sie das erste Dateisystem mit dem zweiten.
4. Entfernen Sie die früheren Dateisysteme.
5. Aktualisieren Sie die `/etc/fstab` Datei wie folgt:
 - a. Erstellen Sie ein Backup bei `/etc/fstab.bak`.
 - b. Kommentieren Sie die vorherigen Einträge für die vorherigen und neuen Dateisysteme.
 - c. Erstellen Sie einen neuen Eintrag für das neue Dateisystem, das den alten Bereitstellungspunkt verwendet.
6. Mounten Sie die Dateisysteme.
7. Starten Sie die Datenbank.

Der folgende Text enthält ein Ausführungsbeispiel für dieses Skript:

```
[root@host1 current]# ./migrate.oracle.fs.pl NTAP /logs,/new/logs  
/oradata,/new/oradata  
ORACLE_SID = [oracle] ? The Oracle base has been set to /orabin  
SQL*Plus: Release 12.1.0.2.0 Production on Thu Dec 3 17:05:50 2015  
Copyright (c) 1982, 2014, Oracle. All rights reserved.  
Connected to:  
Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 - 64bit  
Production  
With the Partitioning, OLAP, Advanced Analytics and Real Application  
Testing options  
SQL> Database closed.  
Database dismounted.  
ORACLE instance shut down.  
SQL> Disconnected from Oracle Database 12c Enterprise Edition Release  
12.1.0.2.0 - 64bit Production  
With the Partitioning, OLAP, Advanced Analytics and Real Application  
Testing options  
NTAP shut down
```

```

sending incremental file list
./
NTAP/
NTAP/1_22_897068759.dbf
    45523968 100% 185.40MB/s    0:00:00 (xfer#1, to-check=15/18)
NTAP/1_23_897068759.dbf
    40601088 100%  81.34MB/s    0:00:00 (xfer#2, to-check=14/18)
...
NTAP/redo/redo02.log
    52429312 100%  70.42MB/s    0:00:00 (xfer#12, to-check=1/18)
NTAP/redo/redo03.log
    52429312 100%  47.08MB/s    0:00:01 (xfer#13, to-check=0/18)
Number of files: 18
Number of files transferred: 13
Total file size: 527032832 bytes
Total transferred file size: 527032832 bytes
Literal data: 527032832 bytes
Matched data: 0 bytes
File list size: 413
File list generation time: 0.001 seconds
File list transfer time: 0.000 seconds
Total bytes sent: 527098156
Total bytes received: 278
sent 527098156 bytes  received 278 bytes  150599552.57 bytes/sec
total size is 527032832  speedup is 1.00
Succesfully replicated filesystem /logs to /new/logs
sending incremental file list
./
NTAP/
NTAP/IOPS.dbf
    10737426432 100% 176.55MB/s    0:00:58 (xfer#1, to-check=10/13)
NTAP/iops.dbf.zip
    22823573 100%   9.48MB/s    0:00:02 (xfer#2, to-check=9/13)
... NTAP/undotbs01.dbf
    309338112 100%  70.76MB/s    0:00:04 (xfer#9, to-check=2/13)
NTAP/undotbs02.dbf
    1073750016 100% 187.65MB/s    0:00:05 (xfer#10, to-check=1/13)
NTAP/users01.dbf
    5251072 100%   5.09MB/s    0:00:00 (xfer#11, to-check=0/13)
Number of files: 13
Number of files transferred: 11
Total file size: 18570092218 bytes
Total transferred file size: 18570092218 bytes
Literal data: 18570092218 bytes
Matched data: 0 bytes
File list size: 277

```

```

File list generation time: 0.001 seconds
File list transfer time: 0.000 seconds
Total bytes sent: 18572359828
Total bytes received: 228
sent 18572359828 bytes received 228 bytes 177725933.55 bytes/sec
total size is 18570092218 speedup is 1.00
Succesfully replicated filesystem /oradata to /new/oradata
swap 0 /logs /new/logs
/new/logs unmounted
/logs unmounted
Mounted updated /logs
Swapped filesystem /logs for /new/logs
swap 1 /oradata /new/oradata
/new/oradata unmounted
/oradata unmounted
Mounted updated /oradata
Swapped filesystem /oradata for /new/oradata
ORACLE_SID = [oracle] ? The Oracle base has been set to /orabin
SQL*Plus: Release 12.1.0.2.0 Production on Thu Dec 3 17:08:59 2015
Copyright (c) 1982, 2014, Oracle. All rights reserved.
Connected to an idle instance.
SQL> ORACLE instance started.
Total System Global Area 805306368 bytes
Fixed Size 2929552 bytes
Variable Size 390073456 bytes
Database Buffers 406847488 bytes
Redo Buffers 5455872 bytes
Database mounted.
Database opened.
SQL> Disconnected from Oracle Database 12c Enterprise Edition Release
12.1.0.2.0 - 64bit Production
With the Partitioning, OLAP, Advanced Analytics and Real Application
Testing options
NTAP started
[root@host1 current]#

```

Oracle ASM SPFile- und Passthwd-Migration

Eine Schwierigkeit beim Abschluss der ASM-Migration sind die ASM-spezifische SPFile- und die Passwort-Datei. Standardmäßig werden diese kritischen Metadatenfiles auf der ersten definierten ASM-Laufwerksgruppe erstellt. Wenn eine bestimmte ASM-Datenträgergruppe evakuiert und entfernt werden muss, müssen die SPFile- und Passwortdatei, die diese ASM-Instanz regelt, verschoben werden.

Ein weiterer Anwendungsfall, in dem diese Dateien eventuell verschoben werden müssen, ist die Implementierung von Datenbankmanagement-Software wie beispielsweise SnapManager für Oracle oder dem SnapCenter Oracle Plug-in. Eine der Funktionen dieser Produkte besteht darin, eine Datenbank schnell wiederherzustellen, indem der Zustand der ASM-LUNs, die die Datendateien hosten, zurückgesetzt wird. Um

dies zu tun, muss die ASM-Laufwerksgruppe offline geschaltet werden, bevor eine Wiederherstellung durchgeführt werden kann. Dies ist kein Problem, solange die Datendateien einer Datenbank in einer dedizierten ASM-Datenträgergruppe isoliert sind.

Wenn diese Datenträgergruppe auch die ASM-Datei spfile/passwd enthält, kann die Datenträgergruppe nur offline geschaltet werden, wenn die gesamte ASM-Instanz heruntergefahren wird. Dies ist ein disruptiver Prozess, was bedeutet, dass die Datei spfile/passwd verschoben werden muss.

Umgebung

1. Datenbank-SID = TOAST
2. Aktuelle Datendateien auf +DATA
3. Aktuelle Logfiles und Controlfiles auf +LOGS
4. Neue ASM-Laufwerksgruppen als eingerichtet +NEWDATA Und +NEWLOGS

Speicherorte für ASM-SPfile/passwd-Dateien

Die Verlagerung dieser Dateien kann ohne Unterbrechungen erfolgen. Aus Sicherheitsgründen empfiehlt NetApp jedoch, die Datenbankumgebung herunterzufahren, damit Sie sicher sein können, dass die Dateien verschoben wurden und die Konfiguration ordnungsgemäß aktualisiert wird. Dieses Verfahren muss wiederholt werden, wenn mehrere ASM-Instanzen auf einem Server vorhanden sind.

Ermitteln Sie ASM-Instanzen

Ermitteln Sie die ASM-Instanzen anhand der in aufgezeichneten Daten oratab Datei: Die ASM-Instanzen werden durch ein +-Symbol gekennzeichnet.

```
-bash-4.1$ cat /etc/oratab | grep '^+'
+ASM:/orabin/grid:N          # line added by Agent
```

Auf diesem Server befindet sich eine ASM-Instanz namens +ASM.

Stellen Sie sicher, dass alle Datenbanken heruntergefahren werden

Der einzige sichtbare smon-Prozess sollte der sman für die verwendete ASM-Instanz sein. Ein weiterer smon-Prozess zeigt an, dass eine Datenbank noch läuft.

```
-bash-4.1$ ps -ef | grep smon
oracle      857      1  0 18:26 ?        00:00:00 asm_smon_+ASM
```

Der einzige smon-Prozess ist die ASM-Instanz selbst. Das bedeutet, dass keine anderen Datenbanken ausgeführt werden und ohne das Risiko einer Störung der Datenbankvorgänge sicher fortgesetzt werden kann.

Suchen Sie Dateien

Ermitteln Sie den aktuellen Speicherort der ASM-Datei und der Passwortdatei mithilfe des spget Und pwget Befehle.

```
bash-4.1$ asmcmd
ASMCMDB> spget
+DATA/spfile.ora
```

```
ASMCMDB> pwget --asm
+DATA/orapwasm
```

Beide Dateien befinden sich an der Basis des +DATA Festplattengruppe.

Dateien kopieren

Kopieren Sie die Dateien mit dem in die neue ASM-Datenträgergruppe `spcopy` Und `pwcopy` Befehle. Wenn die neue Laufwerksgruppe vor kurzem erstellt wurde und derzeit leer ist, muss sie möglicherweise zuerst gemountet werden.

```
ASMCMDB> mount NEWDATA
```

```
ASMCMDB> spcopy +DATA/spfile.ora +NEWDATA/spfile.ora
copying +DATA/spfile.ora -> +NEWDATA/spfilea.ora
```

```
ASMCMDB> pwcopy +DATA/orapwasm +NEWDATA/orapwasm
copying +DATA/orapwasm -> +NEWDATA/orapwasm
```

Die Dateien wurden nun von kopiert +DATA Bis +NEWDATA.

ASM-Instanz aktualisieren

Die ASM-Instanz muss jetzt aktualisiert werden, um die Standortänderung widerzuspiegeln. Der `spset` Und `pwset` Befehle aktualisieren die zum Starten der ASM-Datenträgergruppe erforderlichen ASM-Metadaten.

```
ASMCMDB> spset +NEWDATA/spfile.ora
ASMCMDB> pwset --asm +NEWDATA/orapwasm
```

Aktivieren Sie ASM mit aktualisierten Dateien

Zu diesem Zeitpunkt verwendet die ASM-Instanz weiterhin die früheren Speicherorte dieser Dateien. Die Instanz muss neu gestartet werden, um ein erneutes Lesen der Dateien von ihren neuen Speicherorten zu erzwingen und Sperren für die vorherigen Dateien freizugeben.

```
-bash-4.1$ sqlplus / as sysasm
SQL> shutdown immediate;
ASM diskgroups volume disabled
ASM diskgroups dismounted
ASM instance shutdown
```

```
SQL> startup
ASM instance started
Total System Global Area 1140850688 bytes
Fixed Size                2933400 bytes
Variable Size             1112751464 bytes
ASM Cache                  25165824 bytes
ORA-15032: not all alterations performed
ORA-15017: diskgroup "NEWDATA" cannot be mounted
ORA-15013: diskgroup "NEWDATA" is already mounted
```

Entfernen Sie alte spfile- und Passwortdateien

Wenn der Vorgang erfolgreich durchgeführt wurde, sind die vorherigen Dateien nicht mehr gesperrt und können jetzt entfernt werden.

```
-bash-4.1$ asmcmd
ASMCMD> rm +DATA/spfile.ora
ASMCMD> rm +DATA/orapwasm
```

Kopie von Oracle ASM zu ASM

Oracle ASM ist im Grunde ein schlankes kombiniertes Volume-Manager- und Dateisystem. Da das Dateisystem nicht sofort sichtbar ist, muss RMAN für Kopiervorgänge verwendet werden. Ein auf Kopien basierender Migrationsprozess ist zwar sicher und einfach, kann jedoch mit Unterbrechungen verbunden sein. Die Unterbrechung kann minimiert, aber nicht vollständig beseitigt werden.

Wenn Sie eine unterbrechungsfreie Migration einer ASM-basierten Datenbank wünschen, empfiehlt es sich, die ASM-Fähigkeit zu nutzen, um ASM-Extents auf neue LUNs auszugleichen, während die alten LUNs gelöscht werden. Dies ist im Allgemeinen sicher und unterbrechungsfrei, bietet aber keinen Ausweg. Wenn Funktions- oder Leistungsprobleme auftreten, besteht die einzige Möglichkeit darin, die Daten zurück zur Quelle zu migrieren.

Dieses Risiko kann vermieden werden, indem die Datenbank an den neuen Speicherort kopiert wird, anstatt Daten zu verschieben, sodass die Originaldaten nicht geändert werden. Die Datenbank kann vor der Inbetriebnahme vollständig an ihrem neuen Standort getestet werden, und die ursprüngliche Datenbank steht als Fallback-Option zur Verfügung, wenn Probleme gefunden werden.

Dieses Verfahren ist eine von vielen Optionen, die RMAN einbeziehen. Er ermöglicht einen zweistufigen Prozess, bei dem das erste Backup erstellt und später durch die Protokollwiedergabe synchronisiert wird. Dieser Prozess sollte die Downtime minimieren, da die Datenbank betriebsbereit bleibt und während der

ersten Basiskopie Daten bereitgestellt werden können.

Datenbank kopieren

Oracle RMAN erstellt eine vollständige Kopie der Quelldatenbank der Ebene 0, die sich derzeit in der ASM-Datenträgergruppe befindet +DATA An den neuen Standort am +NEWDATA.

```
-bash-4.1$ rman target /
Recovery Manager: Release 12.1.0.2.0 - Production on Sun Dec 6 17:40:03
2015
Copyright (c) 1982, 2014, Oracle and/or its affiliates. All rights
reserved.
connected to target database: TOAST (DBID=2084313411)
RMAN> backup as copy incremental level 0 database format '+NEWDATA' tag
'ONTAP_MIGRATION';
Starting backup at 06-DEC-15
using target database control file instead of recovery catalog
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=302 device type=DISK
channel ORA_DISK_1: starting datafile copy
input datafile file number=00001
name=+DATA/TOAST/DATAFILE/system.262.897683141
...
input datafile file number=00004
name=+DATA/TOAST/DATAFILE/users.264.897683151
output file name=+NEWDATA/TOAST/DATAFILE/users.258.897759623
tag=ONTAP_MIGRATION RECID=5 STAMP=897759622
channel ORA_DISK_1: datafile copy complete, elapsed time: 00:00:01
channel ORA_DISK_1: starting incremental level 0 datafile backup set
channel ORA_DISK_1: specifying datafile(s) in backup set
including current SPFILE in backup set
channel ORA_DISK_1: starting piece 1 at 06-DEC-15
channel ORA_DISK_1: finished piece 1 at 06-DEC-15
piece
handle=+NEWDATA/TOAST/BACKUPSET/2015_12_06/nnsnn0_ontap_migration_0.262.89
7759623 tag=ONTAP_MIGRATION comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:01
Finished backup at 06-DEC-15
```

Schalter für Archivprotokoll erzwingen

Sie müssen einen Schalter für das Archivprotokoll erzwingen, um sicherzustellen, dass die Archivprotokolle alle Daten enthalten, die erforderlich sind, um die Kopie vollständig konsistent zu machen. Ohne diesen Befehl können Schlüsseldaten in den Wiederherstellungsprotokollen weiterhin vorhanden sein.

```
RMAN> sql 'alter system archive log current';  
sql statement: alter system archive log current
```

Quelldatenbank herunterfahren

Die Unterbrechung beginnt in diesem Schritt, weil die Datenbank heruntergefahren und in einen schreibgeschützten Modus mit eingeschränktem Zugriff versetzt wird. Um die Quelldatenbank herunterzufahren, führen Sie die folgenden Befehle aus:

```
RMAN> shutdown immediate;  
using target database control file instead of recovery catalog  
database closed  
database dismounted  
Oracle instance shut down  
RMAN> startup mount;  
connected to target database (not started)  
Oracle instance started  
database mounted  
Total System Global Area      805306368 bytes  
Fixed Size                    2929552 bytes  
Variable Size                 390073456 bytes  
Database Buffers              406847488 bytes  
Redo Buffers                   5455872 bytes
```

Backup von Controlfile

Sie müssen die controlfile sichern, falls Sie die Migration abbrechen und zum ursprünglichen Speicherort zurückkehren müssen. Eine Kopie der Backup-Steuerdatei ist nicht 100% erforderlich, aber es macht den Prozess des Rücksetzens der Datenbank-Speicherorte zurück an den ursprünglichen Speicherort einfacher.

```
RMAN> backup as copy current controlfile format '/tmp/TOAST.ctrl';  
Starting backup at 06-DEC-15  
allocated channel: ORA_DISK_1  
channel ORA_DISK_1: SID=358 device type=DISK  
channel ORA_DISK_1: starting datafile copy  
copying current control file  
output file name=/tmp/TOAST.ctrl tag=TAG20151206T174753 RECID=6  
STAMP=897760073  
channel ORA_DISK_1: datafile copy complete, elapsed time: 00:00:01  
Finished backup at 06-DEC-15
```

Parameteraktualisierungen

Der aktuelle spfile enthält Verweise auf die Steuerdateien an ihren aktuellen Speicherorten innerhalb der alten ASM-Datenträgergruppe. Es muss bearbeitet werden, was leicht durch das Bearbeiten einer Zwischenversion

von pfile erfolgt.

```
RMAN> create pfile='/tmp/pfile' from spfile;  
Statement processed
```

Aktualisieren Sie pfile

Aktualisieren Sie alle Parameter, die sich auf alte ASM-Datenträgergruppen beziehen, um die neuen Namen der ASM-Datenträgergruppen wiederzugeben. Speichern Sie dann die aktualisierte Datei pfile. Stellen Sie sicher, dass die `db_create` Parameter sind vorhanden.

