



Migrazione dei database Oracle

Enterprise applications

NetApp
May 09, 2024

Sommario

- Migrazione dei database Oracle 1
 - Migrazione dei database Oracle sui sistemi di storage ONTAP 1
 - Pianificazione della migrazione dei database Oracle 1
 - Procedure 4
 - Script di esempio della procedura di migrazione Oracle 106

Migrazione dei database Oracle

Migrazione dei database Oracle sui sistemi di storage ONTAP

L'utilizzo delle funzionalità di una nuova piattaforma di storage impone un requisito inevitabile e prevede il posizionamento dei dati nel nuovo sistema di storage. ONTAP semplifica il processo di migrazione, inclusi aggiornamenti e migrazioni da ONTAP a ONTAP, importazioni di LUN esterne e procedure per l'utilizzo diretto del sistema operativo host o del software di database Oracle.



Questa documentazione sostituisce il report tecnico precedentemente pubblicato *TR-4534: Migrazione dei database Oracle in sistemi di storage NetApp*

Nel caso di un nuovo progetto di database, questo non rappresenta un problema, poiché gli ambienti di database e applicazioni sono stati costruiti in sede. La migrazione, tuttavia, pone sfide speciali in relazione all'interruzione del business, al tempo necessario per il completamento della migrazione, alle competenze necessarie e alla minimizzazione del rischio.

Script

La presente documentazione contiene script di esempio. Questi script forniscono metodi di esempio per automatizzare vari aspetti della migrazione per ridurre la possibilità di errori da parte degli utenti. Gli script possono ridurre le richieste generali del personale IT responsabile della migrazione e accelerare il processo complessivo. Questi script sono ricavati da progetti di migrazione effettivi eseguiti dai servizi di assistenza professionale NetApp e dai partner NetApp. Nella presente documentazione sono riportati alcuni esempi del loro utilizzo.

Pianificazione della migrazione dei database Oracle

La migrazione dei dati Oracle può avvenire a uno di tre livelli: Database, host o storage array.

Le differenze risiedono in quale componente della soluzione globale è responsabile dello spostamento dei dati: Il database, il sistema operativo host o il sistema di archiviazione.

La figura riportata di seguito mostra un esempio dei livelli di migrazione e del flusso di dati. In caso di migrazione a livello di database, i dati vengono spostati dal sistema di storage originale ai livelli di host e database nel nuovo ambiente. La migrazione a livello di host è simile, ma i dati non passano attraverso il livello di applicazione e vengono invece scritti nella nuova posizione utilizzando i processi degli host. Infine, con la migrazione a livello di storage, un array come un sistema NetApp FAS si occupa dello spostamento dei dati.

[Errore: Immagine grafica mancante]

Una migrazione a livello di database si riferisce generalmente all'utilizzo di Oracle log shipping attraverso un database di standby per completare una migrazione a livello di Oracle. Le migrazioni a livello di host vengono eseguite utilizzando le funzionalità native della configurazione del sistema operativo host. Questa configurazione include le operazioni di copia dei file utilizzando comandi quali cp, tar e Oracle Recovery Manager (RMAN) o un gestore del volume logico (LVM) per spostare i byte sottostanti di un file system. Oracle Automatic Storage Management (ASM) è classificato come capacità a livello di host perché viene eseguito al

di sotto del livello dell'applicazione di database. ASM sostituisce il normale volume manager logico su un host. Infine, i dati possono essere migrati a livello di storage array, il che significa sotto il livello del sistema operativo.

Considerazioni sulla pianificazione

La scelta migliore per la migrazione dipende da una combinazione di fattori, inclusa la dimensione dell'ambiente da migrare, la necessità di evitare il downtime e lo sforzo complessivo necessario per eseguire la migrazione. Ovviamente, i database di grandi dimensioni richiedono più tempo e lavoro per la migrazione, ma la complessità di una migrazione di questo tipo è minima. I database di piccole dimensioni possono essere migrati rapidamente, ma se ne devono migrare migliaia, la portata dello sforzo può creare complicazioni. Infine, più grande è il database, più probabile è che l'IT sia business-critical, generando la necessità di ridurre al minimo i downtime mantenendo un percorso di back-out.

Alcune considerazioni per la pianificazione di una strategia di migrazione sono discusse qui.

Dimensioni dei dati

Le dimensioni dei database da migrare influiscono ovviamente sulla pianificazione della migrazione, sebbene le dimensioni non influiscano necessariamente sul tempo di cutover. Quando è necessario migrare una grande quantità di dati, l'aspetto più importante è la larghezza di banda. Le operazioni di copia vengono in genere eseguite con un efficiente i/o sequenziale. Come stima conservativa, si presuppone un utilizzo del 50% della larghezza di banda della rete disponibile per le operazioni di copia. Ad esempio, una porta FC da 8GB GB può trasferire in teoria circa 800Mbps GB. Ipotizzando un utilizzo del 50%, è possibile copiare un database a una velocità di circa 400Mbps KB. Pertanto, un database 10TB può essere copiato in circa sette ore a questa velocità.

La migrazione su distanze più lunghe in genere richiede un approccio più creativo, ad esempio il processo di distribuzione dei log illustrato nella ["Spostamento file dati online"](#). Le reti IP a lunga distanza raramente dispongono di larghezza di banda in qualsiasi punto vicino alle velocità LAN o SAN. In un caso, NetApp ha assistito alla migrazione a lunga distanza di un database 220TB con tassi di generazione di log di archiviazione molto elevati. L'approccio scelto per il trasferimento dei dati era la spedizione giornaliera dei nastri, perché questo metodo offriva la massima larghezza di banda possibile.

Numero di database

In molti casi, il problema dello spostamento di una grande quantità di dati non è la dimensione dei dati, ma piuttosto la complessità della configurazione che supporta il database. Semplicemente sapere che 50TB database devono essere migrati non è sufficiente. Può essere un singolo database mission-critical 50TB, una raccolta di 4 database legacy 000 o un mix di dati di produzione e non. In alcuni casi, gran parte dei dati è costituita da cloni di un database di origine. Non è necessario migrare questi cloni perché possono essere facilmente ricreati, specialmente quando la nuova architettura è progettata per sfruttare i volumi FlexClone di NetApp.

Per la pianificazione della migrazione, è necessario comprendere il numero dei database interessati e la priorità da assegnare. Con l'aumento del numero di database, l'opzione di migrazione preferita tende a essere più bassa e più bassa nello stack. Ad esempio, la copia di un singolo database può essere eseguita facilmente con RMAN e con una breve interruzione del servizio. Si tratta di una replica a livello di host.

Se ci sono 50 database, potrebbe essere più facile evitare di impostare una nuova struttura del file system per ricevere una copia RMAN e spostare invece i dati sul posto. Questo processo può essere eseguito sfruttando la migrazione LVM basata su host per spostare i dati dalle vecchie LUN ai nuovi LUN. In tal modo, la responsabilità viene trasferita dal team di amministratori del database (DBA) al team del sistema operativo e, di conseguenza, i dati vengono migrati in modo trasparente rispetto al database. La configurazione del file

system rimane invariata.

Infine, se occorre migrare 500 database su 200 server, è possibile utilizzare opzioni basate sullo storage come la funzionalità FLI (ONTAP Foreign LUN Import) per eseguire una migrazione diretta delle LUN.

Requisiti di riarchitettura

In genere, per sfruttare le funzionalità del nuovo storage array è necessario modificare il layout dei file del database; tuttavia, non sempre questo avviene. Ad esempio, le funzionalità degli array all-flash EF-Series sono rivolte principalmente alle performance e all'affidabilità della SAN. Nella maggior parte dei casi, i database possono essere migrati su un array EF-Series senza particolari considerazioni sul layout dei dati. Gli unici requisiti sono IOPS elevati, bassa latenza e solida affidabilità. Sebbene esistano Best practice correlate a fattori quali la configurazione RAID o Dynamic Disk Pools, i progetti EF-Series raramente richiedono modifiche significative all'architettura dello storage generale per sfruttare tali funzionalità.

Al contrario, la migrazione a ONTAP richiede in genere una maggiore considerazione del layout del database per garantire che la configurazione finale fornisca il massimo valore. In sé, ONTAP offre molte funzionalità per un ambiente di database, anche senza interventi specifici sull'architettura. Soprattutto, offre la possibilità di migrare senza interruzioni al nuovo hardware quando l'hardware attuale termina la sua vita utile. In generale, la migrazione a ONTAP è l'ultima migrazione che è necessario eseguire. L'hardware successivo viene aggiornato e i dati vengono migrati senza interruzioni sui nuovi supporti.