Im folgenden Beispiel werden die Verweise auf angezeigt `+DATA` Die in geändert wurden `+NEWDATA` Sind gelb markiert. Zwei wichtige Parameter sind die `db_create` Parameter, die neue Dateien am richtigen Speicherort erstellen.

```
*.compatible='12.1.0.2.0'  
*.control_files='+NEWLOGS/TOAST/CONTROLFILE/current.258.897683139'  
*.db_block_size=8192  
*. db_create_file_dest='+NEWDATA'  
*. db_create_online_log_dest_1='+NEWLOGS'  
*.db_domain=''   
*.db_name='TOAST'  
*.diagnostic_dest='/orabin'  
*.dispatchers='(PROTOCOL=TCP) (SERVICE=TOASTXDB) '  
*.log_archive_dest_1='LOCATION='+NEWLOGS'  
*.log_archive_format='%t_%s_%r.dbf'
```

Init.ora-Datei aktualisieren

Die meisten ASM-basierten Datenbanken verwenden einen `init.ora` Datei befindet sich im `$ORACLE_HOME/dbs` Verzeichnis, das einen Punkt auf das Spfile auf der ASM-Datenträgergruppe darstellt. Diese Datei muss an einen Speicherort auf der neuen ASM-Datenträgergruppe umgeleitet werden.

```
-bash-4.1$ cd $ORACLE_HOME/dbs  
-bash-4.1$ cat initTOAST.ora  
SPFILE='+DATA/TOAST/spfileTOAST.ora'
```

Ändern Sie diese Datei wie folgt:

```
SPFILE='+NEWLOGS/TOAST/spfileTOAST.ora
```

Wiederherstellung der Parameterdatei

Der spfile kann nun mit den Daten in der bearbeiteten pfile gefüllt werden.

```
RMAN> create spfile from pfile='/tmp/pfile';  
Statement processed
```

Starten Sie die Datenbank, um neue spfile zu verwenden

Starten Sie die Datenbank, um sicherzustellen, dass sie jetzt den neu erstellten spfile verwendet und dass alle weiteren Änderungen an den Systemparametern korrekt aufgezeichnet werden.

```
RMAN> startup nomount;  
connected to target database (not started)  
Oracle instance started  
Total System Global Area      805306368 bytes  
Fixed Size                    2929552 bytes  
Variable Size                 373296240 bytes  
Database Buffers              423624704 bytes  
Redo Buffers                   5455872 bytes
```

Kontrolldatei wiederherstellen

Die von RMAN erstellte Backup-Kontrolldatei kann auch direkt an dem im neuen spfile angegebenen Speicherort wiederhergestellt werden.

```
RMAN> restore controlfile from  
'+DATA/TOAST/CONTROLFILE/current.258.897683139';  
Starting restore at 06-DEC-15  
using target database control file instead of recovery catalog  
allocated channel: ORA_DISK_1  
channel ORA_DISK_1: SID=417 device type=DISK  
channel ORA_DISK_1: copied control file copy  
output file name=+NEWLOGS/TOAST/CONTROLFILE/current.273.897761061  
Finished restore at 06-DEC-15
```

Mounten Sie die Datenbank und überprüfen Sie die Verwendung der neuen Steuerdatei.

```
RMAN> alter database mount;  
using target database control file instead of recovery catalog  
Statement processed
```

```
SQL> show parameter control_files;
NAME                                TYPE                                VALUE
-----                                -
control_files                        string
+NEWLOGS/TOAST/CONTROLFILE/cur
                                         rent.273.897761061
```

Protokollwiedergabe

Die Datenbank verwendet derzeit die Datendateien am alten Speicherort. Bevor die Kopie verwendet werden kann, müssen sie synchronisiert werden. Die Zeit während des ersten Kopiervorgangs ist verstrichen, und die Änderungen wurden hauptsächlich in den Archivprotokollen protokolliert. Diese Änderungen werden wie folgt repliziert:

1. Führen Sie ein inkrementelles RMAN-Backup durch, das die Archivprotokolle enthält.

```

RMAN> backup incremental level 1 format '+NEWLOGS' for recover of copy
with tag 'ONTAP_MIGRATION' database;
Starting backup at 06-DEC-15
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=62 device type=DISK
channel ORA_DISK_1: starting incremental level 1 datafile backup set
channel ORA_DISK_1: specifying datafile(s) in backup set
input datafile file number=00001
name=+DATA/TOAST/DATAFILE/system.262.897683141
input datafile file number=00002
name=+DATA/TOAST/DATAFILE/sysaux.260.897683143
input datafile file number=00003
name=+DATA/TOAST/DATAFILE/undotbs1.257.897683145
input datafile file number=00004
name=+DATA/TOAST/DATAFILE/users.264.897683151
channel ORA_DISK_1: starting piece 1 at 06-DEC-15
channel ORA_DISK_1: finished piece 1 at 06-DEC-15
piece
handle=+NEWLOGS/TOAST/BACKUPSET/2015_12_06/nnndn1_ontap_migration_0.268.
897762693 tag=ONTAP_MIGRATION comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:01
channel ORA_DISK_1: starting incremental level 1 datafile backup set
channel ORA_DISK_1: specifying datafile(s) in backup set
including current control file in backup set
including current SPFILE in backup set
channel ORA_DISK_1: starting piece 1 at 06-DEC-15
channel ORA_DISK_1: finished piece 1 at 06-DEC-15
piece
handle=+NEWLOGS/TOAST/BACKUPSET/2015_12_06/ncsnn1_ontap_migration_0.267.
897762697 tag=ONTAP_MIGRATION comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:01
Finished backup at 06-DEC-15

```

2. Wiederholen Sie das Protokoll.

```

RMAN> recover copy of database with tag 'ONTAP_MIGRATION';
Starting recover at 06-DEC-15
using channel ORA_DISK_1
channel ORA_DISK_1: starting incremental datafile backup set restore
channel ORA_DISK_1: specifying datafile copies to recover
recovering datafile copy file number=00001
name=+NEWDATA/TOAST/DATAFILE/system.259.897759609
recovering datafile copy file number=00002
name=+NEWDATA/TOAST/DATAFILE/sysaux.263.897759615
recovering datafile copy file number=00003
name=+NEWDATA/TOAST/DATAFILE/undotbs1.264.897759619
recovering datafile copy file number=00004
name=+NEWDATA/TOAST/DATAFILE/users.258.897759623
channel ORA_DISK_1: reading from backup piece
+NEWLOGS/TOAST/BACKUPSET/2015_12_06/nnndn1_ontap_migration_0.268.8977626
93
channel ORA_DISK_1: piece
handle=+NEWLOGS/TOAST/BACKUPSET/2015_12_06/nnndn1_ontap_migration_0.268.
897762693 tag=ONTAP_MIGRATION
channel ORA_DISK_1: restored backup piece 1
channel ORA_DISK_1: restore complete, elapsed time: 00:00:01
Finished recover at 06-DEC-15

```

Aktivierung

Die wiederhergestellte Steuerdatei verweist weiterhin auf die Datendateien am ursprünglichen Speicherort und enthält auch die Pfadinformationen für die kopierten Datendateien.

1. Um die aktiven Datendateien zu ändern, führen Sie den `switch database to copy` Befehl.

```

RMAN> switch database to copy;
datafile 1 switched to datafile copy
"+NEWDATA/TOAST/DATAFILE/system.259.897759609"
datafile 2 switched to datafile copy
"+NEWDATA/TOAST/DATAFILE/sysaux.263.897759615"
datafile 3 switched to datafile copy
"+NEWDATA/TOAST/DATAFILE/undotbs1.264.897759619"
datafile 4 switched to datafile copy
"+NEWDATA/TOAST/DATAFILE/users.258.897759623"

```

Die aktiven Datendateien sind nun die kopierten Datendateien, aber es können immer noch Änderungen in den letzten Redo-Protokollen enthalten sein.

2. Um alle verbleibenden Protokolle wiederzugeben, führen Sie den `recover database` Befehl. Wenn die Meldung angezeigt wird `media recovery complete` Wird angezeigt, der Prozess war erfolgreich.

```

RMAN> recover database;
Starting recover at 06-DEC-15
using channel ORA_DISK_1
starting media recovery
media recovery complete, elapsed time: 00:00:01
Finished recover at 06-DEC-15

```

Bei diesem Vorgang wurde nur der Speicherort der normalen Datendateien geändert. Die temporären Datendateien müssen umbenannt werden, müssen aber nicht kopiert werden, da sie nur temporär sind. Die Datenbank ist derzeit nicht verfügbar, sodass es keine aktiven Daten in den temporären Datendateien gibt.

- Um die temporären Datendateien zu verschieben, geben Sie zuerst ihren Speicherort an.

```

RMAN> select file#||' '||name from v$tempfile;
FILE#||' '||NAME
-----
-----
1 +DATA/TOAST/TEMPFILE/temp.263.897683145

```

- Verschieben Sie temporäre Datendateien mithilfe eines RMAN-Befehls, der den neuen Namen für jede Datendatei festlegt. Bei Oracle Managed Files (OMF) ist der vollständige Name nicht erforderlich; die ASM-Datenträgergruppe reicht aus. Wenn die Datenbank geöffnet wird, verknüpft OMF mit dem entsprechenden Speicherort in der ASM-Datenträgergruppe. Um Dateien zu verschieben, führen Sie die folgenden Befehle aus:

```

run {
set newname for tempfile 1 to '+NEWDATA';
switch tempfile all;
}

```

```

RMAN> run {
2> set newname for tempfile 1 to '+NEWDATA';
3> switch tempfile all;
4> }
executing command: SET NEWNAME
renamed tempfile 1 to +NEWDATA in control file

```

Migration des Wiederherstellungsprotokolls

Der Migrationsprozess ist fast abgeschlossen, aber die Wiederherstellungsprotokolle befinden sich immer noch in der ursprünglichen ASM-Laufwerksgruppe. Wiederherstellungsprotokolle können nicht direkt verschoben werden. Stattdessen wird ein neuer Satz von Wiederherstellungsprotokollen erstellt und der Konfiguration hinzugefügt, gefolgt von einem Drop der alten Protokolle.

1. Ermitteln Sie die Anzahl der Redo-Log-Gruppen und deren jeweilige Gruppennummern.

```

RMAN> select group#||' '||member from v$logfile;
GROUP#||' '||MEMBER
-----
-----
1 +DATA/TOAST/ONLINELOG/group_1.261.897683139
2 +DATA/TOAST/ONLINELOG/group_2.259.897683139
3 +DATA/TOAST/ONLINELOG/group_3.256.897683139

```

2. Geben Sie die Größe der Wiederherstellungsprotokolle ein.

```

RMAN> select group#||' '||bytes from v$log;
GROUP#||' '||BYTES
-----
-----
1 52428800
2 52428800
3 52428800

```

3. Erstellen Sie für jedes Redo-Protokoll eine neue Gruppe mit einer passenden Konfiguration. Wenn Sie OMF nicht verwenden, müssen Sie den vollständigen Pfad angeben. Dies ist auch ein Beispiel, das den verwendet `db_create_online_log` Parameter. Wie bereits gezeigt, wurde dieser Parameter auf `+NEWLOGS` gesetzt. Mit dieser Konfiguration können Sie die folgenden Befehle verwenden, um neue Online-Protokolle zu erstellen, ohne einen Dateispeicherort oder sogar eine bestimmte ASM-Datenträgergruppe angeben zu müssen.

```

RMAN> alter database add logfile size 52428800;
Statement processed
RMAN> alter database add logfile size 52428800;
Statement processed
RMAN> alter database add logfile size 52428800;
Statement processed

```

4. Öffnen Sie die Datenbank.

```

SQL> alter database open;
Database altered.

```

5. Die alten Protokolle ablegen.

```
RMAN> alter database drop logfile group 1;
Statement processed
```

6. Wenn ein Fehler auftritt, der verhindert, dass Sie ein aktives Protokoll ablegen, erzwingen Sie einen Wechsel zum nächsten Protokoll, um die Sperre freizugeben und einen globalen Kontrollpunkt zu erzwingen. Ein Beispiel ist unten dargestellt. Der Versuch, die Logfile-Gruppe 3, die sich am alten Speicherort befand, zu löschen, wurde abgelehnt, da noch aktive Daten in dieser Logdatei vorhanden waren. Eine Protokollarchivierung nach einem Kontrollpunkt ermöglicht das Löschen der Protokolldatei.

```
RMAN> alter database drop logfile group 3;
RMAN-00571: =====
RMAN-00569: ===== ERROR MESSAGE STACK FOLLOWS =====
RMAN-00571: =====
RMAN-03002: failure of sql statement command at 12/08/2015 20:23:51
ORA-01623: log 3 is current log for instance TOAST (thread 4) - cannot
drop
ORA-00312: online log 3 thread 1:
'+LOGS/TOAST/ONLINELOG/group_3.259.897563549'
RMAN> alter system switch logfile;
Statement processed
RMAN> alter system checkpoint;
Statement processed
RMAN> alter database drop logfile group 3;
Statement processed
```

7. Überprüfen Sie die Umgebung, um sicherzustellen, dass alle standortbasierten Parameter aktualisiert werden.

```
SQL> select name from v$datafile;
SQL> select member from v$logfile;
SQL> select name from v$tempfile;
SQL> show parameter spfile;
SQL> select name, value from v$parameter where value is not null;
```

8. Im folgenden Skript wird erläutert, wie dieser Prozess vereinfacht werden kann:

```

[root@host1 current]# ./checkdbdata.pl TOAST
TOAST datafiles:
+NEWDATA/TOAST/DATAFILE/system.259.897759609
+NEWDATA/TOAST/DATAFILE/sysaux.263.897759615
+NEWDATA/TOAST/DATAFILE/undotbs1.264.897759619
+NEWDATA/TOAST/DATAFILE/users.258.897759623
TOAST redo logs:
+NEWLOGS/TOAST/ONLINELOG/group_4.266.897763123
+NEWLOGS/TOAST/ONLINELOG/group_5.265.897763125
+NEWLOGS/TOAST/ONLINELOG/group_6.264.897763125
TOAST temp datafiles:
+NEWDATA/TOAST/TEMPFILE/temp.260.897763165
TOAST spfile
spfile                                string
+NEWDATA/spfiletoast.ora
TOAST key parameters
control_files +NEWLOGS/TOAST/CONTROLFILE/current.273.897761061
log_archive_dest_1 LOCATION=+NEWLOGS
db_create_file_dest +NEWDATA
db_create_online_log_dest_1 +NEWLOGS

```

9. Wenn die ASM-Datenträgergruppen vollständig evakuiert wurden, können sie jetzt mit abgehängt werden `asmcmd`. In vielen Fällen sind jedoch die Dateien, die zu anderen Datenbanken oder der ASM-Datei `spfile/passwd` gehören, noch vorhanden.

```

-bash-4.1$ . oraenv
ORACLE_SID = [TOAST] ? +ASM
The Oracle base remains unchanged with value /orabin
-bash-4.1$ asmcmd
ASMCMD> umount DATA
ASMCMD>

```

Kopie von Oracle ASM auf das Dateisystem

Das Verfahren zum Kopieren von Oracle ASM in ein Dateisystem ähnelt dem Verfahren zum Kopieren von ASM zu ASM mit ähnlichen Vorteilen und Einschränkungen. Der Hauptunterschied ist die Syntax der verschiedenen Befehle und Konfigurationsparameter bei der Verwendung eines sichtbaren Dateisystems im Gegensatz zu einer ASM-Datenträgergruppe.

Datenbank kopieren

Oracle RMAN wird verwendet, um eine (vollständige) Kopie der Quelldatenbank zu erstellen, die sich derzeit in der ASM-Datenträgergruppe befindet `+DATA` An den neuen Standort am `/oradata`.

```

RMAN> backup as copy incremental level 0 database format
'/oradata/TOAST/%U' tag 'ONTAP_MIGRATION';
Starting backup at 13-MAY-16
using target database control file instead of recovery catalog
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=377 device type=DISK
channel ORA_DISK_1: starting datafile copy
input datafile file number=00001 name=+ASM0/TOAST/system01.dbf
output file name=/oradata/TOAST/data_D-TOAST_I-2098173325_TS-SYSTEM_FNO-
1_01r5fhjg tag=ONTAP_MIGRATION RECID=1 STAMP=911722099
channel ORA_DISK_1: datafile copy complete, elapsed time: 00:00:07
channel ORA_DISK_1: starting datafile copy
input datafile file number=00002 name=+ASM0/TOAST/sysaux01.dbf
output file name=/oradata/TOAST/data_D-TOAST_I-2098173325_TS-SYSAUX_FNO-
2_02r5fhjo tag=ONTAP_MIGRATION RECID=2 STAMP=911722106
channel ORA_DISK_1: datafile copy complete, elapsed time: 00:00:07
channel ORA_DISK_1: starting datafile copy
input datafile file number=00003 name=+ASM0/TOAST/undotbs101.dbf
output file name=/oradata/TOAST/data_D-TOAST_I-2098173325_TS-UNDOTBS1_FNO-
3_03r5fhjt tag=ONTAP_MIGRATION RECID=3 STAMP=911722113
channel ORA_DISK_1: datafile copy complete, elapsed time: 00:00:07
channel ORA_DISK_1: starting datafile copy
copying current control file
output file name=/oradata/TOAST/cf_D-TOAST_id-2098173325_04r5fhk5
tag=ONTAP_MIGRATION RECID=4 STAMP=911722118
channel ORA_DISK_1: datafile copy complete, elapsed time: 00:00:01
channel ORA_DISK_1: starting datafile copy
input datafile file number=00004 name=+ASM0/TOAST/users01.dbf
output file name=/oradata/TOAST/data_D-TOAST_I-2098173325_TS-USERS_FNO-
4_05r5fhk6 tag=ONTAP_MIGRATION RECID=5 STAMP=911722118
channel ORA_DISK_1: datafile copy complete, elapsed time: 00:00:01
channel ORA_DISK_1: starting incremental level 0 datafile backup set
channel ORA_DISK_1: specifying datafile(s) in backup set
including current SPFILE in backup set
channel ORA_DISK_1: starting piece 1 at 13-MAY-16
channel ORA_DISK_1: finished piece 1 at 13-MAY-16
piece handle=/oradata/TOAST/06r5fhk7_1_1 tag=ONTAP_MIGRATION comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:01
Finished backup at 13-MAY-16

```

Schalter für Archivprotokoll erzwingen

Der Wechsel des Archivprotokolls muss erzwungen werden, um sicherzustellen, dass die Archivprotokolle alle erforderlichen Daten enthalten, damit die Kopie vollständig konsistent ist. Ohne diesen Befehl können Schlüsseldaten in den Wiederherstellungsprotokollen weiterhin vorhanden sein. Um einen Archivprotokollschalter zu erzwingen, führen Sie den folgenden Befehl aus:

```
RMAN> sql 'alter system archive log current';
sql statement: alter system archive log current
```