Con una certa pianificazione, ancora più benefici sono disponibili. Le considerazioni più importanti riguardano l'uso delle istantanee. Le snapshot sono la base per l'esecuzione di backup, ripristini e operazioni di cloning quasi istantanei. Come esempio della potenza delle istantanee, il più grande utilizzo noto è con un singolo database di 996TB in esecuzione su circa 250 LUN su 6 controller. È possibile eseguire il backup di questo database in 2 minuti, ripristinarlo in 2 minuti e clonarlo in 15 minuti. Tra gli altri benefici, è inclusa la capacità di spostare i dati nel cluster in risposta alle variazioni del carico di lavoro e all'applicazione di controlli di qualità del servizio (QoS) per offrire performance buone e coerenti in un ambiente multi-database.

Tecnologie come controlli della QoS, trasferimento dei dati, snapshot e cloning funzionano praticamente in ogni configurazione. Tuttavia, un certo pensiero è generalmente richiesto per elevare i benefici. In alcuni casi, i layout dello storage del database possono richiedere modifiche di progettazione per massimizzare l'investimento nel nuovo storage array. Tali modifiche di progettazione possono influire sulla strategia di migrazione perché le migrazioni basate su host o su storage replicano il layout dei dati originale. Per completare la migrazione e offrire un layout dei dati ottimizzato per ONTAP potrebbero essere necessari ulteriori passaggi. Le procedure illustrate nella ["Panoramica delle procedure di migrazione Oracle"](#) in seguito, dimostrano alcuni metodi non solo per migrare un database, ma anche per eseguirne la migrazione nel layout finale ottimale con il minimo sforzo.

Tempo di cutover

Occorre determinare il disservizio massimo consentito del servizio durante il cutover. È un errore comune presumere che l'intero processo di migrazione causi interruzioni. È possibile eseguire numerose attività prima dell'inizio di qualsiasi interruzione del servizio e completare la migrazione senza interruzioni o black-out attraverso diverse opzioni. Anche quando è inevitabile un'interruzione, è comunque necessario definire il fuori servizio massimo consentito poiché la durata del tempo di cutover varia da procedura a procedura.

Ad esempio, la copia di un database 10TB richiede in genere circa sette ore. Se le esigenze aziendali rendono possibile un'interruzione di sette ore, la copia dei file è un'opzione semplice e sicura per la migrazione. Se cinque ore sono inaccettabili, un semplice log-processo di spedizione (vedere ["Log shipping di Oracle"](#)) può essere impostato con il minimo sforzo per ridurre il tempo di cutover a circa 15 minuti. Durante questo periodo, un amministratore di database può completare il processo. Se 15 minuti sono inaccettabili, è possibile automatizzare il processo di cutover finale tramite script per ridurre il tempo di cutover a pochi minuti. È

sempre possibile accelerare una migrazione, anche se ciò comporta costi di tempo e lavoro. Gli obiettivi del tempo di cutover devono basarsi su ciò che è accettabile per l'azienda.

Percorso di ritorno

Nessuna migrazione è completamente priva di rischi. Anche se la tecnologia funziona perfettamente, c'è sempre la possibilità di errori da parte dell'utente. Il rischio associato a un percorso di migrazione scelto deve essere preso in considerazione insieme alle conseguenze di una migrazione non riuscita. Ad esempio, la capacità di migrazione trasparente dello storage online di Oracle ASM è una delle sue caratteristiche principali e questo metodo è uno dei più affidabili. Tuttavia, i dati vengono copiati irreversibilmente con questo metodo. Nel caso altamente improbabile in cui si verifichi un problema con ASM, non esiste un facile percorso di back-out. L'unica opzione è ripristinare l'ambiente originale o utilizzare ASM per riportare la migrazione ai LUN originali. Il rischio può essere minimizzato, ma non eliminato, eseguendo un backup di tipo snapshot sul sistema di storage originale, supponendo che il sistema sia in grado di eseguire tale operazione.

Prova

Alcune procedure di migrazione devono essere verificate completamente prima dell'esecuzione. La necessità di migrazione e verifica del processo di cutover è una richiesta comune con i database mission-critical per i quali la migrazione deve avere successo e il downtime deve essere ridotto al minimo. Inoltre, i test di accettazione da parte dell'utente sono spesso inclusi come parte del lavoro di post-migrazione e il sistema complessivo può essere riportato in produzione solo dopo il completamento di questi test.

In caso di necessità di prove, diverse funzionalità di ONTAP possono rendere il processo molto più semplice. In particolare, le istantanee possono ripristinare un ambiente di test e creare rapidamente più copie di un ambiente di database efficienti in termini di spazio.

Procedure

Panoramica delle procedure di migrazione Oracle

Sono disponibili molte procedure per il database di migrazione Oracle. La giusta dipende dalle vostre esigenze aziendali.

In molti casi, gli amministratori di sistema e i DBA dispongono dei propri metodi preferiti per trasferire i dati dei volumi fisici, eseguire il mirroring e il demirroring o utilizzare Oracle RMAN per copiare i dati.

Queste procedure vengono fornite principalmente come guida per il personale IT meno esperto di alcune delle opzioni disponibili. Inoltre, vengono illustrate le attività, i requisiti di tempo e le richieste di competenze per ogni approccio alla migrazione. Ciò consente ad altre parti, come NetApp e i servizi professionali dei partner o i responsabili dell'IT, di apprezzare più pienamente i requisiti di ogni procedura.

Non esiste un'unica Best practice per la creazione di una strategia di migrazione. La creazione di un piano richiede prima di tutto la comprensione delle opzioni di disponibilità e quindi la selezione del metodo più adatto alle esigenze dell'azienda. La figura seguente illustra le considerazioni di base e le conclusioni tipiche dei clienti, ma non è applicabile a tutte le situazioni.

Ad esempio, un passaggio solleva il problema della dimensione totale del database. Il passaggio successivo dipende dal fatto che il database sia maggiore o minore di 1TB. I passaggi consigliati sono solo questi: Consigli basati su pratiche tipiche del cliente. La maggior parte dei clienti non utilizzerebbe DataGuard per copiare un database di piccole dimensioni, ma alcuni potrebbero farlo. La maggior parte dei clienti non tenterebbe di copiare un database 50TB per il tempo necessario, ma alcuni potrebbero avere una finestra di manutenzione sufficientemente grande da consentire tale operazione.

È possibile trovare un diagramma di flusso dei tipi di considerazioni sul percorso di migrazione più adatto "qui".

Spostamento file dati online

Oracle 12cR1 e versioni successive includono la possibilità di spostare un file dati mentre il database rimane online. Inoltre funziona tra diversi tipi di filesystem. Ad esempio, è possibile spostare un file dati da un filesystem xfs ad ASM. Questo metodo non viene generalmente utilizzato su larga scala a causa del numero di operazioni singole di spostamento del file di dati che sarebbero necessarie, ma è un'opzione che vale la pena considerare con database più piccoli con meno file di dati.

Inoltre, il semplice spostamento di un file dati è un'ottima opzione per la migrazione di parti di database esistenti. Ad esempio, è possibile ricollocare i file di dati meno attivi in uno storage più conveniente, ad esempio un volume FabricPool che consente di memorizzare blocchi inattivi in Object Store.

Migrazione a livello di database

La migrazione a livello di database significa consentire il trasferimento dei dati. In particolare, ciò significa spedizione dei log. Tecnologie come RMAN e ASM sono prodotti Oracle, ma, ai fini della migrazione, operano a livello di host in cui copiano i file e gestiscono i volumi.

Spedizione dei log

La base per la migrazione a livello di database è il log di archivio di Oracle, che contiene un registro delle modifiche apportate al database. Nella maggior parte dei casi, un registro di archiviazione fa parte di una strategia di backup e ripristino. Il processo di ripristino inizia con il ripristino di un database e quindi la riproduzione di uno o più log di archivio per portare il database allo stato desiderato. Questa stessa tecnologia di base può essere utilizzata per eseguire una migrazione con interruzioni delle operazioni minime o nulle. Cosa ancora più importante, questa tecnologia consente la migrazione senza intaccare il database originale, preservando un percorso di back-out.

Il processo di migrazione inizia con il ripristino di un backup del database su un server secondario. È possibile farlo in vari modi, ma la maggior parte dei clienti utilizza la normale applicazione di backup per ripristinare i file di dati. Una volta ripristinati i file di dati, gli utenti stabiliscono un metodo per la distribuzione dei log. L'obiettivo è creare un feed costante di log di archivio generati dal database primario e riprodurli sul database ripristinato per mantenerli entrambi vicini allo stesso stato. Quando arriva il tempo di cutover, il database di origine viene completamente arrestato e i log di archivio finali e, in alcuni casi, i log di redo vengono copiati e riprodotti. È fondamentale che i log di ripristino vengano presi in considerazione anche perché potrebbero contenere alcune delle transazioni finali impegnate.