Quelldatenbank herunterfahren

Die Unterbrechung beginnt in diesem Schritt, weil die Datenbank heruntergefahren und in einen schreibgeschützten Modus mit eingeschränktem Zugriff versetzt wird. Um die Quelldatenbank herunterzufahren, führen Sie die folgenden Befehle aus:

```
RMAN> shutdown immediate;
using target database control file instead of recovery catalog
database closed
database dismounted
Oracle instance shut down
RMAN> startup mount;
connected to target database (not started)
Oracle instance started
database mounted
Total System Global Area      805306368 bytes
Fixed Size                    2929552 bytes
Variable Size                 331353200 bytes
Database Buffers              465567744 bytes
Redo Buffers                   5455872 bytes
```

Backup von Controlfile

Sichern Sie controlfiles, falls Sie die Migration abbrechen und zum ursprünglichen Speicherort zurückkehren müssen. Eine Kopie der Backup-Steuerdatei ist nicht 100% erforderlich, aber es macht den Prozess des Rücksetzens der Datenbank-Speicherorte zurück an den ursprünglichen Speicherort einfacher.

```
RMAN> backup as copy current controlfile format '/tmp/TOAST.ctrl';
Starting backup at 08-DEC-15
using channel ORA_DISK_1
channel ORA_DISK_1: starting datafile copy
copying current control file
output file name=/tmp/TOAST.ctrl tag=TAG20151208T194540 RECID=30
STAMP=897939940
channel ORA_DISK_1: datafile copy complete, elapsed time: 00:00:01
Finished backup at 08-DEC-15
```

Parameteraktualisierungen

```
RMAN> create pfile='/tmp/pfile' from spfile;
Statement processed
```

Aktualisieren Sie pfile

Alle Parameter, die sich auf alte ASM-Datenträgergruppen beziehen, sollten aktualisiert und in einigen Fällen gelöscht werden, wenn sie nicht mehr relevant sind. Aktualisieren Sie sie, um die neuen Dateisystempfade wiederzugeben, und speichern Sie die aktualisierte Datei pfile. Stellen Sie sicher, dass der vollständige Zielpfad aufgeführt ist. Um diese Parameter zu aktualisieren, führen Sie die folgenden Befehle aus:

```
*.audit_file_dest='/orabin/admin/TOAST/adump'
*.audit_trail='db'
*.compatible='12.1.0.2.0'
*.control_files='/logs/TOAST/arch/control01.ctl','/logs/TOAST/redo/control
02.ctl'
*.db_block_size=8192
*.db_domain=''
*.db_name='TOAST'
*.diagnostic_dest='/orabin'
*.dispatchers='(PROTOCOL=TCP) (SERVICE=TOASTXDB) '
*.log_archive_dest_1='LOCATION=/logs/TOAST/arch'
*.log_archive_format='%t_%s_%r.dbf'
*.open_cursors=300
*.pga_aggregate_target=256m
*.processes=300
*.remote_login_passwordfile='EXCLUSIVE'
*.sga_target=768m
*.undo_tablespace='UNDOTBS1'
```

Deaktivieren Sie die ursprüngliche init.ora-Datei

Diese Datei befindet sich im `$ORACLE_HOME/dbs` Verzeichnis und befindet sich in der Regel in einem pfile, das als Zeiger auf den spfile auf der ASM-Datenträgergruppe dient. Um sicherzustellen, dass der ursprüngliche Spfile nicht mehr verwendet wird, benennen Sie ihn um. Löschen Sie sie jedoch nicht, da diese Datei erforderlich ist, wenn die Migration abgebrochen werden muss.

```
[oracle@jfscl ~]$ cd $ORACLE_HOME/dbs
[oracle@jfscl dbs]$ cat initTOAST.ora
SPFILE='+ASM0/TOAST/spfileTOAST.ora'
[oracle@jfscl dbs]$ mv initTOAST.ora initTOAST.ora.prev
[oracle@jfscl dbs]$
```

Wiederherstellung der Parameterdatei

Dies ist der letzte Schritt bei der Verlagerung von Spfile. Der ursprüngliche spfile wird nicht mehr verwendet und die Datenbank wird derzeit mit der Zwischendatei gestartet (aber nicht gemountet). Der Inhalt dieser Datei kann wie folgt an den neuen Speicherort spfile geschrieben werden:

```
RMAN> create spfile from pfile='/tmp/pfile';  
Statement processed
```

Starten Sie die Datenbank, um neue spfile zu verwenden

Sie müssen die Datenbank starten, um die Sperren der Zwischendatei freizugeben und die Datenbank nur mit der neuen Datei spfile zu starten. Das Starten der Datenbank beweist auch, dass der neue spfile-Speicherort korrekt ist und seine Daten gültig sind.

```
RMAN> shutdown immediate;  
Oracle instance shut down  
RMAN> startup nomount;  
connected to target database (not started)  
Oracle instance started  
Total System Global Area      805306368 bytes  
Fixed Size                     2929552 bytes  
Variable Size                 331353200 bytes  
Database Buffers              465567744 bytes  
Redo Buffers                   5455872 bytes
```

Kontrolldatei wiederherstellen

Auf dem Pfad wurde eine Sicherungscontrolldatei erstellt /tmp/TOAST.ctrl Früher im Verfahren. Der neue spfile definiert die Speicherorte der controlfile als /logfs/TOAST/ctrl/ctrlfile1.ctrl Und /logfs/TOAST/redo/ctrlfile2.ctrl. Diese Dateien sind jedoch noch nicht vorhanden.

1. Mit diesem Befehl werden die controlfile-Daten auf den im spfile definierten Pfaden wiederhergestellt.

```
RMAN> restore controlfile from '/tmp/TOAST.ctrl';  
Starting restore at 13-MAY-16  
using channel ORA_DISK_1  
channel ORA_DISK_1: copied control file copy  
output file name=/logs/TOAST/arch/control01.ctl  
output file name=/logs/TOAST/redo/control02.ctl  
Finished restore at 13-MAY-16
```

2. Geben Sie den Mount-Befehl ein, damit die Steuerdateien korrekt erkannt werden und gültige Daten enthalten.

```
RMAN> alter database mount;
Statement processed
released channel: ORA_DISK_1
```

Um den zu validieren `control_files` Parameter, führen Sie den folgenden Befehl aus:

```
SQL> show parameter control_files;
NAME                                TYPE                                VALUE
-----                                -----                                -
control_files                        string                              /logs/TOAST/arch/control01ctl
                                        '
                                        /logs/TOAST/redo/control02c
                                        t1
```

Protokollwiedergabe

Die Datenbank verwendet derzeit die Datendateien am alten Speicherort. Bevor die Kopie verwendet werden kann, müssen die Datendateien synchronisiert werden. Die Zeit während des ersten Kopiervorgangs ist verstrichen, und Änderungen wurden hauptsächlich in den Archivprotokollen protokolliert. Diese Änderungen werden in den folgenden beiden Schritten repliziert.

1. Führen Sie ein inkrementelles RMAN-Backup durch, das die Archivprotokolle enthält.

```

RMAN> backup incremental level 1 format '/logs/TOAST/arch/%U' for
recover of copy with tag 'ONTAP_MIGRATION' database;
Starting backup at 13-MAY-16
using target database control file instead of recovery catalog
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=124 device type=DISK
channel ORA_DISK_1: starting incremental level 1 datafile backup set
channel ORA_DISK_1: specifying datafile(s) in backup set
input datafile file number=00001 name=+ASM0/TOAST/system01.dbf
input datafile file number=00002 name=+ASM0/TOAST/sysaux01.dbf
input datafile file number=00003 name=+ASM0/TOAST/undotbs101.dbf
input datafile file number=00004 name=+ASM0/TOAST/users01.dbf
channel ORA_DISK_1: starting piece 1 at 13-MAY-16
channel ORA_DISK_1: finished piece 1 at 13-MAY-16
piece handle=/logs/TOAST/arch/09r5fj8i_1_1 tag=ONTAP_MIGRATION
comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:01
Finished backup at 13-MAY-16
RMAN-06497: WARNING: control file is not current, control file
AUTOBACKUP skipped

```

2. Wiederholen Sie die Protokolle.

```

RMAN> recover copy of database with tag 'ONTAP_MIGRATION';
Starting recover at 13-MAY-16
using channel ORA_DISK_1
channel ORA_DISK_1: starting incremental datafile backup set restore
channel ORA_DISK_1: specifying datafile copies to recover
recovering datafile copy file number=00001 name=/oradata/TOAST/data_D-
TOAST_I-2098173325_TS-SYSTEM_FNO-1_01r5fhjg
recovering datafile copy file number=00002 name=/oradata/TOAST/data_D-
TOAST_I-2098173325_TS-SYSAUX_FNO-2_02r5fhjo
recovering datafile copy file number=00003 name=/oradata/TOAST/data_D-
TOAST_I-2098173325_TS-UNDOTBS1_FNO-3_03r5fhjt
recovering datafile copy file number=00004 name=/oradata/TOAST/data_D-
TOAST_I-2098173325_TS-USERS_FNO-4_05r5fhk6
channel ORA_DISK_1: reading from backup piece
/logs/TOAST/arch/09r5fj8i_1_1
channel ORA_DISK_1: piece handle=/logs/TOAST/arch/09r5fj8i_1_1
tag=ONTAP_MIGRATION
channel ORA_DISK_1: restored backup piece 1
channel ORA_DISK_1: restore complete, elapsed time: 00:00:01
Finished recover at 13-MAY-16
RMAN-06497: WARNING: control file is not current, control file
AUTOBACKUP skipped

```

Aktivierung

Die wiederhergestellte Steuerdatei verweist weiterhin auf die Datendateien am ursprünglichen Speicherort und enthält auch die Pfadinformationen für die kopierten Datendateien.

1. Um die aktiven Datendateien zu ändern, führen Sie den `switch database to copy` Befehl:

```

RMAN> switch database to copy;
datafile 1 switched to datafile copy "/oradata/TOAST/data_D-TOAST_I-
2098173325_TS-SYSTEM_FNO-1_01r5fhjg"
datafile 2 switched to datafile copy "/oradata/TOAST/data_D-TOAST_I-
2098173325_TS-SYSAUX_FNO-2_02r5fhjo"
datafile 3 switched to datafile copy "/oradata/TOAST/data_D-TOAST_I-
2098173325_TS-UNDOTBS1_FNO-3_03r5fhjt"
datafile 4 switched to datafile copy "/oradata/TOAST/data_D-TOAST_I-
2098173325_TS-USERS_FNO-4_05r5fhk6"

```

2. Obwohl die Datendateien vollständig konsistent sein sollten, ist ein letzter Schritt erforderlich, um die verbleibenden Änderungen, die in den Online-Wiederherstellungsprotokollen aufgezeichnet werden, wiederzugeben. Verwenden Sie die `recover database` Befehl, um diese Änderungen erneut einzuspielen und die Kopie 100 % mit dem Original zu identisch zu machen. Die Kopie ist jedoch noch nicht geöffnet.

```

RMAN> recover database;
Starting recover at 13-MAY-16
using channel ORA_DISK_1
starting media recovery
archived log for thread 1 with sequence 28 is already on disk as file
+ASM0/TOAST/redo01.log
archived log file name=+ASM0/TOAST/redo01.log thread=1 sequence=28
media recovery complete, elapsed time: 00:00:00
Finished recover at 13-MAY-16

```

Temporäre Datendateien Verschieben

1. Ermitteln Sie den Speicherort der temporären Datendateien, die noch auf der ursprünglichen Laufwerksgruppe verwendet werden.

```

RMAN> select file#||' '||name from v$tempfile;
FILE#||' '||NAME
-----
1 +ASM0/TOAST/temp01.dbf

```

2. Um die Datendateien zu verschieben, führen Sie die folgenden Befehle aus. Wenn es viele Tempfiles gibt, verwenden Sie einen Texteditor, um den RMAN-Befehl zu erstellen, und schneiden Sie ihn dann aus und fügen Sie ihn ein.

```

RMAN> run {
2> set newname for tempfile 1 to '/oradata/TOAST/temp01.dbf';
3> switch tempfile all;
4> }
executing command: SET NEWNAME
renamed tempfile 1 to /oradata/TOAST/temp01.dbf in control file

```

Migration des Wiederherstellungsprotokolls

Der Migrationsprozess ist fast abgeschlossen, aber die Wiederherstellungsprotokolle befinden sich immer noch in der ursprünglichen ASM-Laufwerksgruppe. Wiederherstellungsprotokolle können nicht direkt verschoben werden. Stattdessen wird ein neuer Satz von Wiederherstellungsprotokollen erstellt und der Konfiguration hinzugefügt, gefolgt von einem Drop der alten Protokolle.

1. Ermitteln Sie die Anzahl der Redo-Log-Gruppen und deren jeweilige Gruppennummern.

```

RMAN> select group#||' '||member from v$logfile;
GROUP#||' '||MEMBER
-----
-----
1 +ASM0/TOAST/redo01.log
2 +ASM0/TOAST/redo02.log
3 +ASM0/TOAST/redo03.log

```

2. Geben Sie die Größe der Wiederherstellungsprotokolle ein.

```

RMAN> select group#||' '||bytes from v$log;
GROUP#||' '||BYTES
-----
-----
1 52428800
2 52428800
3 52428800

```

3. Erstellen Sie für jedes Wiederherstellungsprotokoll eine neue Gruppe, indem Sie die gleiche Größe wie die aktuelle Wiederherstellungsprotokollgruppe verwenden, die den neuen Speicherort des Dateisystems verwendet.

```

RMAN> alter database add logfile '/logs/TOAST/redo/log00.rdo' size
52428800;
Statement processed
RMAN> alter database add logfile '/logs/TOAST/redo/log01.rdo' size
52428800;
Statement processed
RMAN> alter database add logfile '/logs/TOAST/redo/log02.rdo' size
52428800;
Statement processed

```

4. Entfernen Sie die alten Logfile-Gruppen, die sich noch im vorherigen Speicher befinden.

```

RMAN> alter database drop logfile group 4;
Statement processed
RMAN> alter database drop logfile group 5;
Statement processed
RMAN> alter database drop logfile group 6;
Statement processed

```

5. Wenn ein Fehler auftritt, der das Löschen eines aktiven Protokolls blockiert, erzwingen Sie einen Switch zum nächsten Protokoll, um die Sperre freizugeben und einen globalen Kontrollpunkt zu erzwingen. Ein

Beispiel ist unten dargestellt. Der Versuch, die Logfile-Gruppe 3, die sich am alten Speicherort befand, zu löschen, wurde abgelehnt, da noch aktive Daten in dieser Logdatei vorhanden waren. Eine Protokollarchivierung, gefolgt von einem Kontrollpunkt, ermöglicht das Löschen von Logdateien.

```

RMAN> alter database drop logfile group 4;
RMAN-00571: =====
RMAN-00569: ===== ERROR MESSAGE STACK FOLLOWS =====
RMAN-00571: =====
RMAN-03002: failure of sql statement command at 12/08/2015 20:23:51
ORA-01623: log 4 is current log for instance TOAST (thread 4) - cannot
drop
ORA-00312: online log 4 thread 1:
'+NEWLOGS/TOAST/ONLINELOG/group_4.266.897763123'
RMAN> alter system switch logfile;
Statement processed
RMAN> alter system checkpoint;
Statement processed
RMAN> alter database drop logfile group 4;
Statement processed

```

6. Überprüfen Sie die Umgebung, um sicherzustellen, dass alle standortbasierten Parameter aktualisiert werden.

```

SQL> select name from v$datafile;
SQL> select member from v$logfile;
SQL> select name from v$tempfile;
SQL> show parameter spfile;
SQL> select name, value from v$parameter where value is not null;

```

7. Das folgende Skript zeigt, wie Sie diesen Prozess vereinfachen können.

```

[root@jfscl current]# ./checkdbdata.pl TOAST
TOAST datafiles:
/oradata/TOAST/data_D-TOAST_I-2098173325_TS-SYSTEM_FNO-1_01r5fhjg
/oradata/TOAST/data_D-TOAST_I-2098173325_TS-SYSAUX_FNO-2_02r5fhjo
/oradata/TOAST/data_D-TOAST_I-2098173325_TS-UNDOTBS1_FNO-3_03r5fhjt
/oradata/TOAST/data_D-TOAST_I-2098173325_TS-USERS_FNO-4_05r5fhk6
TOAST redo logs:
/logs/TOAST/redo/log00.rdo
/logs/TOAST/redo/log01.rdo
/logs/TOAST/redo/log02.rdo
TOAST temp datafiles:
/oradata/TOAST/temp01.dbf
TOAST spfile
spfile                                string
/orabin/product/12.1.0/dbhome_
                                         1/dbs/spfileTOAST.ora
TOAST key parameters
control_files /logs/TOAST/arch/control01.ctl,
/logs/TOAST/redo/control02.ctl
log_archive_dest_1 LOCATION=/logs/TOAST/arch

```

8. Wenn die ASM-Datenträgergruppen vollständig evakuiert wurden, können sie jetzt mit abgehängt werden `asmcmd`. In vielen Fällen können Dateien, die zu anderen Datenbanken oder der ASM-Datei `spfile/passwd` gehören, weiterhin vorhanden sein.

```

-bash-4.1$ . oraenv
ORACLE_SID = [TOAST] ? +ASM
The Oracle base remains unchanged with value /orabin
-bash-4.1$ asmcmd
ASMCMD> umount DATA
ASMCMD>

```

Bereinigung der Datendatei

Der Migrationsprozess kann je nach Verwendung von Oracle RMAN zu Datendateien mit langer oder kryptischer Syntax führen. Im hier gezeigten Beispiel wurde das Backup mit dem Dateiformat von durchgeführt `/oradata/TOAST/%U. %U` Gibt an, dass RMAN für jede Datendatei einen eindeutigen Standardnamen erstellen sollte. Das Ergebnis ist ähnlich wie im folgenden Text dargestellt. Die traditionellen Namen der Datendateien sind in die Namen eingebettet. Dies kann mithilfe des in dargestellten skriptgesteuerten Ansatzes bereinigt werden "[Bereinigung der ASM-Migration](#)".

```

[root@jfscl current]# ./fixuniquenames.pl TOAST
#sqlplus Commands
shutdown immediate;
startup mount;
host mv /oradata/TOAST/data_D-TOAST_I-2098173325_TS-SYSTEM_FNO-1_01r5fhjg
/oradata/TOAST/system.dbf
host mv /oradata/TOAST/data_D-TOAST_I-2098173325_TS-SYSAUX_FNO-2_02r5fhjo
/oradata/TOAST/sysaux.dbf
host mv /oradata/TOAST/data_D-TOAST_I-2098173325_TS-UNDOTBS1_FNO-
3_03r5fhjt /oradata/TOAST/undotbs1.dbf
host mv /oradata/TOAST/data_D-TOAST_I-2098173325_TS-USERS_FNO-4_05r5fhk6
/oradata/TOAST/users.dbf
alter database rename file '/oradata/TOAST/data_D-TOAST_I-2098173325_TS-
SYSTEM_FNO-1_01r5fhjg' to '/oradata/TOAST/system.dbf';
alter database rename file '/oradata/TOAST/data_D-TOAST_I-2098173325_TS-
SYSAUX_FNO-2_02r5fhjo' to '/oradata/TOAST/sysaux.dbf';
alter database rename file '/oradata/TOAST/data_D-TOAST_I-2098173325_TS-
UNDOTBS1_FNO-3_03r5fhjt' to '/oradata/TOAST/undotbs1.dbf';
alter database rename file '/oradata/TOAST/data_D-TOAST_I-2098173325_TS-
USERS_FNO-4_05r5fhk6' to '/oradata/TOAST/users.dbf';
alter database open;

```

Oracle ASM-Ausgleich

Wie bereits erläutert, kann eine Oracle ASM-Festplattengruppe mithilfe des Ausgleichs transparent auf ein neues Storage-System migriert werden. Zusammenfassend ist zu sagen, dass beim Ausbalancieren der vorhandenen LUN-Gruppe LUNs gleicher Größe hinzugefügt werden müssen, gefolgt von einem Drop-Vorgang der vorherigen LUN. Oracle ASM verlagert die zugrunde liegenden Daten automatisch in einem optimalen Layout auf neuen Speicher und gibt dann die alten LUNs nach Abschluss frei.