Una volta trasferiti e riprodotti questi log, entrambi i database sono coerenti l'uno con l'altro. A questo punto, la maggior parte dei clienti esegue alcuni test di base. In caso di errori durante il processo di migrazione, la riproduzione del registro dovrebbe segnalare errori e errori. È comunque consigliabile eseguire alcuni test rapidi basati su query note o su attività guidate dalle applicazioni per verificare che la configurazione sia ottimale. È inoltre pratica comune creare una tabella di test finale prima di chiudere il database originale per verificare se è presente nel database migrato. Questa operazione garantisce che non siano stati commessi errori durante la sincronizzazione finale del registro.

Una semplice migrazione log-shipping può essere configurata fuori banda rispetto al database originale, il che lo rende particolarmente utile per i database mission-critical. Non sono richieste modifiche alla configurazione per il database di origine e il ripristino e la configurazione iniziale dell'ambiente di migrazione non hanno alcun effetto sulle operazioni di produzione. Una volta configurato, il log shipping pone alcune richieste di i/o sui server di produzione. Tuttavia, il log shipping è costituito da semplici letture sequenziali dei registri di archivio, che hanno scarse probabilità di influire sulle prestazioni del database di produzione.

La distribuzione dei log si è dimostrata particolarmente utile per progetti di migrazione a lunga distanza e ad

alta velocità di cambiamento. In un'istanza, è stata eseguita la migrazione di un singolo database 220TB in una nuova posizione a circa 500 km di distanza. La velocità di modifica era estremamente elevata e le restrizioni di sicurezza impedivano l'utilizzo di una connessione di rete. La spedizione dei log è stata eseguita utilizzando nastro e corriere. Una copia del database di origine è stata inizialmente ripristinata utilizzando le procedure descritte di seguito. Quindi, i registri sono stati spediti settimanalmente tramite corriere fino al momento del cutover, al momento della consegna del set finale di nastri e dell'applicazione dei registri al database di replica.

Oracle DataGuard

In alcuni casi, è garantito un ambiente DataGuard completo. Non è corretto utilizzare il termine DataGuard per fare riferimento a qualsiasi configurazione del database di standby o di distribuzione dei log. Oracle DataGuard è un framework completo per la gestione della replica dei database, ma non è una tecnologia di replica. Il vantaggio principale di un ambiente DataGuard completo in uno sforzo di migrazione è lo switchover trasparente da un database all'altro. DataGuard consente inoltre uno switchover trasparente nel database originale in caso di problemi, ad esempio problemi di prestazioni o connettività di rete nel nuovo ambiente. Un ambiente DataGuard completamente configurato richiede la configurazione non solo del livello del database ma anche delle applicazioni in modo che le applicazioni siano in grado di rilevare una modifica nella posizione del database primario. In generale, non è necessario utilizzare DataGuard per completare una migrazione, ma alcuni clienti hanno una vasta esperienza DataGuard in-house e già si affidano a essa per le attività di migrazione.

Riarchitettura

Come discusso in precedenza, per sfruttare le funzionalità avanzate degli storage array è talvolta necessario modificare il layout del database. Inoltre, una modifica nel protocollo di storage, come il passaggio da ASM a un file system NFS, altera necessariamente il layout del file system.

Uno dei principali vantaggi dei metodi di distribuzione dei log, incluso DataGuard, è che la destinazione di replica non deve corrispondere all'origine. Non vi sono problemi con l'utilizzo di un approccio di log-shipping per migrare da ASM a un normale file system o viceversa. Il layout preciso dei file di dati può essere modificato a destinazione per ottimizzare l'uso della tecnologia Pluggable Database (PDB) o per impostare i controlli QoS in modo selettivo su determinati file. In altre parole, un processo di migrazione basato sul log shipping consente di ottimizzare il layout dello storage del database in modo semplice e sicuro.

Risorse dei server

Un limite alla migrazione a livello di database è la necessità di un secondo server. Questo secondo server può essere utilizzato in due modi:

1. È possibile utilizzare il secondo server come nuova casa permanente per il database.
2. È possibile utilizzare il secondo server come server di staging temporaneo. Una volta completata e testata la migrazione dei dati nel nuovo storage array, i file system LUN o NFS vengono disconnessi dal server di staging e riconnessi al server originale.

La prima opzione è la più semplice, ma l'utilizzo potrebbe non essere possibile in ambienti molto grandi che richiedono server molto potenti. La seconda opzione richiede ulteriore lavoro per riportare i file system nella posizione originale. Si tratta di una semplice operazione in cui NFS viene utilizzato come protocollo storage, poiché i file system possono essere smontati dal server di staging e rimontati sul server originale.