Der Migrationsprozess nutzt effiziente sequenzielle I/O-Vorgänge und führt im Allgemeinen keine Performance-Unterbrechung durch. Bei Bedarf kann die Migrationsrate jedoch gedrosselt werden.

Identifizieren Sie die zu migrierenden Daten

```

SQL> select name||' '||group_number||' '||total_mb||' '||path||'
' ||header_status from v$asm_disk;
NEWDATA_0003 1 10240 /dev/mapper/3600a098038303537762b47594c315864 MEMBER
NEWDATA_0002 1 10240 /dev/mapper/3600a098038303537762b47594c315863 MEMBER
NEWDATA_0000 1 10240 /dev/mapper/3600a098038303537762b47594c315861 MEMBER
NEWDATA_0001 1 10240 /dev/mapper/3600a098038303537762b47594c315862 MEMBER
SQL> select group_number||' '||name from v$asm_diskgroup;
1 NEWDATA

```

Erstellen neuer LUNs

Erstellen Sie neue LUNs gleicher Größe und legen Sie die Mitgliedschaft für Benutzer und Gruppen nach Bedarf fest. Die LUNs sollten als angezeigt werden CANDIDATE Festplatten.

```
SQL> select name||' '||group_number||' '||total_mb||' '||path||'
' ||header_status from v$asm_disk;
 0 0 /dev/mapper/3600a098038303537762b47594c31586b CANDIDATE
 0 0 /dev/mapper/3600a098038303537762b47594c315869 CANDIDATE
 0 0 /dev/mapper/3600a098038303537762b47594c315858 CANDIDATE
 0 0 /dev/mapper/3600a098038303537762b47594c31586a CANDIDATE
NEWDATA_0003 1 10240 /dev/mapper/3600a098038303537762b47594c315864 MEMBER
NEWDATA_0002 1 10240 /dev/mapper/3600a098038303537762b47594c315863 MEMBER
NEWDATA_0000 1 10240 /dev/mapper/3600a098038303537762b47594c315861 MEMBER
NEWDATA_0001 1 10240 /dev/mapper/3600a098038303537762b47594c315862 MEMBER
```

Neue LUNS hinzufügen

Während die Add- und Drop-Vorgänge zusammen ausgeführt werden können, ist es in der Regel einfacher, neue LUNs in zwei Schritten hinzuzufügen. Fügen Sie zunächst die neuen LUNs der Festplattengruppe hinzu. Dieser Schritt führt dazu, dass die Hälfte der Extents von den aktuellen ASM-LUNs auf die neuen LUNs migriert wird.

Die Ausgleichskraft gibt die Rate an, mit der Daten übertragen werden. Je höher die Zahl, desto höher ist die Parallelität der Datenübertragung. Die Migration erfolgt mit effizienten sequenziellen I/O-Vorgängen, die wahrscheinlich keine Performance-Probleme verursachen. Auf Wunsch kann die Ausgleichskraft einer laufenden Migration jedoch mit dem angepassten `alter diskgroup [name] rebalance power [level]` Befehl. Für typische Migrationen wird der Wert 5 verwendet.

```
SQL> alter diskgroup NEWDATA add disk
'/dev/mapper/3600a098038303537762b47594c31586b' rebalance power 5;
Diskgroup altered.
SQL> alter diskgroup NEWDATA add disk
'/dev/mapper/3600a098038303537762b47594c315869' rebalance power 5;
Diskgroup altered.
SQL> alter diskgroup NEWDATA add disk
'/dev/mapper/3600a098038303537762b47594c315858' rebalance power 5;
Diskgroup altered.
SQL> alter diskgroup NEWDATA add disk
'/dev/mapper/3600a098038303537762b47594c31586a' rebalance power 5;
Diskgroup altered.
```

Überwachen Sie den Betrieb

Ein Ausgleichsoperation kann auf verschiedene Weise überwacht und verwaltet werden. Für dieses Beispiel haben wir den folgenden Befehl verwendet.

```
SQL> select group_number,operation,state from v$asm_operation;
GROUP_NUMBER OPERA STAT
-----
1 REBAL RUN
1 REBAL WAIT
```

Nach Abschluss der Migration werden keine Vorgänge zur Ausbalancierung gemeldet.

```
SQL> select group_number,operation,state from v$asm_operation;
no rows selected
```

Alte LUNs ablegen

Die Migration ist nun zur Hälfte abgeschlossen. Einige grundlegende Performance-Tests stellen sicher, dass die Umgebung sich in einem ordnungsgemäßen Zustand befindet. Nach Bestätigung können die verbleibenden Daten durch Löschen der alten LUNs verschoben werden. Beachten Sie, dass dies nicht zur sofortigen Freigabe der LUNs führt. Der Drop-Vorgang signalisiert Oracle ASM, die Extents zuerst zu verschieben und dann die LUN freizugeben.

```
sqlplus / as sysasm
SQL> alter diskgroup NEWDATA drop disk NEWDATA_0000 rebalance power 5;
Diskgroup altered.
SQL> alter diskgroup NEWDATA drop disk NEWDATA_0001 rebalance power 5;
Diskgroup altered.
SQL> alter diskgroup newdata drop disk NEWDATA_0002 rebalance power 5;
Diskgroup altered.
SQL> alter diskgroup newdata drop disk NEWDATA_0003 rebalance power 5;
Diskgroup altered.
```

Überwachen Sie den Betrieb

Der Ausgleichsoperation kann auf verschiedene Weise überwacht und verwaltet werden. Für dieses Beispiel haben wir den folgenden Befehl verwendet:

```
SQL> select group_number,operation,state from v$asm_operation;
GROUP_NUMBER OPERA STAT
-----
1 REBAL RUN
1 REBAL WAIT
```

Nach Abschluss der Migration werden keine Vorgänge zur Ausbalancierung gemeldet.

```
SQL> select group_number,operation,state from v$asm_operation;
no rows selected
```

Entfernen Sie alte LUNs

Bevor Sie die alten LUNs aus der Laufwerksgruppe entfernen, sollten Sie den Header-Status einer letzten Prüfung entnehmen. Nachdem eine LUN aus ASM freigegeben wurde, wird kein Name mehr aufgeführt, und der Kopfzeilenstatus wird als aufgeführt FORMER. Dies bedeutet, dass diese LUNs sicher aus dem System entfernt werden können.

```
SQL> select name||' '||group_number||' '||total_mb||' '||path||'
'||header_status from v$asm_disk;
NAME||' '||GROUP_NUMBER||' '||TOTAL_MB||' '||PATH||' '||HEADER_STATUS
-----
-----
0 0 /dev/mapper/3600a098038303537762b47594c315863 FORMER
0 0 /dev/mapper/3600a098038303537762b47594c315864 FORMER
0 0 /dev/mapper/3600a098038303537762b47594c315861 FORMER
0 0 /dev/mapper/3600a098038303537762b47594c315862 FORMER
NEWDATA_0005 1 10240 /dev/mapper/3600a098038303537762b47594c315869 MEMBER
NEWDATA_0007 1 10240 /dev/mapper/3600a098038303537762b47594c31586a MEMBER
NEWDATA_0004 1 10240 /dev/mapper/3600a098038303537762b47594c31586b MEMBER
NEWDATA_0006 1 10240 /dev/mapper/3600a098038303537762b47594c315858 MEMBER
8 rows selected.
```

LVM-Migration

Das hier vorgestellte Verfahren zeigt die Prinzipien einer LVM-basierten Migration einer Volume-Gruppe namens `datavg`. Die Beispiele stammen aus Linux LVM, die Prinzipien gelten jedoch gleichermaßen für AIX, HP-UX und VxVM. Die genauen Befehle können variieren.

1. Identifizieren Sie die LUNs, die sich derzeit im befinden `datavg` Volume-Gruppe.

```
[root@host1 ~]# pvdisplay -C | grep datavg
/dev/mapper/3600a098038303537762b47594c31582f datavg lvm2 a-- 10.00g
10.00g
/dev/mapper/3600a098038303537762b47594c31585a datavg lvm2 a-- 10.00g
10.00g
/dev/mapper/3600a098038303537762b47594c315859 datavg lvm2 a-- 10.00g
10.00g
/dev/mapper/3600a098038303537762b47594c31586c datavg lvm2 a-- 10.00g
10.00g
```

2. Erstellen Sie neue LUNs mit derselben oder einer etwas größeren physischen Größe und definieren Sie sie als physische Volumes.

```
[root@host1 ~]# pvcreate /dev/mapper/3600a098038303537762b47594c315864
Physical volume "/dev/mapper/3600a098038303537762b47594c315864"
successfully created
[root@host1 ~]# pvcreate /dev/mapper/3600a098038303537762b47594c315863
Physical volume "/dev/mapper/3600a098038303537762b47594c315863"
successfully created
[root@host1 ~]# pvcreate /dev/mapper/3600a098038303537762b47594c315862
Physical volume "/dev/mapper/3600a098038303537762b47594c315862"
successfully created
[root@host1 ~]# pvcreate /dev/mapper/3600a098038303537762b47594c315861
Physical volume "/dev/mapper/3600a098038303537762b47594c315861"
successfully created
```

3. Fügen Sie die neuen Volumes zur Volume-Gruppe hinzu.

```
[root@host1 tmp]# vgextend datavg
/dev/mapper/3600a098038303537762b47594c315864
Volume group "datavg" successfully extended
[root@host1 tmp]# vgextend datavg
/dev/mapper/3600a098038303537762b47594c315863
Volume group "datavg" successfully extended
[root@host1 tmp]# vgextend datavg
/dev/mapper/3600a098038303537762b47594c315862
Volume group "datavg" successfully extended
[root@host1 tmp]# vgextend datavg
/dev/mapper/3600a098038303537762b47594c315861
Volume group "datavg" successfully extended
```

4. Stellen Sie das aus `pvmove` Befehl, um die Extents jeder aktuellen LUN in die neue LUN zu verschieben. Der `-i [seconds]` Argument überwacht den Fortschritt des Vorgangs.

```
[root@host1 tmp]# pvmove -i 10
/dev/mapper/3600a098038303537762b47594c31582f
/dev/mapper/3600a098038303537762b47594c315864
  /dev/mapper/3600a098038303537762b47594c31582f: Moved: 0.0%
  /dev/mapper/3600a098038303537762b47594c31582f: Moved: 14.2%
  /dev/mapper/3600a098038303537762b47594c31582f: Moved: 28.4%
  /dev/mapper/3600a098038303537762b47594c31582f: Moved: 42.5%
  /dev/mapper/3600a098038303537762b47594c31582f: Moved: 57.1%
  /dev/mapper/3600a098038303537762b47594c31582f: Moved: 72.3%
  /dev/mapper/3600a098038303537762b47594c31582f: Moved: 87.3%
  /dev/mapper/3600a098038303537762b47594c31582f: Moved: 100.0%
[root@host1 tmp]# pvmove -i 10
/dev/mapper/3600a098038303537762b47594c31585a
/dev/mapper/3600a098038303537762b47594c315863
  /dev/mapper/3600a098038303537762b47594c31585a: Moved: 0.0%
  /dev/mapper/3600a098038303537762b47594c31585a: Moved: 14.9%
  /dev/mapper/3600a098038303537762b47594c31585a: Moved: 29.9%
  /dev/mapper/3600a098038303537762b47594c31585a: Moved: 44.8%
  /dev/mapper/3600a098038303537762b47594c31585a: Moved: 60.1%
  /dev/mapper/3600a098038303537762b47594c31585a: Moved: 75.8%
  /dev/mapper/3600a098038303537762b47594c31585a: Moved: 90.9%
  /dev/mapper/3600a098038303537762b47594c31585a: Moved: 100.0%
[root@host1 tmp]# pvmove -i 10
/dev/mapper/3600a098038303537762b47594c315859
/dev/mapper/3600a098038303537762b47594c315862
  /dev/mapper/3600a098038303537762b47594c315859: Moved: 0.0%
  /dev/mapper/3600a098038303537762b47594c315859: Moved: 14.8%
  /dev/mapper/3600a098038303537762b47594c315859: Moved: 29.8%
  /dev/mapper/3600a098038303537762b47594c315859: Moved: 45.5%
  /dev/mapper/3600a098038303537762b47594c315859: Moved: 61.1%
  /dev/mapper/3600a098038303537762b47594c315859: Moved: 76.6%
  /dev/mapper/3600a098038303537762b47594c315859: Moved: 91.7%
  /dev/mapper/3600a098038303537762b47594c315859: Moved: 100.0%
[root@host1 tmp]# pvmove -i 10
/dev/mapper/3600a098038303537762b47594c31586c
/dev/mapper/3600a098038303537762b47594c315861
  /dev/mapper/3600a098038303537762b47594c31586c: Moved: 0.0%
  /dev/mapper/3600a098038303537762b47594c31586c: Moved: 15.0%
  /dev/mapper/3600a098038303537762b47594c31586c: Moved: 30.4%
  /dev/mapper/3600a098038303537762b47594c31586c: Moved: 46.0%
  /dev/mapper/3600a098038303537762b47594c31586c: Moved: 61.4%
  /dev/mapper/3600a098038303537762b47594c31586c: Moved: 77.2%
  /dev/mapper/3600a098038303537762b47594c31586c: Moved: 92.3%
  /dev/mapper/3600a098038303537762b47594c31586c: Moved: 100.0%
```

5. Wenn dieser Vorgang abgeschlossen ist, löschen Sie die alten LUNs aus der Volume-Gruppe mithilfe von `vgreduce` Befehl. Wenn die LUN erfolgreich war, kann sie jetzt sicher aus dem System entfernt werden.

```
[root@host1 tmp]# vgreduce datavg
/dev/mapper/3600a098038303537762b47594c31582f
Removed "/dev/mapper/3600a098038303537762b47594c31582f" from volume
group "datavg"
[root@host1 tmp]# vgreduce datavg
/dev/mapper/3600a098038303537762b47594c31585a
  Removed "/dev/mapper/3600a098038303537762b47594c31585a" from volume
group "datavg"
[root@host1 tmp]# vgreduce datavg
/dev/mapper/3600a098038303537762b47594c315859
  Removed "/dev/mapper/3600a098038303537762b47594c315859" from volume
group "datavg"
[root@host1 tmp]# vgreduce datavg
/dev/mapper/3600a098038303537762b47594c31586c
  Removed "/dev/mapper/3600a098038303537762b47594c31586c" from volume
group "datavg"
```

Import fremder LUNs

Planung

Die Verfahren zur Migration von SAN-Ressourcen mit FLI sind in NetApp dokumentiert ["ONTAP Dokumentation zum Importieren fremder LUNs"](#).

Aus Sicht der Datenbank und des Hosts sind keine besonderen Schritte erforderlich. Nachdem die FC-Zonen aktualisiert wurden und die LUNs auf ONTAP verfügbar werden, sollte die LVM in der Lage sein, die LVM-Metadaten von den LUNs zu lesen. Außerdem sind die Volume-Gruppen ohne weitere Konfigurationsschritte einsatzbereit. In seltenen Fällen können Umgebungen Konfigurationsdateien enthalten, die hartcodiert waren und Verweise auf das vorherige Storage-Array enthalten. Zum Beispiel ein Linux-System, das enthalten `/etc/multipath.conf` Regeln, die auf einen WWN eines bestimmten Geräts verwiesen haben, müssen aktualisiert werden, um die von FLI eingeführten Änderungen wiederzugeben.



Informationen zu unterstützten Konfigurationen finden Sie in der NetApp Kompatibilitätsmatrix. Falls Ihr System nicht im Lieferumfang enthalten ist, wenden Sie sich an Ihren NetApp Ansprechpartner.

Dieses Beispiel zeigt die Migration von ASM- und LVM-LUNs, die auf einem Linux-Server gehostet werden. FLI wird auf anderen Betriebssystemen unterstützt, und obwohl die Host-seitigen Befehle unterschiedlich sein können, sind die Prinzipien identisch, und die ONTAP-Verfahren sind identisch.

LVM-LUNs identifizieren

Der erste Schritt zur Vorbereitung besteht darin, die zu migrierenden LUNs zu identifizieren. In dem hier gezeigten Beispiel werden zwei SAN-basierte Dateisysteme in gemountet `/orabin` Und `/backups`.

```
[root@host1 ~]# df -k
Filesystem                1K-blocks      Used Available Use%
Mounted on
/dev/mapper/rhel-root      52403200    8811464  43591736  17% /
devtmpfs                   65882776         0  65882776   0% /dev
...
fas8060-nfs-public:/install 199229440 119368128  79861312  60%
/install
/dev/mapper/sanvg-lvorabin  20961280  12348476   8612804  59%
/orabin
/dev/mapper/sanvg-lvbackups 73364480  62947536  10416944  86%
/backups
```

Der Name der Volume-Gruppe kann aus dem Gerätenamen extrahiert werden, der das Format (Name der Volume-Gruppe)-(Name des logischen Volumes) verwendet. In diesem Fall wird die Volume-Gruppe aufgerufen sanvg.