I file system basati su blocchi richiedono lavoro extra per l'aggiornamento dello zoning FC o degli iSCSI initiator. Con la maggior parte dei gestori di volumi logici (incluso ASM), i LUN vengono automaticamente rilevati e portati online una volta resi disponibili sul server originale. Tuttavia, alcune implementazioni di file system e LVM potrebbero richiedere più lavoro per esportare e importare i dati. La procedura precisa può variare, ma in genere è facile stabilire una procedura semplice e ripetibile per completare la migrazione e

ripristinare i dati sul server originale.

Sebbene sia possibile impostare la distribuzione dei log e replicare un database all'interno di un singolo ambiente server, la nuova istanza deve avere un SID di processo diverso per riprodurre i log. È possibile visualizzare temporaneamente il database con un diverso gruppo di ID di processo con un SID diverso e modificarlo in un secondo momento. Tuttavia, questo può portare a numerose e complicate attività di gestione ed espone l'ambiente di database al rischio di errori dell'utente.

Migrazione a livello di host

Migrare i dati a livello di host significa utilizzare il sistema operativo host e le utility associate per completare la migrazione. Questo processo include qualsiasi utility che copia i dati, inclusi Oracle RMAN e Oracle ASM.

Copia dei dati

Il valore di un'operazione di copia semplice non deve essere sottovalutato. Le moderne infrastrutture di rete sono in grado di spostare i dati a velocità misurate in gigabyte al secondo, mentre le operazioni di copia dei file si basano su un efficiente i/o di lettura e scrittura sequenziale. L'interruzione è inevitabile con un'operazione di copia dell'host rispetto alla spedizione dei log, ma la migrazione non riguarda solo lo spostamento dei dati. In genere sono incluse le modifiche alla rete, il tempo di riavvio del database e i test post-migrazione.

Il tempo effettivo richiesto per copiare i dati potrebbe non essere significativo. Inoltre, l'operazione di copia preserva un percorso di back-out garantito perché i dati originali non vengono intatti. In caso di problemi durante il processo di migrazione, è possibile riattivare i file system originali con i dati originali.

Riformulazione

Replatforming si riferisce a una modifica del tipo di CPU. Quando un database viene migrato da una piattaforma Solaris, AIX o HP-UX tradizionale a x86 Linux, i dati devono essere riformattati a causa delle modifiche apportate all'architettura della CPU. Le CPU SPARC, IA64 e POWER sono note come grandi processori endian, mentre le architetture x86 e x86_64 sono note come Little endian. Di conseguenza, alcuni dati all'interno dei file di dati Oracle vengono ordinati in modo diverso a seconda del processore in uso.

Tradizionalmente, i clienti utilizzano DataPump per replicare i dati su più piattaforme. DataPump è un'utilità che crea un tipo speciale di esportazione dei dati logici che può essere importata più rapidamente nel database di destinazione. Poiché crea una copia logica dei dati, DataPump lascia alle spalle le dipendenze dell'endianness del processore. Anche se alcuni clienti usano DataPump per il replatform, con Oracle 11g è ora disponibile un'opzione più rapida: Tablespace trasportabili su più piattaforme. Questo avanzamento consente di convertire un tablespace in un diverso formato endian. Si tratta di una trasformazione fisica che offre prestazioni migliori rispetto a un'esportazione DataPump, che deve convertire i byte fisici in dati logici e quindi riconvertirli in byte fisici.

Una discussione completa su DataPump e tablespace trasportabili va oltre la documentazione relativa al NetApp dell'ambito, ma NetApp offre alcuni consigli basati sulla nostra esperienza nell'assistenza ai clienti durante la migrazione a un nuovo log di storage array con una nuova architettura della CPU:

- Se si utilizza DataPump, il tempo necessario per completare la migrazione deve essere misurato in un ambiente di test. A volte i clienti vengono sorpresi del tempo necessario per completare la migrazione. Questo downtime aggiuntivo e inatteso può causare interruzioni delle attività.
- Molti clienti credono erroneamente che gli spazi di tabella trasportabili su più piattaforme non richiedano la conversione dei dati. Quando si utilizza una CPU con un endian diverso, viene utilizzato un RMAN `convert` l'operazione deve essere eseguita sui file di dati in anticipo. Non si tratta di un'operazione istantanea. In alcuni casi, il processo di conversione può essere accelerato avendo più thread che operano su file di dati diversi, ma il processo di conversione non può essere evitato.