Der `pvdisplay` Mit dem Befehl können Sie die LUNs identifizieren, die diese Volume-Gruppe unterstützen. In diesem Fall sind 10 LUNs vorhanden sanvg Volume-Gruppe.

```
[root@host1 ~]# pvdisplay -C -o pv_name,pv_size,pv_fmt,vg_name
PV                               PSize  VG
/dev/mapper/3600a0980383030445424487556574266 10.00g sanvg
/dev/mapper/3600a0980383030445424487556574267 10.00g sanvg
/dev/mapper/3600a0980383030445424487556574268 10.00g sanvg
/dev/mapper/3600a0980383030445424487556574269 10.00g sanvg
/dev/mapper/3600a098038303044542448755657426a 10.00g sanvg
/dev/mapper/3600a098038303044542448755657426b 10.00g sanvg
/dev/mapper/3600a098038303044542448755657426c 10.00g sanvg
/dev/mapper/3600a098038303044542448755657426d 10.00g sanvg
/dev/mapper/3600a098038303044542448755657426e 10.00g sanvg
/dev/mapper/3600a098038303044542448755657426f 10.00g sanvg
/dev/sda2                          278.38g rhel
```

ASM-LUNs identifizieren

ASM-LUNs müssen ebenfalls migriert werden. Um die Anzahl der LUNs und LUN-Pfade von sqlplus als sysasm-Benutzer zu erhalten, führen Sie den folgenden Befehl aus:

```

SQL> select path||' '||os_mb from v$asm_disk;
PATH||' '||OS_MB
-----
-----
/dev/oracleasm/disks/ASM0 10240
/dev/oracleasm/disks/ASM9 10240
/dev/oracleasm/disks/ASM8 10240
/dev/oracleasm/disks/ASM7 10240
/dev/oracleasm/disks/ASM6 10240
/dev/oracleasm/disks/ASM5 10240
/dev/oracleasm/disks/ASM4 10240
/dev/oracleasm/disks/ASM1 10240
/dev/oracleasm/disks/ASM3 10240
/dev/oracleasm/disks/ASM2 10240
10 rows selected.
SQL>

```

Änderungen am FC-Netzwerk

Die aktuelle Umgebung enthält 20 zu migrierende LUNs. Aktualisieren Sie das aktuelle SAN, damit ONTAP auf die aktuellen LUNs zugreifen kann. Daten werden noch nicht migriert, aber ONTAP muss die Konfigurationsinformationen der aktuellen LUNs lesen, um das neue Zuhause für diese Daten zu erstellen.

Mindestens ein HBA-Port auf dem All Flash FAS/FAS System muss als Initiator-Port konfiguriert sein. Zudem müssen die FC-Zonen aktualisiert werden, damit ONTAP auf die LUNs auf dem fremden Storage Array zugreifen können. Bei einigen Speicher-Arrays ist die LUN-Maskierung konfiguriert, wodurch WWNs auf eine bestimmte LUN zugreifen können. In diesen Fällen muss die LUN-Maskierung ebenfalls aktualisiert werden, um Zugriff auf die ONTAP-WWNs zu gewähren.

Nach Abschluss dieses Schritts sollte ONTAP in der Lage sein, das fremde Speicher-Array mit dem anzuzeigen `storage array show` Befehl. Das Schlüsselfeld, das zurückgegeben wird, ist das Präfix, das zur Identifizierung der fremden LUN auf dem System verwendet wird. Im folgenden Beispiel werden die LUNs auf dem Fremdarray angezeigt `FOREIGN_1` Wird in ONTAP mit dem Präfix von angezeigt `FOR-1`.

Identifizierung von Fremdarrays

```

Cluster01::> storage array show -fields name,prefix
name           prefix
-----
FOREIGN_1      FOR-1
Cluster01::>

```

Identifizierung fremder LUNs

Die LUNs können durch Bestehen des aufgelistet werden `array-name` Bis zum `storage disk show` Befehl. Die zurückgegebenen Daten werden während des Migrationsvorgangs mehrfach referenziert.

```

Cluster01::> storage disk show -array-name FOREIGN_1 -fields disk,serial
disk      serial-number
-----  -
FOR-1.1   800DT$HuVWBX
FOR-1.2   800DT$HuVWBZ
FOR-1.3   800DT$HuVWBW
FOR-1.4   800DT$HuVWBX
FOR-1.5   800DT$HuVWB/
FOR-1.6   800DT$HuVWBa
FOR-1.7   800DT$HuVWBd
FOR-1.8   800DT$HuVWBb
FOR-1.9   800DT$HuVWBc
FOR-1.10  800DT$HuVWBc
FOR-1.11  800DT$HuVWBf
FOR-1.12  800DT$HuVWBg
FOR-1.13  800DT$HuVWBh
FOR-1.14  800DT$HuVWBh
FOR-1.15  800DT$HuVWBj
FOR-1.16  800DT$HuVWBk
FOR-1.17  800DT$HuVWBm
FOR-1.18  800DT$HuVWBn
FOR-1.19  800DT$HuVWBn
FOR-1.20  800DT$HuVWBn
20 entries were displayed.
Cluster01::>

```

Registrieren Sie LUNs für Fremdarrays als Importkandidaten

Die ausländischen LUNs werden zunächst als jeder bestimmte LUN-Typ klassifiziert. Bevor Daten importiert werden können, müssen die LUNs als fremd gekennzeichnet werden und daher als Kandidat für den Importprozess. Um diesen Schritt abzuschließen, geben Sie die Seriennummer an den weiter `storage disk modify` Wie im folgenden Beispiel gezeigt. Beachten Sie, dass bei diesem Prozess nur die LUN als fremd innerhalb von ONTAP markiert wird. Es werden keine Daten auf die fremde LUN selbst geschrieben.

```

Cluster01::*> storage disk modify {-serial-number 800DT$HuVWBW} -is
-foreign true
Cluster01::*> storage disk modify {-serial-number 800DT$HuVWBX} -is
-foreign true
...
Cluster01::*> storage disk modify {-serial-number 800DT$HuVWBn} -is
-foreign true
Cluster01::*> storage disk modify {-serial-number 800DT$HuVWBn} -is
-foreign true
Cluster01::*>

```

Erstellung von Volumes zum Hosten migrierter LUNs

Ein Volume ist erforderlich, um die migrierten LUNs zu hosten. Die genaue Volume-Konfiguration hängt von der Planung der Nutzung von ONTAP Funktionen ab. In diesem Beispiel werden die ASM-LUNs in einem Volume platziert und die LVM-LUNs in einem zweiten Volume platziert. Auf diese Weise können Sie die LUNs als unabhängige Gruppen managen, beispielsweise für Tiering, die Erstellung von Snapshots oder die Einstellung von QoS-Kontrollen.

Stellen Sie die ein `snapshot-policy`to`none`. Der Migrationsprozess kann sehr viel Datenfluktuation beinhalten. Daher kann es zu einem starken Anstieg des Platzverbrauchs kommen, wenn Snapshots versehentlich erstellt werden, weil unerwünschte Daten in den Snapshots erfasst werden.

```
Cluster01::> volume create -volume new_asm -aggregate data_02 -size 120G
-snapshot-policy none
[Job 1152] Job succeeded: Successful
Cluster01::> volume create -volume new_lvm -aggregate data_02 -size 120G
-snapshot-policy none
[Job 1153] Job succeeded: Successful
Cluster01::>
```

Erstellen Sie ONTAP-LUNs

Nach der Erstellung der Volumes müssen die neuen LUNs erstellt werden. Normalerweise erfordert die Erstellung einer LUN, dass der Benutzer Informationen wie die LUN-Größe angeben muss. In diesem Fall wird jedoch das Argument für eine fremde Festplatte an den Befehl übergeben. Infolgedessen repliziert ONTAP die aktuellen LUN-Konfigurationsdaten von der angegebenen Seriennummer. Außerdem werden die LUN-Geometrie und Partitionstabellen-Daten verwendet, um die LUN-Ausrichtung anzupassen und eine optimale Performance herzustellen.

In diesem Schritt müssen die Seriennummern mit dem Fremdarray verglichen werden, um sicherzustellen, dass die richtige fremde LUN mit der richtigen neuen LUN abgeglichen wird.

```
Cluster01::*> lun create -vserver vserver1 -path /vol/new_asm/LUN0 -ostype
linux -foreign-disk 800DT$HuVWBW
Created a LUN of size 10g (10737418240)
Cluster01::*> lun create -vserver vserver1 -path /vol/new_asm/LUN1 -ostype
linux -foreign-disk 800DT$HuVWBX
Created a LUN of size 10g (10737418240)
...
Created a LUN of size 10g (10737418240)
Cluster01::*> lun create -vserver vserver1 -path /vol/new_lvm/LUN8 -ostype
linux -foreign-disk 800DT$HuVWBn
Created a LUN of size 10g (10737418240)
Cluster01::*> lun create -vserver vserver1 -path /vol/new_lvm/LUN9 -ostype
linux -foreign-disk 800DT$HuVWBo
Created a LUN of size 10g (10737418240)
```

Erstellen Sie Importbeziehungen

Die LUNs wurden jetzt erstellt, sind aber nicht als Replikationsziel konfiguriert. Bevor dieser Schritt durchgeführt werden kann, müssen die LUNs zunächst in den Offline-Modus versetzt werden. Dieser zusätzliche Schritt dient dem Schutz von Daten vor Benutzerfehlern. Wenn ONTAP die Durchführung einer Migration auf einer Online-LUN zulässt, besteht das Risiko, dass durch einen typografischen Fehler aktive Daten überschrieben werden. Durch den zusätzlichen Schritt, den Benutzer zum ersten Mal offline zu schalten, wird überprüft, ob die richtige Ziel-LUN als Migrationsziel verwendet wird.

```
Cluster01::*> lun offline -vserver vserver1 -path /vol/new_asm/LUN0
Warning: This command will take LUN "/vol/new_asm/LUN0" in Vserver
        "vserver1" offline.
Do you want to continue? {y|n}: y
Cluster01::*> lun offline -vserver vserver1 -path /vol/new_asm/LUN1
Warning: This command will take LUN "/vol/new_asm/LUN1" in Vserver
        "vserver1" offline.
Do you want to continue? {y|n}: y
...
Warning: This command will take LUN "/vol/new_lvm/LUN8" in Vserver
        "vserver1" offline.
Do you want to continue? {y|n}: y
Cluster01::*> lun offline -vserver vserver1 -path /vol/new_lvm/LUN9
Warning: This command will take LUN "/vol/new_lvm/LUN9" in Vserver
        "vserver1" offline.
Do you want to continue? {y|n}: y
```

Nachdem die LUNs offline sind, können Sie die Importbeziehung wiederherstellen, indem Sie die Seriennummer der fremden LUN an den übergeben `lun import create` Befehl.

```
Cluster01::*> lun import create -vserver vserver1 -path /vol/new_asm/LUN0
-foreign-disk 800DT$HuVWBW
Cluster01::*> lun import create -vserver vserver1 -path /vol/new_asm/LUN1
-foreign-disk 800DT$HuVWBX
...
Cluster01::*> lun import create -vserver vserver1 -path /vol/new_lvm/LUN8
-foreign-disk 800DT$HuVWBn
Cluster01::*> lun import create -vserver vserver1 -path /vol/new_lvm/LUN9
-foreign-disk 800DT$HuVWBo
Cluster01::*>
```

Nachdem alle Importbeziehungen eingerichtet sind, können die LUNs wieder online geschaltet werden.

```
Cluster01::*> lun online -vserver vserver1 -path /vol/new_asm/LUN0
Cluster01::*> lun online -vserver vserver1 -path /vol/new_asm/LUN1
...
Cluster01::*> lun online -vserver vserver1 -path /vol/new_lvm/LUN8
Cluster01::*> lun online -vserver vserver1 -path /vol/new_lvm/LUN9
Cluster01::*>
```

Erstellen einer Initiatorgruppe

Eine Initiatorgruppe (Initiatorgruppe) ist Teil der ONTAP LUN-Masking-Architektur. Auf eine neu erstellte LUN kann nur dann zugegriffen werden, wenn einem Host der erste Zugriff gewährt wurde. Dazu wird eine Initiatorgruppe erstellt, die entweder die FC-WWNs oder iSCSI-Initiatornamen auflistet, denen Zugriff gewährt werden soll. Zum Zeitpunkt der Erstellung dieses Berichts wurde FLI nur für FC LUNs unterstützt. Die Konvertierung in iSCSI nach der Migration ist jedoch eine einfache Aufgabe, wie in dargestellt "[Protokollkonvertierung](#)".

In diesem Beispiel wird eine Initiatorgruppe erstellt, die zwei WWNs enthält, die den beiden auf dem HBA des Hosts verfügbaren Ports entsprechen.

```
Cluster01::*> igroup create linuxhost -protocol fcp -ostype linux
-initiator 21:00:00:0e:1e:16:63:50 21:00:00:0e:1e:16:63:51
```

Ordnen Sie neue LUNs dem Host zu

Nach der Erstellung der Initiatorgruppe werden die LUNs dann der definierten Initiatorgruppe zugeordnet. Diese LUNs sind nur für die WWNs dieser Initiatorgruppe verfügbar. NetApp geht in dieser Phase des Migrationsprozesses davon aus, dass der Host nicht auf ONTAP abgegrenzt wurde. Dies ist wichtig, denn wenn der Host gleichzeitig auf das fremde Array und das neue ONTAP-System begrenzt ist, besteht das Risiko, dass LUNs mit derselben Seriennummer auf jedem Array erkannt werden können. Diese Situation kann zu Fehlfunktionen des Multipfad-Funktionszubehörs oder zu Schäden an Daten führen.

```
Cluster01::*> lun map -vserver vserver1 -path /vol/new_asm/LUN0 -igroup
linuxhost
Cluster01::*> lun map -vserver vserver1 -path /vol/new_asm/LUN1 -igroup
linuxhost
...
Cluster01::*> lun map -vserver vserver1 -path /vol/new_lvm/LUN8 -igroup
linuxhost
Cluster01::*> lun map -vserver vserver1 -path /vol/new_lvm/LUN9 -igroup
linuxhost
Cluster01::*>
```

Umstellung

Aufgrund der Notwendigkeit, die FC-Netzwerkconfiguration zu ändern, sind Unterbrechungen beim Import fremder LUNs unvermeidbar. Die Unterbrechung muss

jedoch nicht viel länger dauern als die Zeit, die für den Neustart der Datenbankumgebung und die Aktualisierung des FC-Zoning für die Umstellung der Host-FC-Konnektivität von der fremden LUN auf ONTAP erforderlich ist.

Dieser Prozess lässt sich wie folgt zusammenfassen:

1. Legen Sie alle LUN-Aktivitäten auf den fremden LUNs still.
2. Umleiten von Host-FC-Verbindungen zum neuen ONTAP-System
3. Starten Sie den Importvorgang.
4. Ermitteln Sie die LUNs neu.
5. Starten Sie die Datenbank neu.

Sie müssen nicht warten, bis der Migrationsprozess abgeschlossen ist. Sobald die Migration einer bestimmten LUN beginnt, ist sie auf ONTAP verfügbar und kann Daten bereitstellen, während der Datenkopiervorgang fortgesetzt wird. Alle Lesevorgänge werden an die fremde LUN weitergeleitet, und alle Schreibvorgänge werden synchron auf beide Arrays geschrieben. Der Kopiervorgang läuft sehr schnell ab und der Overhead bei der Umleitung des FC-Datenverkehrs ist minimal. Die Auswirkungen auf die Performance sollten daher kurzlebig und minimal sein. Wenn Bedenken bestehen, können Sie den Neustart der Umgebung verzögern, bis der Migrationsprozess abgeschlossen ist und die Importbeziehungen gelöscht wurden.

Datenbank herunterfahren

Der erste Schritt bei der Stilllegung der Umgebung in diesem Beispiel ist das Herunterfahren der Datenbank.

```
[oracle@host1 bin]$ . oraenv
ORACLE_SID = [oracle] ? FLIDB
The Oracle base remains unchanged with value /orabin
[oracle@host1 bin]$ sqlplus / as sysdba
SQL*Plus: Release 12.1.0.2.0
Copyright (c) 1982, 2014, Oracle. All rights reserved.
Connected to:
Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 - 64bit
Production
With the Partitioning, Automatic Storage Management, OLAP, Advanced
Analytics
and Real Application Testing options
SQL> shutdown immediate;
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL>
```

Netzdienste herunterfahren

Zu den migrierten SAN-basierten Dateisystemen gehören auch die Oracle ASM-Services. Um die zugrunde liegenden LUNs stilllegen zu können, müssen die Dateisysteme getrennt werden. Dies bedeutet wiederum, dass alle Prozesse mit offenen Dateien auf diesem Dateisystem angehalten werden.

```
[oracle@host1 bin]$ ./crsctl stop has -f
CRS-2791: Starting shutdown of Oracle High Availability Services-managed
resources on 'host1'
CRS-2673: Attempting to stop 'ora.evmd' on 'host1'
CRS-2673: Attempting to stop 'ora.DATA.dg' on 'host1'
CRS-2673: Attempting to stop 'ora.LISTENER.lsnr' on 'host1'
CRS-2677: Stop of 'ora.DATA.dg' on 'host1' succeeded
CRS-2673: Attempting to stop 'ora.asm' on 'host1'
CRS-2677: Stop of 'ora.LISTENER.lsnr' on 'host1' succeeded
CRS-2677: Stop of 'ora.evmd' on 'host1' succeeded
CRS-2677: Stop of 'ora.asm' on 'host1' succeeded
CRS-2673: Attempting to stop 'ora.cssd' on 'host1'
CRS-2677: Stop of 'ora.cssd' on 'host1' succeeded
CRS-2793: Shutdown of Oracle High Availability Services-managed resources
on 'host1' has completed
CRS-4133: Oracle High Availability Services has been stopped.
[oracle@host1 bin]$
```

Entfernen Sie Dateisysteme

Wenn alle Prozesse heruntergefahren werden, ist der umount-Vorgang erfolgreich. Wenn die Berechtigung verweigert wird, muss es einen Prozess mit einer Sperre auf dem Dateisystem geben. Der `fuser` Der Befehl kann bei der Identifizierung dieser Prozesse helfen.

```
[root@host1 ~]# umount /orabin
[root@host1 ~]# umount /backups
```

Deaktivieren Sie Volume-Gruppen

Nachdem alle Dateisysteme in einer bestimmten Volume-Gruppe getrennt wurden, kann die Volume-Gruppe deaktiviert werden.

```
[root@host1 ~]# vgchange --activate n sanvg
  0 logical volume(s) in volume group "sanvg" now active
[root@host1 ~]#
```

Änderungen am FC-Netzwerk

Die FC-Zonen können jetzt aktualisiert werden, um den gesamten Zugriff vom Host auf das fremde Array zu entfernen und den Zugriff auf ONTAP zu ermöglichen.

Importvorgang starten

Um die LUN-Importprozesse zu starten, führen Sie den `lun import start` Befehl.

```

Cluster01::lun import*> lun import start -vserver vserver1 -path
/vol/new_asm/LUN0
Cluster01::lun import*> lun import start -vserver vserver1 -path
/vol/new_asm/LUN1
...
Cluster01::lun import*> lun import start -vserver vserver1 -path
/vol/new_lvm/LUN8
Cluster01::lun import*> lun import start -vserver vserver1 -path
/vol/new_lvm/LUN9
Cluster01::lun import*>

```

Überwachen Sie den Importfortschritt

Der Importvorgang kann mit dem überwacht werden `lun import show` Befehl. Wie unten dargestellt, läuft der Import aller 20 LUNs, was bedeutet, dass die Daten jetzt über ONTAP zugänglich sind, obwohl der Kopiervorgang noch fortschreitet.

```

Cluster01::lun import*> lun import show -fields path,percent-complete
vserver    foreign-disk path                percent-complete
-----
vserver1   800DT$HuVWB/ /vol/new_asm/LUN4 5
vserver1   800DT$HuVWBW /vol/new_asm/LUN0 5
vserver1   800DT$HuVWBX /vol/new_asm/LUN1 6
vserver1   800DT$HuVWBZ /vol/new_asm/LUN2 6
vserver1   800DT$HuVWBZ /vol/new_asm/LUN3 5
vserver1   800DT$HuVWBa /vol/new_asm/LUN5 4
vserver1   800DT$HuVWBb /vol/new_asm/LUN6 4
vserver1   800DT$HuVWBc /vol/new_asm/LUN7 4
vserver1   800DT$HuVWBd /vol/new_asm/LUN8 4
vserver1   800DT$HuVWBe /vol/new_asm/LUN9 4
vserver1   800DT$HuVWBf /vol/new_lvm/LUN0 5
vserver1   800DT$HuVWBg /vol/new_lvm/LUN1 4
vserver1   800DT$HuVWBh /vol/new_lvm/LUN2 4
vserver1   800DT$HuVWBi /vol/new_lvm/LUN3 3
vserver1   800DT$HuVWBj /vol/new_lvm/LUN4 3
vserver1   800DT$HuVWBk /vol/new_lvm/LUN5 3
vserver1   800DT$HuVWB1 /vol/new_lvm/LUN6 4
vserver1   800DT$HuVWBm /vol/new_lvm/LUN7 3
vserver1   800DT$HuVWBn /vol/new_lvm/LUN8 2
vserver1   800DT$HuVWBo /vol/new_lvm/LUN9 2
20 entries were displayed.

```

Wenn Sie einen Offline-Prozess benötigen, verzögern Sie die Neuermittlung oder den Neustart von Diensten, bis der `lun import show` Befehl anzeigt, dass alle Migration erfolgreich und abgeschlossen ist. Anschließend können Sie den Migrationsprozess wie unter beschrieben abschließen ["Import fremder LUNs –](#)

Abschluss".

Wenn Sie eine Online-Migration benötigen, fahren Sie mit der Neuerkennung der LUNs in ihrem neuen Zuhause fort, und führen Sie die Dienste aus.

Nach SCSI-Geräteänderungen suchen

In den meisten Fällen besteht die einfachste Möglichkeit, neue LUNs neu zu ermitteln, darin, den Host neu zu starten. Dadurch werden alte veraltete Geräte automatisch entfernt, alle neuen LUNs ordnungsgemäß erkannt und verbundene Geräte wie Multipathing-Geräte erstellt. Das Beispiel zeigt einen vollständig online-Prozess zu Demonstrationszwecken.

Achtung: Bevor Sie einen Host neu starten, stellen Sie sicher, dass alle Einträge in `/etc/fstab` diese Referenz migrierte SAN-Ressourcen werden kommentiert. Wenn dies nicht durchgeführt wird und Probleme mit dem LUN-Zugriff auftreten, wird das OS möglicherweise nicht gebootet. Diese Situation beschädigt Daten nicht. Es kann jedoch sehr unbequem sein, in den Rettungsmodus oder einen ähnlichen Modus zu starten und die zu korrigieren `/etc/fstab` Damit das OS gebootet werden kann, um die Fehlerbehebung zu ermöglichen.

Die LUNs auf der in diesem Beispiel verwendeten Linux-Version können erneut mit dem gescannt werden `rescan-scsi-bus.sh` Befehl. Wenn der Befehl erfolgreich war, sollte jeder LUN-Pfad in der Ausgabe angezeigt werden. Die Ausgabe kann schwer zu interpretieren sein, wenn die Zoning- und `igroup`-Konfiguration korrekt war, sollten viele LUNs scheinen, die eine enthalten `NETAPP` Anbieterzeichenfolge.

```

[root@host1 /]# rescan-scsi-bus.sh
Scanning SCSI subsystem for new devices
Scanning host 0 for SCSI target IDs 0 1 2 3 4 5 6 7, all LUNs
  Scanning for device 0 2 0 0 ...
OLD: Host: scsi0 Channel: 02 Id: 00 Lun: 00
      Vendor: LSI      Model: RAID SAS 6G 0/1  Rev: 2.13
      Type:   Direct-Access                    ANSI SCSI revision: 05
Scanning host 1 for SCSI target IDs 0 1 2 3 4 5 6 7, all LUNs
  Scanning for device 1 0 0 0 ...
OLD: Host: scsi1 Channel: 00 Id: 00 Lun: 00
      Vendor: Optiarc  Model: DVD RW AD-7760H  Rev: 1.41
      Type:   CD-ROM                      ANSI SCSI revision: 05
Scanning host 2 for SCSI target IDs 0 1 2 3 4 5 6 7, all LUNs
Scanning host 3 for SCSI target IDs 0 1 2 3 4 5 6 7, all LUNs
Scanning host 4 for SCSI target IDs 0 1 2 3 4 5 6 7, all LUNs
Scanning host 5 for SCSI target IDs 0 1 2 3 4 5 6 7, all LUNs
Scanning host 6 for SCSI target IDs 0 1 2 3 4 5 6 7, all LUNs
Scanning host 7 for all SCSI target IDs, all LUNs
  Scanning for device 7 0 0 10 ...
OLD: Host: scsi7 Channel: 00 Id: 00 Lun: 10
      Vendor: NETAPP   Model: LUN C-Mode      Rev: 8300
      Type:   Direct-Access                    ANSI SCSI revision: 05
  Scanning for device 7 0 0 11 ...
OLD: Host: scsi7 Channel: 00 Id: 00 Lun: 11
      Vendor: NETAPP   Model: LUN C-Mode      Rev: 8300
      Type:   Direct-Access                    ANSI SCSI revision: 05
  Scanning for device 7 0 0 12 ...
...
OLD: Host: scsi9 Channel: 00 Id: 01 Lun: 18
      Vendor: NETAPP   Model: LUN C-Mode      Rev: 8300
      Type:   Direct-Access                    ANSI SCSI revision: 05
  Scanning for device 9 0 1 19 ...
OLD: Host: scsi9 Channel: 00 Id: 01 Lun: 19
      Vendor: NETAPP   Model: LUN C-Mode      Rev: 8300
      Type:   Direct-Access                    ANSI SCSI revision: 05
0 new or changed device(s) found.
0 remapped or resized device(s) found.
0 device(s) removed.

```

Überprüfen Sie auf Multipath-Geräte

Der LUN-Erkennungsprozess löst auch die Wiederherstellung von Multipath-Geräten aus, der Linux-Multipathing-Treiber hat jedoch bekanntermaßen gelegentlich Probleme. Die Ausgabe von `multipath - ll` Sollte überprüft werden, um sicherzustellen, dass die Ausgabe wie erwartet aussieht. Die folgende Ausgabe zeigt beispielsweise Multipath-Geräte, die mit einem verknüpft sind NETAPP Anbieterzeichenfolge. Jedes Gerät verfügt über vier Pfade, wobei zwei mit einer Priorität von 50 und zwei mit einer Priorität von 10. Obwohl die

genaue Ausgabe mit verschiedenen Versionen von Linux variieren kann, sieht diese Ausgabe wie erwartet aus.



Überprüfen Sie anhand der Dokumentation der Host-Dienstprogramme die Version von Linux, die Sie verwenden `/etc/multipath.conf` Die Einstellungen sind korrekt.

```
[root@host1 /]# multipath -ll
3600a098038303558735d493762504b36 dm-5 NETAPP ,LUN C-Mode
size=10G features='4 queue_if_no_path pg_init_retries 50
retain_attached_hw_handle' hwhandler='1 alua' wp=rw
|+- policy='service-time 0' prio=50 status=active
| |- 7:0:1:4 sdat 66:208 active ready running
| `-- 9:0:1:4 sdbn 68:16 active ready running
`+- policy='service-time 0' prio=10 status=enabled
  |- 7:0:0:4 sdf 8:80 active ready running
  `-- 9:0:0:4 sdz 65:144 active ready running
3600a098038303558735d493762504b2d dm-10 NETAPP ,LUN C-Mode
size=10G features='4 queue_if_no_path pg_init_retries 50
retain_attached_hw_handle' hwhandler='1 alua' wp=rw
|+- policy='service-time 0' prio=50 status=active
| |- 7:0:1:8 sdax 67:16 active ready running
| `-- 9:0:1:8 sdbr 68:80 active ready running
`+- policy='service-time 0' prio=10 status=enabled
  |- 7:0:0:8 sdj 8:144 active ready running
  `-- 9:0:0:8 sdad 65:208 active ready running
...
3600a098038303558735d493762504b37 dm-8 NETAPP ,LUN C-Mode
size=10G features='4 queue_if_no_path pg_init_retries 50
retain_attached_hw_handle' hwhandler='1 alua' wp=rw
|+- policy='service-time 0' prio=50 status=active
| |- 7:0:1:5 sdau 66:224 active ready running
| `-- 9:0:1:5 sdbo 68:32 active ready running
`+- policy='service-time 0' prio=10 status=enabled
  |- 7:0:0:5 sdg 8:96 active ready running
  `-- 9:0:0:5 sdaa 65:160 active ready running
3600a098038303558735d493762504b4b dm-22 NETAPP ,LUN C-Mode
size=10G features='4 queue_if_no_path pg_init_retries 50
retain_attached_hw_handle' hwhandler='1 alua' wp=rw
|+- policy='service-time 0' prio=50 status=active
| |- 7:0:1:19 sdbi 67:192 active ready running
| `-- 9:0:1:19 sdcc 69:0 active ready running
`+- policy='service-time 0' prio=10 status=enabled
  |- 7:0:0:19 sdu 65:64 active ready running
  `-- 9:0:0:19 sdao 66:128 active ready running
```

Reaktivieren Sie die LVM-Volume-Gruppe

Wenn die LVM-LUNs ordnungsgemäß erkannt wurden, wird das angezeigte `vgchange --activate y` Befehl sollte erfolgreich sein. Dies ist ein gutes Beispiel für den Nutzen eines logischen Volume-Managers. Eine Änderung des WWN einer LUN oder auch einer Seriennummer ist unwichtig, da die Metadaten der Volume-Gruppe auf die LUN selbst geschrieben werden.

Das Betriebssystem hat die LUNs gescannt und eine kleine Menge an auf die LUN geschriebenen Daten ermittelt, die sie als physisches Volume des identifizieren `sanvg volumegroup`. Anschließend wurden alle erforderlichen Geräte erstellt. Sie müssen nur die Volume-Gruppe erneut aktivieren.

```
[root@host1 ~]# vgchange --activate y sanvg
Found duplicate PV fpCzdLTuKfy2xDZjailNliJh3TjLUBiT: using
/dev/mapper/3600a098038303558735d493762504b46 not /dev/sdp
Using duplicate PV /dev/mapper/3600a098038303558735d493762504b46 from
subsystem DM, ignoring /dev/sdp
2 logical volume(s) in volume group "sanvg" now active
```

Dateisysteme neu einbinden

Nachdem die Volume-Gruppe wieder aktiviert wurde, können die Dateisysteme mit allen ursprünglichen Daten gemountet werden. Wie bereits erwähnt, sind die Dateisysteme voll funktionsfähig, selbst wenn die Datenreplikation in der Back-Gruppe weiterhin aktiv ist.

```

[root@host1 ~]# mount /orabin
[root@host1 ~]# mount /backups
[root@host1 ~]# df -k

```

Filesystem	1K-blocks	Used	Available	Use%	
Mounted on					
/dev/mapper/rhel-root	52403200	8837100	43566100	17%	/
devtmpfs	65882776	0	65882776	0%	/dev
tmpfs	6291456	84	6291372	1%	
/dev/shm					
tmpfs	65898668	9884	65888784	1%	/run
tmpfs	65898668	0	65898668	0%	
/sys/fs/cgroup					
/dev/sda1	505580	224828	280752	45%	/boot
fas8060-nfs-public:/install	199229440	119368256	79861184	60%	
/install					
fas8040-nfs-routable:/snapomatic	9961472	30528	9930944	1%	
/snapomatic					
tmpfs	13179736	16	13179720	1%	
/run/user/42					
tmpfs	13179736	0	13179736	0%	
/run/user/0					
/dev/mapper/sanvg-lvorabin	20961280	12357456	8603824	59%	
/orabin					
/dev/mapper/sanvg-lvbackups	73364480	62947536	10416944	86%	
/backups					

Neuscannen für ASM-Geräte

Die ASMLib-Geräte sollten beim erneuten Scannen der SCSI-Geräte neu erkannt worden sein. Die Wiedererkennung kann online überprüft werden, indem ASMLib neu gestartet und anschließend die Datenträger gescannt werden.



Dieser Schritt ist nur für ASM-Konfigurationen relevant, in denen ASMLib verwendet wird.

Achtung: Wenn ASMLib nicht verwendet wird, ist die `/dev/mapper` Geräte sollten automatisch neu erstellt worden sein. Die Berechtigungen sind jedoch möglicherweise nicht korrekt. Sie müssen spezielle Berechtigungen für die zugrunde liegenden Geräte für ASM festlegen, wenn ASMLib nicht vorhanden ist. Dies wird in der Regel durch spezielle Einträge in entweder der erreicht `/etc/multipath.conf` Oder `udev` Regeln oder möglicherweise in beiden Regelsätzen. Diese Dateien müssen möglicherweise aktualisiert werden, um Änderungen in der Umgebung in Bezug auf WWNs oder Seriennummern widerzuspiegeln, um sicherzustellen, dass die ASM-Geräte weiterhin über die richtigen Berechtigungen verfügen.

In diesem Beispiel werden beim Neustart von ASMLib und beim Scannen nach Festplatten die gleichen 10 ASM-LUNs wie in der ursprünglichen Umgebung angezeigt.

```
[root@host1 ~]# oracleasm exit
Unmounting ASMLib driver filesystem: /dev/oracleasm
Unloading module "oracleasm": oracleasm
[root@host1 ~]# oracleasm init
Loading module "oracleasm": oracleasm
Configuring "oracleasm" to use device physical block size
Mounting ASMLib driver filesystem: /dev/oracleasm
[root@host1 ~]# oracleasm scandisks
Reloading disk partitions: done
Cleaning any stale ASM disks...
Scanning system for ASM disks...
Instantiating disk "ASM0"
Instantiating disk "ASM1"
Instantiating disk "ASM2"
Instantiating disk "ASM3"
Instantiating disk "ASM4"
Instantiating disk "ASM5"
Instantiating disk "ASM6"
Instantiating disk "ASM7"
Instantiating disk "ASM8"
Instantiating disk "ASM9"
```

Starten Sie die Grid-Services neu

Da die LVM- und ASM-Geräte jetzt online und verfügbar sind, können die Grid-Dienste neu gestartet werden.

```
[root@host1 ~]# cd /orabin/product/12.1.0/grid/bin
[root@host1 bin]# ./crsctl start has
```

Datenbank neu starten

Nach dem Neustart der Netzdienste kann die Datenbank gestartet werden. Möglicherweise müssen Sie einige Minuten warten, bis die ASM-Dienste vollständig verfügbar sind, bevor Sie versuchen, die Datenbank zu starten.

```
[root@host1 bin]# su - oracle
[oracle@host1 ~]$ . oraenv
ORACLE_SID = [oracle] ? FLIDB
The Oracle base has been set to /orabin
[oracle@host1 ~]$ sqlplus / as sysdba
SQL*Plus: Release 12.1.0.2.0
Copyright (c) 1982, 2014, Oracle. All rights reserved.
Connected to an idle instance.
SQL> startup
ORACLE instance started.
Total System Global Area 3221225472 bytes
Fixed Size 4502416 bytes
Variable Size 1207962736 bytes
Database Buffers 1996488704 bytes
Redo Buffers 12271616 bytes
Database mounted.
Database opened.
SQL>
```

Abschluss

Aus Host-Sicht ist die Migration abgeschlossen, aber I/O wird weiterhin vom fremden Array bedient, bis die Importbeziehungen gelöscht werden.