Migrazione guidata dal volume logico

Le LVM funzionano prendendo un gruppo di uno o più LUN e suddividendoli in piccole unità generalmente denominate estensioni. Il pool di estensioni viene quindi utilizzato come origine per creare volumi logici essenzialmente virtualizzati. Questo livello di virtualizzazione offre valore in vari modi:

- I volumi logici possono utilizzare estensioni tratte da più LUN. Quando un file system viene creato su un volume logico, può utilizzare le funzionalità con le performance complete di tutte le LUN. Inoltre, promuove il caricamento uniforme di tutte le LUN nel gruppo di volumi, offrendo performance più prevedibili.
- I volumi logici possono essere ridimensionati aggiungendo e, in alcuni casi, rimuovendo le estensioni. Il ridimensionamento di un file system su un volume logico avviene in genere senza interruzione delle attività.
- È possibile migrare i volumi logici senza interruzioni spostando le estensioni sottostanti.

La migrazione tramite LVM funziona in due modi: Spostare un'estensione o specchiare/demirrorizzare un'estensione. La migrazione LVM utilizza l'efficiente i/o sequenziale a blocchi di grandi dimensioni e solo raramente crea problemi di performance. In tal caso, sono solitamente disponibili opzioni per la riduzione della velocità di i/O. In tal modo, si aumenta il tempo necessario per completare la migrazione, riducendo al contempo il carico di i/o sui sistemi host e di storage.

Specchiatura e demirrorazione

Alcuni gestori di volumi, come AIX LVM, consentono all'utente di specificare il numero di copie per ogni estensione e di controllare quali periferiche ospitano ciascuna copia. La migrazione viene eseguita prelevando un volume logico esistente, eseguendo il mirroring delle estensioni sottostanti nei nuovi volumi, attendendo la sincronizzazione delle copie e rilasciando la copia precedente. Se si desidera un percorso di back-out, è possibile creare un'istantanea dei dati originali prima del punto in cui viene rilasciata la copia speculare. In alternativa, è possibile arrestare brevemente il server per mascherare i LUN originali prima di eliminare forzatamente le copie mirror contenute. In tal modo, si preserva una copia recuperabile dei dati nella loro posizione originale.

Estensione della migrazione

Quasi tutti i gestori di volumi consentono la migrazione delle estensioni e talvolta esistono diverse opzioni. Ad esempio, alcuni responsabili di volume consentono a un amministratore di spostare le singole estensioni per un volume logico specifico dal vecchio al nuovo storage. I gestori di volume come Linux LVM2 offrono `pvmove` che riposiziona tutti gli extent sul dispositivo LUN specificato in un nuovo LUN. Una volta evacuata, la vecchia LUN può essere rimossa.



Il rischio principale per le operazioni è la rimozione delle LUN vecchie e non utilizzate dalla configurazione. È necessario prestare la massima attenzione quando si modifica la suddivisione in zone FC e si rimuovono i dispositivi LUN obsoleti.

Gestione automatica dello storage Oracle

Oracle ASM è un volume manager e un file system logici combinati. A un livello elevato, Oracle ASM prende una raccolta di LUN, le suddivide in piccole unità di allocazione e le presenta come un singolo volume noto come gruppo di dischi ASM. ASM include inoltre la possibilità di eseguire il mirroring del gruppo di dischi impostando il livello di ridondanza. Un volume può essere senza mirror (ridondanza esterna), con mirroring (ridondanza normale) o con mirroring a tre vie (ridondanza elevata). Prestare attenzione durante la configurazione del livello di ridondanza perché non può essere modificato dopo la creazione.

ASM fornisce anche funzionalità di file system. Sebbene il file system non sia visibile direttamente dall'host, il database Oracle può creare, spostare ed eliminare file e directory in un gruppo di dischi ASM. Inoltre, è

possibile navigare nella struttura utilizzando l'utilità `asmcmd`.

Come per altre implementazioni LVM, Oracle ASM ottimizza le performance di i/o mediante lo striping e il bilanciamento del carico dell'i/o di ciascun file su tutti i LUN disponibili. In secondo luogo, è possibile riposizionare le estensioni sottostanti per consentire sia il ridimensionamento del gruppo di dischi ASM sia la migrazione. Oracle ASM automatizza il processo mediante l'operazione di ribilanciamento. Le nuove LUN vengono aggiunte a un gruppo di dischi ASM e le vecchie LUN vengono eliminate, innescando il trasferimento dell'estensione e la successiva caduta della LUN evacuata dal gruppo di dischi. Questo processo è uno dei metodi di migrazione più comprovati e l'affidabilità di ASM nel fornire una migrazione trasparente è probabilmente la sua caratteristica più importante.