Bevor Sie die Beziehungen löschen, müssen Sie bestätigen, dass der Migrationsprozess für alle LUNs abgeschlossen ist.

```

Cluster01::*> lun import show -vserver vserver1 -fields foreign-
disk,path,operational-state
vserver    foreign-disk path                operational-state
-----
vserver1 800DT$HuVWB/ /vol/new_asm/LUN4 completed
vserver1 800DT$HuVWBW /vol/new_asm/LUN0 completed
vserver1 800DT$HuVWBX /vol/new_asm/LUN1 completed
vserver1 800DT$HuVWBZ /vol/new_asm/LUN2 completed
vserver1 800DT$HuVWBa /vol/new_asm/LUN5 completed
vserver1 800DT$HuVWBb /vol/new_asm/LUN6 completed
vserver1 800DT$HuVWBc /vol/new_asm/LUN7 completed
vserver1 800DT$HuVWBd /vol/new_asm/LUN8 completed
vserver1 800DT$HuVWB e /vol/new_asm/LUN9 completed
vserver1 800DT$HuVWBf /vol/new_lvm/LUN0 completed
vserver1 800DT$HuVWBg /vol/new_lvm/LUN1 completed
vserver1 800DT$HuVWBh /vol/new_lvm/LUN2 completed
vserver1 800DT$HuVWB i /vol/new_lvm/LUN3 completed
vserver1 800DT$HuVWBj /vol/new_lvm/LUN4 completed
vserver1 800DT$HuVWBk /vol/new_lvm/LUN5 completed
vserver1 800DT$HuVWB l /vol/new_lvm/LUN6 completed
vserver1 800DT$HuVWBm /vol/new_lvm/LUN7 completed
vserver1 800DT$HuVWBn /vol/new_lvm/LUN8 completed
vserver1 800DT$HuVWB o /vol/new_lvm/LUN9 completed
20 entries were displayed.

```

Importbeziehungen löschen

Löschen Sie nach Abschluss des Migrationsprozesses die Migrationsbeziehung. Anschließend wird die I/O ausschließlich von den Laufwerken auf ONTAP bedient.

```

Cluster01::*> lun import delete -vserver vserver1 -path /vol/new_asm/LUN0
Cluster01::*> lun import delete -vserver vserver1 -path /vol/new_asm/LUN1
...
Cluster01::*> lun import delete -vserver vserver1 -path /vol/new_lvm/LUN8
Cluster01::*> lun import delete -vserver vserver1 -path /vol/new_lvm/LUN9

```

Registrierung ausländischer LUNs aufheben

Ändern Sie schließlich die Festplatte, um die zu entfernen `is-foreign` Bezeichnung.

```

Cluster01::*> storage disk modify {-serial-number 800DT$HuVWBW} -is
-foreign false
Cluster01::*> storage disk modify {-serial-number 800DT$HuVWBX} -is
-foreign false
...
Cluster01::*> storage disk modify {-serial-number 800DT$HuVWBn} -is
-foreign false
Cluster01::*> storage disk modify {-serial-number 800DT$HuVWBo} -is
-foreign false
Cluster01::*>

```

Protokollkonvertierung

Das Ändern des Protokolls für den Zugriff auf eine LUN ist eine gängige Anforderung.

In einigen Fällen ist die Migration der Daten in die Cloud Teil einer Gesamtstrategie. TCP/IP ist das Protokoll der Cloud, und der Wechsel von FC zu iSCSI ermöglicht eine einfachere Migration in verschiedene Cloud-Umgebungen. In anderen Fällen kann iSCSI wünschenswert sein, die gesunkenen Kosten eines IP SAN zu nutzen. Gelegentlich kann eine Migration ein anderes Protokoll als temporäre Maßnahme verwenden. Wenn beispielsweise ein fremdes Array und ONTAP-basierte LUNs nicht auf denselben HBAs koexistieren können, können Sie iSCSI-LUNs verwenden, die lang genug sind, um Daten vom alten Array zu kopieren. Nachdem die alten LUNs aus dem System entfernt wurden, können Sie sie wieder zu FC konvertieren.

Das folgende Verfahren zeigt die Konvertierung von FC zu iSCSI, jedoch gelten die allgemeinen Prinzipien für eine umgekehrte iSCSI- zu FC-Konvertierung.

Installieren Sie den iSCSI-Initiator

Die meisten Betriebssysteme enthalten standardmäßig einen Software-iSCSI-Initiator, aber wenn dieser nicht enthalten ist, kann er problemlos installiert werden.

```

[root@host1 /]# yum install -y iscsi-initiator-utils
Loaded plugins: langpacks, product-id, search-disabled-repos,
subscription-
                : manager
Resolving Dependencies
--> Running transaction check
--> Package iscsi-initiator-utils.x86_64 0:6.2.0.873-32.el7 will be
updated
--> Processing Dependency: iscsi-initiator-utils = 6.2.0.873-32.el7 for
package: iscsi-initiator-utils-iscsiuio-6.2.0.873-32.el7.x86_64
--> Package iscsi-initiator-utils.x86_64 0:6.2.0.873-32.0.2.el7 will be
an update
--> Running transaction check
--> Package iscsi-initiator-utils-iscsiuio.x86_64 0:6.2.0.873-32.el7 will
be updated
--> Package iscsi-initiator-utils-iscsiuio.x86_64 0:6.2.0.873-32.0.2.el7

```

will be an update

--> Finished Dependency Resolution

Dependencies Resolved

=====
===

Package	Arch	Version	Repository
---------	------	---------	------------

=====
===

Updating:

iscsi-initiator-utils	x86_64	6.2.0.873-32.0.2.el7	ol7_latest	416 k
-----------------------	--------	----------------------	------------	-------

Updating for dependencies:

iscsi-initiator-utils-iscsiuio	x86_64	6.2.0.873-32.0.2.el7	ol7_latest	84 k
--------------------------------	--------	----------------------	------------	------

Transaction Summary

=====
===

Upgrade 1 Package (+1 Dependent package)

Total download size: 501 k

Downloading packages:

No Presto metadata available for ol7_latest

(1/2): iscsi-initiator-utils-6.2.0.873-32.0.2.el7.x86_6 | 416 kB 00:00

(2/2): iscsi-initiator-utils-iscsiuio-6.2.0.873-32.0.2. | 84 kB 00:00

Total 2.8 MB/s | 501 kB

00:00Cluster01

Running transaction check

Running transaction test

Transaction test succeeded

Running transaction

Updating : iscsi-initiator-utils-iscsiuio-6.2.0.873-32.0.2.el7.x86
1/4

Updating : iscsi-initiator-utils-6.2.0.873-32.0.2.el7.x86_64
2/4

Cleanup : iscsi-initiator-utils-iscsiuio-6.2.0.873-32.el7.x86_64
3/4

Cleanup : iscsi-initiator-utils-6.2.0.873-32.el7.x86_64
4/4

rhel-7-server-eus-rpms/7Server/x86_64/productid | 1.7 kB 00:00

rhel-7-server-rpms/7Server/x86_64/productid | 1.7 kB 00:00

Verifying : iscsi-initiator-utils-6.2.0.873-32.0.2.el7.x86_64
1/4

Verifying : iscsi-initiator-utils-iscsiuio-6.2.0.873-32.0.2.el7.x86
2/4

```
Verifying   : iscsi-initiator-utils-iscsiuio-6.2.0.873-32.el7.x86_64
3/4
Verifying   : iscsi-initiator-utils-6.2.0.873-32.el7.x86_64
4/4
Updated:
  iscsi-initiator-utils.x86_64 0:6.2.0.873-32.0.2.el7
Dependency Updated:
  iscsi-initiator-utils-iscsiuio.x86_64 0:6.2.0.873-32.0.2.el7
Complete!
[root@host1 /]#
```

Identifizieren Sie den iSCSI-Initiatornamen

Während der Installation wird ein eindeutiger iSCSI-Initiatorname generiert. Unter Linux befindet sie sich im `/etc/iscsi/initiatorname.iscsi` Datei: Dieser Name dient zur Identifizierung des Hosts auf dem IP-SAN.

```
[root@host1 /]# cat /etc/iscsi/initiatorname.iscsi
InitiatorName=iqn.1992-05.com.redhat:497bd66ca0
```

Erstellen Sie eine neue Initiatorgruppe

Eine Initiatorgruppe (Initiatorgruppe) ist Teil der ONTAP LUN-Masking-Architektur. Auf eine neu erstellte LUN kann nur dann zugegriffen werden, wenn einem Host der erste Zugriff gewährt wurde. Hierzu wird eine Initiatorgruppe erstellt, die entweder die FC-WWNs oder die iSCSI-Initiatornamen enthält, die Zugriff erfordern.

In diesem Beispiel wird eine Initiatorgruppe erstellt, die den iSCSI-Initiator des Linux Hosts enthält.

```
Cluster01::*> igroup create -igroup linuxiscsi -protocol iscsi -ostype
linux -initiator iqn.1994-05.com.redhat:497bd66ca0
```

Fahren Sie die Umgebung herunter

Vor dem Ändern des LUN-Protokolls müssen die LUNs vollständig stillgelegt werden. Jede Datenbank auf einer der zu konvertierenden LUNs muss heruntergefahren, die File-Systeme deaktiviert und die Volume-Gruppen deaktiviert werden. Wenn ASM verwendet wird, stellen Sie sicher, dass die ASM-Laufwerksgruppe getrennt ist und fahren Sie alle Netzdienste herunter.

LUN-Zuordnungen zum FC-Netzwerk aufheben

Nachdem die LUNs vollständig stillgelegt sind, entfernen Sie die Zuordnungen von der ursprünglichen FC-Initiatorgruppe.

```

Cluster01::*> lun unmap -vserver vserver1 -path /vol/new_asm/LUN0 -igroup
linuxhost
Cluster01::*> lun unmap -vserver vserver1 -path /vol/new_asm/LUN1 -igroup
linuxhost
...
Cluster01::*> lun unmap -vserver vserver1 -path /vol/new_lvm/LUN8 -igroup
linuxhost
Cluster01::*> lun unmap -vserver vserver1 -path /vol/new_lvm/LUN9 -igroup
linuxhost

```

LUN-Zuordnung zum IP-Netzwerk neu

Gewähren Sie der neuen iSCSI-basierten Initiatorgruppe Zugriff auf jede LUN.

```

Cluster01::*> lun map -vserver vserver1 -path /vol/new_asm/LUN0 -igroup
linuxiscsi
Cluster01::*> lun map -vserver vserver1 -path /vol/new_asm/LUN1 -igroup
linuxiscsi
...
Cluster01::*> lun map -vserver vserver1 -path /vol/new_lvm/LUN8 -igroup
linuxiscsi
Cluster01::*> lun map -vserver vserver1 -path /vol/new_lvm/LUN9 -igroup
linuxiscsi
Cluster01::*>

```

iSCSI-Ziele erkennen

Die iSCSI-Erkennung besteht aus zwei Phasen. Zum einen werden die Ziele ermittelt. Dies ist nicht dasselbe wie beim Erkennen einer LUN. Der `iscsiadm` Der unten abgebildete Befehl prüft die vom angegebene Portalgruppe `-p argument` Und speichert eine Liste aller IP-Adressen und Ports, die iSCSI-Dienste anbieten. In diesem Fall gibt es vier IP-Adressen, die iSCSI-Dienste auf dem Standardport 3260 haben.



Dieser Befehl kann mehrere Minuten dauern, wenn eine der Ziel-IP-Adressen nicht erreicht werden kann.

```

[root@host1 ~]# iscsiadm -m discovery -t st -p fas8060-iscsi-public1
10.63.147.197:3260,1033 iqn.1992-
08.com.netapp:sn.807615e9ef6111e5a5ae90e2ba5b9464:vs.3
10.63.147.198:3260,1034 iqn.1992-
08.com.netapp:sn.807615e9ef6111e5a5ae90e2ba5b9464:vs.3
172.20.108.203:3260,1030 iqn.1992-
08.com.netapp:sn.807615e9ef6111e5a5ae90e2ba5b9464:vs.3
172.20.108.202:3260,1029 iqn.1992-
08.com.netapp:sn.807615e9ef6111e5a5ae90e2ba5b9464:vs.3

```

ISCSI-LUNs erkennen

Nachdem die iSCSI-Ziele erkannt wurden, starten Sie den iSCSI-Dienst neu, um die verfügbaren iSCSI-LUNs zu ermitteln und zugehörige Geräte wie Multipath- oder ASMLib-Geräte zu erstellen.

```
[root@host1 ~]# service iscsi restart
Redirecting to /bin/systemctl restart iscsi.service
```

Starten Sie die Umgebung neu

Starten Sie die Umgebung neu, indem Sie Volume-Gruppen erneut aktivieren, Dateisysteme neu mounten, RAC-Dienste neu starten usw. Als Vorsichtsmaßnahme empfiehlt NetApp, den Server nach Abschluss des Konvertierungsprozesses neu zu starten, um sicherzustellen, dass alle Konfigurationsdateien korrekt sind und alle veralteten Geräte entfernt werden.

Achtung: Bevor Sie einen Host neu starten, stellen Sie sicher, dass alle Einträge in `/etc/fstab` diese Referenz migrierte SAN-Ressourcen werden kommentiert. Wenn dieser Schritt nicht durchgeführt wird und Probleme mit dem LUN-Zugriff auftreten, kann es zu einem Betriebssystem kommen, das nicht gebootet wird. Dieses Problem beschädigt die Daten nicht. Es kann jedoch sehr unbequem sein, in den Rettungsmodus oder einen ähnlichen Modus zu starten und zu korrigieren `/etc/fstab` Damit das Betriebssystem gestartet werden kann, um die Fehlerbehebung zu ermöglichen.

Beispielskripts

Die vorgestellten Skripte werden als Beispiele für das Skript verschiedener Betriebssystem- und Datenbankaufgaben bereitgestellt. Sie werden wie sie sind geliefert. Wenn für eine bestimmte Vorgehensweise Support erforderlich ist, wenden Sie sich an NetApp oder einen NetApp Reseller.

Datenbank wird heruntergefahren

Das folgende Perl-Skript nimmt ein einziges Argument der Oracle SID und fährt eine Datenbank herunter. Sie kann als Oracle-Benutzer oder als root ausgeführt werden.

```

#!/usr/bin/perl
use strict;
use warnings;
my $oraclesid=$ARGV[0];
my $oracleuser='oracle';
my @out;
my $uid=$<;
if ($uid == 0) {
@out=`su - $oracleuser -c '. oraenv << EOF1
77 Migration of Oracle Databases to NetApp Storage Systems © 2021 NetApp,
Inc. All rights reserved
$oraclesid
EOF1
sqlplus / as sysdba << EOF2
shutdown immediate;
EOF2
`
`;}
else {
@out=`. oraenv << EOF1
$oraclesid
EOF4
sqlplus / as sysdba << EOF2
shutdown immediate;
EOF2
`;};
print @out;
if ("@out" =~ /ORACLE instance shut down/) {
print "$oraclesid shut down\n";
exit 0;}
elsif ("@out" =~ /Connected to an idle instance/) {
print "$oraclesid already shut down\n";
exit 0;}
else {
print "$oraclesid failed to shut down\n";
exit 1;}

```

Starten der Datenbank

Das folgende Perl-Skript nimmt ein einziges Argument der Oracle SID und fährt eine Datenbank herunter. Sie kann als Oracle-Benutzer oder als root ausgeführt werden.

```

#!/usr/bin/perl
use strict;
use warnings;
my $oraclesid=$ARGV[0];
my $oracleuser='oracle';
my @out;
my $uid=$<;
if ($uid == 0) {
@out=`su - $oracleuser -c '. oraenv << EOF1
$oraclesid
EOF1
sqlplus / as sysdba << EOF2
startup;
EOF2
`
`;}
else {
@out=`. oraenv << EOF3
$oraclesid
EOF1
sqlplus / as sysdba << EOF2
startup;
EOF2
`;};
print @out;
if ("@out" =~ /Database opened/) {
print "$oraclesid started\n";
exit 0;}
elsif ("@out" =~ /cannot start already-running ORACLE/) {
print "$oraclesid already started\n";
exit 1;}
else {
78 Migration of Oracle Databases to NetApp Storage Systems © 2021 NetApp,
Inc. All rights reserved
print "$oraclesid failed to start\n";
exit 1;}

```

Konvertieren Sie das Dateisystem in schreibgeschützt

Das folgende Skript nimmt ein Dateisystemargument an und versucht, es als schreibgeschützt zu entfernen und wieder zu mounten. Dies ist bei Migrationsprozessen sinnvoll, bei denen ein Dateisystem für die Datenreplikation verfügbar gehalten werden muss und dennoch vor versehentlichen Schäden geschützt werden muss.

```

#!/usr/bin/perl
use strict;
#use warnings;
my $filesystem=$ARGV[0];
my @out=`umount '$filesystem'`;
if ($? == 0) {
    print "$filesystem unmounted\n";
    @out = `mount -o ro '$filesystem'`;
    if ($? == 0) {
        print "$filesystem mounted read-only\n";
        exit 0;}}
else {
    print "Unable to unmount $filesystem\n";
    exit 1;}
print @out;

```

Ersetzen Sie das Dateisystem

Das folgende Skriptbeispiel wird verwendet, um ein Dateisystem durch ein anderes zu ersetzen. Da die Datei `/etc/fstab` bearbeitet wird, muss sie als root ausgeführt werden. Es akzeptiert ein einzelnes kommagetrenntes Argument des alten und des neuen Dateisystems.

1. Führen Sie zum Ersetzen des Dateisystems das folgende Skript aus:

```

#!/usr/bin/perl
use strict;
#use warnings;
my $oldfs;
my $newfs;
my @oldfstab;
my @newfstab;
my $source;
my $mountpoint;
my $leftover;
my $oldfstabentry='';
my $newfstabentry='';
my $migratedfstabentry='';
($oldfs, $newfs) = split (',', $ARGV[0]);
open(my $filehandle, '<', '/etc/fstab') or die "Could not open
/etc/fstab\n";
while (my $line = <$filehandle>) {
    chomp $line;
    ($source, $mountpoint, $leftover) = split(/[ , ]/, $line, 3);
    if ($mountpoint eq $oldfs) {
        $oldfstabentry = "#Removed by swap script $source $oldfs $leftover";}

```

```

elseif ($mountpoint eq $newfs) {
    $newfstabentry = "#Removed by swap script $source $newfs $leftover";
    $migratedfstabentry = "$source $oldfs $leftover";}
else {
    push (@newfstab, "$line\n")}}
79 Migration of Oracle Databases to NetApp Storage Systems © 2021
NetApp, Inc. All rights reserved
push (@newfstab, "$oldfstabentry\n");
push (@newfstab, "$newfstabentry\n");
push (@newfstab, "$migratedfstabentry\n");
close($filehandle);
if ($oldfstabentry eq ''){
    die "Could not find $oldfs in /etc/fstab\n";}
if ($newfstabentry eq ''){
    die "Could not find $newfs in /etc/fstab\n";}
my @out=`umount '$newfs'`;
if ($? == 0) {
    print "$newfs unmounted\n";}
else {
    print "Unable to unmount $newfs\n";
    exit 1;}
@out=`umount '$oldfs'`;
if ($? == 0) {
    print "$oldfs unmounted\n";}
else {
    print "Unable to unmount $oldfs\n";
    exit 1;}
system("cp /etc/fstab /etc/fstab.bak");
open ($filehandle, ">", '/etc/fstab') or die "Could not open /etc/fstab
for writing\n";
for my $line (@newfstab) {
    print $filehandle $line;}
close($filehandle);
@out=`mount '$oldfs'`;
if ($? == 0) {
    print "Mounted updated $oldfs\n";
    exit 0;}
else{
    print "Unable to mount updated $oldfs\n";
    exit 1;}
exit 0;

```

Nehmen Sie als Beispiel für die Verwendung dieses Skripts an, dass die Daten in enthalten sind /oradata Wird auf migriert /neworadata Und /logs Wird auf migriert /newlogs. Eine der einfachsten Methoden, um diese Aufgabe durchzuführen, besteht darin, das neue Gerät mit einem einfachen Dateikopiervorgang wieder in den ursprünglichen Bereitstellungspunkt zu verschieben.

2. Gehen Sie davon aus, dass die alten und die neuen Dateisysteme im vorhanden sind /etc/fstab Datei wie folgt:

```
cluster01:/vol_oradata /oradata nfs rw,bg,vers=3,rsize=65536,wsiz=65536
0 0
cluster01:/vol_logs /logs nfs rw,bg,vers=3,rsize=65536,wsiz=65536 0 0
cluster01:/vol_neworadata /neworadata nfs
rw,bg,vers=3,rsize=65536,wsiz=65536 0 0
cluster01:/vol_newlogs /newlogs nfs rw,bg,vers=3,rsize=65536,wsiz=65536
0 0
```

3. Wenn dieses Skript ausgeführt wird, wird das aktuelle Dateisystem abgehängt und durch das neue ersetzt:

```
[root@jpsc3 scripts]# ./swap.fs.pl /oradata,/neworadata
/neworadata unmounted
/oradata unmounted
Mounted updated /oradata
[root@jpsc3 scripts]# ./swap.fs.pl /logs,/newlogs
/newlogs unmounted
/logs unmounted
Mounted updated /logs
```

4. Das Skript aktualisiert auch die /etc/fstab Entsprechende Datei erstellen. Im hier gezeigten Beispiel sind folgende Änderungen enthalten:

```
#Removed by swap script cluster01:/vol_oradata /oradata nfs
rw,bg,vers=3,rsize=65536,wsiz=65536 0 0
#Removed by swap script cluster01:/vol_neworadata /neworadata nfs
rw,bg,vers=3,rsize=65536,wsiz=65536 0 0
cluster01:/vol_neworadata /oradata nfs
rw,bg,vers=3,rsize=65536,wsiz=65536 0 0
#Removed by swap script cluster01:/vol_logs /logs nfs
rw,bg,vers=3,rsize=65536,wsiz=65536 0 0
#Removed by swap script cluster01:/vol_newlogs /newlogs nfs
rw,bg,vers=3,rsize=65536,wsiz=65536 0 0
cluster01:/vol_newlogs /logs nfs rw,bg,vers=3,rsize=65536,wsiz=65536 0
0
```

Automatisierte Datenbankmigration

Dieses Beispiel zeigt, wie Skripts zum Herunterfahren, Starten und Ersetzen von Dateisystemen genutzt werden können, um eine Migration vollständig zu automatisieren.

```

#!/usr/bin/perl
use strict;
#use warnings;
my $oraclesid=$ARGV[0];
my @oldfs;
my @newfs;
my $x=1;
while ($x < scalar(@ARGV)) {
    ($oldfs[$x-1], $newfs[$x-1]) = split (',', $ARGV[$x]);
    $x+=1;}
my @out=`./dbshut.pl '$oraclesid'`;
print @out;
if ($? ne 0) {
    print "Failed to shut down database\n";
    exit 0;}
$x=0;
while ($x < scalar(@oldfs)) {
    my @out=`./mk.fs.readonly.pl '$oldfs[$x]'`;
    if ($? ne 0) {
        print "Failed to make filesystem $oldfs[$x] readonly\n";
        exit 0;}
    $x+=1;}
$x=0;
while ($x < scalar(@oldfs)) {
    my @out=`rsync -rlpogt --stats --progress --exclude='.snapshot'
'$oldfs[$x]/' '/$newfs[$x]/'`;
    print @out;
    if ($? ne 0) {
        print "Failed to copy filesystem $oldfs[$x] to $newfs[$x]\n";
        exit 0;}
    else {
        print "Succesfully replicated filesystem $oldfs[$x] to
$newfs[$x]\n";}
    $x+=1;}
$x=0;
while ($x < scalar(@oldfs)) {
    print "swap $x $oldfs[$x] $newfs[$x]\n";
    my @out=`./swap.fs.pl '$oldfs[$x],$newfs[$x]'`;
    print @out;
    if ($? ne 0) {
        print "Failed to swap filesystem $oldfs[$x] for $newfs[$x]\n";
        exit 1;}
    else {
        print "Swapped filesystem $oldfs[$x] for $newfs[$x]\n";}
    $x+=1;}
my @out=`./dbstart.pl '$oraclesid'`;

```

```
print @out;
```

Dateispeicherorte anzeigen

Dieses Skript sammelt eine Reihe wichtiger Datenbankparameter und druckt sie in einem leicht lesbaren Format aus. Dieses Skript kann bei der Überprüfung von Datenlayouts nützlich sein. Darüber hinaus kann das Skript für andere Zwecke geändert werden.

```
#!/usr/bin/perl
#use strict;
#use warnings;
my $oraclesid=$ARGV[0];
my $oracleuser='oracle';
my @out;
sub dosql{
    my $command = @_ [0];
    my @lines;
    my $uid=$<;
    if ($uid == 0) {
        @lines=`su - $oracleuser -c "export ORAENV_ASK=NO;export
ORACLE_SID=$oraclesid;. oraenv -s << EOF1
EOF1
sqlplus -S / as sysdba << EOF2
set heading off
$command
EOF2
"
        `; }
    else {
        $command=~s/\\\\\\\\\\\\\\\\/\\/g;
        @lines=`export ORAENV_ASK=NO;export ORACLE_SID=$oraclesid;. oraenv
-s << EOF1
EOF1
sqlplus -S / as sysdba << EOF2
set heading off
$command
EOF2
        `; };
    return @lines}
print "\n";
@out=dosql('select name from v\\\\\\\\\\\\$datafile;');
print "$oraclesid datafiles:\n";
for $line (@out) {
    chomp($line);
    if (length($line)>0) {print "$line\n";}}
print "\n";
```

```

@out=dosql('select member from v\\\\\\\\$logfile;');
print "$oraclesid redo logs:\n";
for $line (@out) {
    chomp($line);
    if (length($line)>0) {print "$line\n";}}
print "\n";
@out=dosql('select name from v\\\\\\\\$tempfile;');
print "$oraclesid temp datafiles:\n";
for $line (@out) {
    chomp($line);
    if (length($line)>0) {print "$line\n";}}
print "\n";
@out=dosql('show parameter spfile;');
print "$oraclesid spfile\n";
for $line (@out) {
    chomp($line);
    if (length($line)>0) {print "$line\n";}}
print "\n";
@out=dosql('select name||\'' \|'\|value from v\\\\\\\\$parameter where
isdefault=\''FALSE\'';');
print "$oraclesid key parameters\n";
for $line (@out) {
    chomp($line);
    if ($line =~ /control_files/) {print "$line\n";}
    if ($line =~ /db_create/) {print "$line\n";}
    if ($line =~ /db_file_name_convert/) {print "$line\n";}
    if ($line =~ /log_archive_dest/) {print "$line\n";}}
    if ($line =~ /log_file_name_convert/) {print "$line\n";}
    if ($line =~ /pdb_file_name_convert/) {print "$line\n";}
    if ($line =~ /spfile/) {print "$line\n";}
print "\n";

```

Bereinigung der ASM-Migration

```

#!/usr/bin/perl
#use strict;
#use warnings;
my $oraclesid=$ARGV[0];
my $oracleuser='oracle';
my @out;
sub dosql{
    my $command = @_ [0];
    my @lines;
    my $uid=$<;
    if ($uid == 0) {

```

```

@lines=`su - $oracleuser -c "export ORAENV_ASK=NO;export
ORACLE_SID=$oraclesid;. oraenv -s << EOF1
EOF1
sqlplus -S / as sysdba << EOF2
set heading off
$command
EOF2
"
    `;}
    else {
        $command=~s/\\\\\\\\\\\\\\\\/\\/g;
        @lines=`export ORAENV_ASK=NO;export ORACLE_SID=$oraclesid;. oraenv
-s << EOF1
EOF1
sqlplus -S / as sysdba << EOF2
set heading off
$command
EOF2
    `;}
return @lines}
print "\n";
@out=dosql('select name from v\\\\\\\\$datafile;');
print @out;
print "shutdown immediate;\n";
print "startup mount;\n";
print "\n";
for $line (@out) {
    if (length($line) > 1) {
        chomp($line);
        ($first, $second,$third,$fourth)=split('_', $line);
        $fourth =~ s/^TS-//;
        $newname=lc("$fourth.dbf");
        $path2file=$line;
        $path2file=~ /(^.*\\.\/)/;
        print "host mv $line $1$newname\n";}}
print "\n";
for $line (@out) {
    if (length($line) > 1) {
        chomp($line);
        ($first, $second,$third,$fourth)=split('_', $line);
        $fourth =~ s/^TS-//;
        $newname=lc("$fourth.dbf");
        $path2file=$line;
        $path2file=~ /(^.*\\.\/)/;
        print "alter database rename file '$line' to
'$1$newname';\n";}}

```

```
print "alter database open;\n";  
print "\n";
```

Namenskonvertierung von ASM in Dateisystem

```

set serveroutput on;
set wrap off;
declare
    cursor df is select file#, name from v$datafile;
    cursor tf is select file#, name from v$tempfile;
    cursor lf is select member from v$logfile;
    firstline boolean := true;
begin
    dbms_output.put_line(CHR(13));
    dbms_output.put_line('Parameters for log file conversion:');
    dbms_output.put_line(CHR(13));
    dbms_output.put('*.log_file_name_convert = ');
    for lfrec in lf loop
        if (firstline = true) then
            dbms_output.put('''' || lfrec.member || ''', ');
            dbms_output.put(''''/NEW_PATH/' ||
regexp_replace(lfrec.member, '^.*./', '') || ''');
        else
            dbms_output.put(', ''' || lfrec.member || ''', ');
            dbms_output.put(''''/NEW_PATH/' ||
regexp_replace(lfrec.member, '^.*./', '') || ''');
        end if;
        firstline:=false;
    end loop;
    dbms_output.put_line(CHR(13));
    dbms_output.put_line(CHR(13));
    dbms_output.put_line('rman duplication script:');
    dbms_output.put_line(CHR(13));
    dbms_output.put_line('run');
    dbms_output.put_line('{');
    for dfrec in df loop
        dbms_output.put_line('set newname for datafile ' ||
dfrec.file# || ' to ''' || dfrec.name || ''';');
    end loop;
    for tfrec in tf loop
        dbms_output.put_line('set newname for tempfile ' ||
tfrec.file# || ' to ''' || tfrec.name || ''';');
    end loop;
    dbms_output.put_line('duplicate target database for standby backup
location INSERT_PATH_HERE;');
    dbms_output.put_line('}');
end;
/

```

Wiedergabe von Protokollen in der Datenbank

Dieses Skript akzeptiert ein einzelnes Argument einer Oracle SID für eine Datenbank, die sich im Mount-Modus befindet, und versucht, alle derzeit verfügbaren Archivprotokolle wiederzugeben.

```
#!/usr/bin/perl
use strict;
my $oraclesid=$ARGV[0];
my $oracleuser='oracle';
84 Migration of Oracle Databases to NetApp Storage Systems © 2021 NetApp,
Inc. All rights reserved
my $uid = $<;
my @out;
if ($uid == 0) {
@out=`su - $oracleuser -c '. oraenv << EOF1
$oraclesid
EOF1
sqlplus / as sysdba << EOF2
recover database until cancel;
auto
EOF2
`;
}
else {
@out=`. oraenv << EOF1
$oraclesid
EOF1
sqlplus / as sysdba << EOF2
recover database until cancel;
auto
EOF2
`;
}
print @out;
```

Wiedergabe von Protokollen in der Standby-Datenbank

Dieses Skript ist identisch mit dem vorhergehenden Skript, außer dass es für eine Standby-Datenbank konzipiert ist.

```
#!/usr/bin/perl
use strict;
my $oraclesid=$ARGV[0];
my $oracleuser='oracle';
my $uid = $<;
my @out;
if ($uid == 0) {
@out=`su - $oracleuser -c '. oraenv << EOF1
$oraclesid
EOF1
sqlplus / as sysdba << EOF2
recover standby database until cancel;
auto
EOF2
`
};}
else {
@out=`. oraenv << EOF1
$oraclesid
EOF1
sqlplus / as sysdba << EOF2
recover standby database until cancel;
auto
EOF2
`;
}
print @out;
```

Copyright-Informationen

Copyright © 2026 NetApp. Alle Rechte vorbehalten. Gedruckt in den USA. Dieses urheberrechtlich geschützte Dokument darf ohne die vorherige schriftliche Genehmigung des Urheberrechtinhabers in keiner Form und durch keine Mittel – weder grafische noch elektronische oder mechanische, einschließlich Fotokopieren, Aufnehmen oder Speichern in einem elektronischen Abrufsystem – auch nicht in Teilen, vervielfältigt werden.

Software, die von urheberrechtlich geschütztem NetApp Material abgeleitet wird, unterliegt der folgenden Lizenz und dem folgenden Haftungsausschluss:

DIE VORLIEGENDE SOFTWARE WIRD IN DER VORLIEGENDEN FORM VON NETAPP ZUR VERFÜGUNG GESTELLT, D. H. OHNE JEGLICHE EXPLIZITE ODER IMPLIZITE GEWÄHRLEISTUNG, EINSCHLIESSLICH, JEDOCH NICHT BESCHRÄNKT AUF DIE STILLSCHWEIGENDE GEWÄHRLEISTUNG DER MARKTGÄNGIGKEIT UND EIGNUNG FÜR EINEN BESTIMMTEN ZWECK, DIE HIERMIT AUSGESCHLOSSEN WERDEN. NETAPP ÜBERNIMMT KEINERLEI HAFTUNG FÜR DIREKTE, INDIREKTE, ZUFÄLLIGE, BESONDERE, BEISPIELHAFT SCHÄDEN ODER FOLGESCHÄDEN (EINSCHLIESSLICH, JEDOCH NICHT BESCHRÄNKT AUF DIE BESCHAFFUNG VON ERSATZWAREN ODER -DIENSTLEISTUNGEN, NUTZUNGS-, DATEN- ODER GEWINNVERLUSTE ODER UNTERBRECHUNG DES GESCHÄFTSBETRIEBS), UNABHÄNGIG DAVON, WIE SIE VERURSACHT WURDEN UND AUF WELCHER HAFTUNGSTHEORIE SIE BERUHEN, OB AUS VERTRAGLICH FESTGELEGTER HAFTUNG, VERSCHULDENSUNABHÄNGIGER HAFTUNG ODER DELIKTSHAFTUNG (EINSCHLIESSLICH FAHRLÄSSIGKEIT ODER AUF ANDEREM WEGE), DIE IN IRGEND EINER WEISE AUS DER NUTZUNG DIESER SOFTWARE RESULTIEREN, SELBST WENN AUF DIE MÖGLICHKEIT DERARTIGER SCHÄDEN HINGEWIESEN WURDE.

NetApp behält sich das Recht vor, die hierin beschriebenen Produkte jederzeit und ohne Vorankündigung zu ändern. NetApp übernimmt keine Verantwortung oder Haftung, die sich aus der Verwendung der hier beschriebenen Produkte ergibt, es sei denn, NetApp hat dem ausdrücklich in schriftlicher Form zugestimmt. Die Verwendung oder der Erwerb dieses Produkts stellt keine Lizenzierung im Rahmen eines Patentrechts, Markenrechts oder eines anderen Rechts an geistigem Eigentum von NetApp dar.

Das in diesem Dokument beschriebene Produkt kann durch ein oder mehrere US-amerikanische Patente, ausländische Patente oder anhängige Patentanmeldungen geschützt sein.

ERLÄUTERUNG ZU „RESTRICTED RIGHTS“: Nutzung, Vervielfältigung oder Offenlegung durch die US-Regierung unterliegt den Einschränkungen gemäß Unterabschnitt (b)(3) der Klausel „Rights in Technical Data – Noncommercial Items“ in DFARS 252.227-7013 (Februar 2014) und FAR 52.227-19 (Dezember 2007).

Die hierin enthaltenen Daten beziehen sich auf ein kommerzielles Produkt und/oder einen kommerziellen Service (wie in FAR 2.101 definiert) und sind Eigentum von NetApp, Inc. Alle technischen Daten und die Computersoftware von NetApp, die unter diesem Vertrag bereitgestellt werden, sind gewerblicher Natur und wurden ausschließlich unter Verwendung privater Mittel entwickelt. Die US-Regierung besitzt eine nicht ausschließliche, nicht übertragbare, nicht unterlizenzierbare, weltweite, limitierte unwiderrufliche Lizenz zur Nutzung der Daten nur in Verbindung mit und zur Unterstützung des Vertrags der US-Regierung, unter dem die Daten bereitgestellt wurden. Sofern in den vorliegenden Bedingungen nicht anders angegeben, dürfen die Daten ohne vorherige schriftliche Genehmigung von NetApp, Inc. nicht verwendet, offengelegt, vervielfältigt, geändert, aufgeführt oder angezeigt werden. Die Lizenzrechte der US-Regierung für das US-Verteidigungsministerium sind auf die in DFARS-Klausel 252.227-7015(b) (Februar 2014) genannten Rechte beschränkt.

Markeninformationen

NETAPP, das NETAPP Logo und die unter <http://www.netapp.com/TM> aufgeführten Marken sind Marken von NetApp, Inc. Andere Firmen und Produktnamen können Marken der jeweiligen Eigentümer sein.