Poiché il livello di mirroring di Oracle ASM è fisso, non può essere utilizzato con il metodo di migrazione mirror e demirroring.

Migrazione a livello di storage

Migrazione a livello di storage: Migrazione al di sotto del livello dell'applicazione e del sistema operativo. In passato, questo a volte significava l'utilizzo di dispositivi specializzati che copiano i LUN a livello di rete, ma queste funzionalità ora si trovano in modo nativo in ONTAP.

SnapMirror

La migrazione di database da un sistema NetApp all'altro viene eseguita quasi universalmente con il software di replica dei dati NetApp SnapMirror. Il processo prevede la configurazione di una relazione di mirroring per i volumi da migrare, in modo che possano essere sincronizzati e quindi in attesa della finestra di cutover. Quando arriva, il database di origine viene arrestato, viene eseguito un aggiornamento finale del mirror e il mirror viene interrotto. I volumi di replica sono quindi pronti per l'uso, montando una directory del file system NFS contenuta oppure rilevando i LUN contenuti e avviando il database.

Il riposizionamento dei volumi in un singolo cluster ONTAP non viene preso in considerazione dalla migrazione, ma piuttosto da una routine `volume move` operazione. SnapMirror viene utilizzato come motore di replica dei dati all'interno del cluster. Questo processo è completamente automatizzato. Non esistono ulteriori passaggi da eseguire per la migrazione quando gli attributi del volume, come la mappatura delle LUN o le autorizzazioni di esportazione NFS, vengono spostati con il volume stesso. Il trasferimento non comporta interruzioni per le operazioni dell'host. In alcuni casi, l'accesso alla rete deve essere aggiornato per garantire che l'accesso ai dati appena ricollocati sia nel modo più efficiente possibile, ma anche queste attività non comportano interruzione delle attività.

Importazione di LUN esterne (FLI)

FLI è una funzione che consente a un sistema Data ONTAP con versione 8,3 o superiore di migrare una LUN esistente da un altro storage array. La procedura è semplice: Il sistema ONTAP viene sottoposto a zoning sull'array di storage esistente come se fosse un qualsiasi altro host SAN. Data ONTAP può quindi controllare le LUN legacy desiderate ed eseguire la migrazione dei dati sottostanti. Inoltre, il processo di importazione utilizza le impostazioni di efficienza del nuovo volume durante la migrazione dei dati, vale a dire che i dati possono essere compressi e deduplicati inline durante il processo di migrazione.

La prima implementazione di FLI in Data ONTAP 8,3 consentiva solo la migrazione offline. Si trattava di un trasferimento molto veloce, ma i dati LUN continuavano a non essere disponibili fino al completamento della migrazione. La migrazione online è stata introdotta in Data ONTAP 8,3.1. Questo tipo di migrazione consente di ridurre al minimo le interruzioni, consentendo a ONTAP di fornire dati LUN durante il processo di trasferimento. Si verifica una breve interruzione mentre l'host viene sottoposto a zoning per l'utilizzo dei LUN tramite ONTAP. Tuttavia, non appena tali modifiche vengono apportate, i dati sono ancora una volta accessibili e rimangono accessibili per l'intero processo di migrazione.

L'i/o in lettura viene fornito con un proxy tramite ONTAP fino al completamento dell'operazione di copia, mentre l'i/o in scrittura viene scritta in modo sincrono su LUN esterna e ONTAP. Le due copie LUN vengono mantenute sincronizzate in questo modo fino a quando l'amministratore non esegue un cutover completo che rilascia la LUN esterna e non replica più le scritture.

FLI è progettato per funzionare con FC, ma se si desidera passare a iSCSI, la LUN migrata può essere facilmente rimappata come una LUN iSCSI al termine della migrazione.

Tra le caratteristiche di FLI vi è il rilevamento e la regolazione automatici dell'allineamento. In questo contesto, il termine allineamento si riferisce a una partizione su un dispositivo LUN. Per ottenere prestazioni ottimali è necessario allineare l'i/o ai blocchi da 4K KB. Se una partizione viene posizionata su un offset che non è multiplo di 4K, le prestazioni ne risentono.

Esiste un secondo aspetto dell'allineamento che non può essere corretto regolando un offset di partizione, ovvero la dimensione del blocco del file system. Ad esempio, un file system ZFS generalmente utilizza per impostazione predefinita una dimensione di blocco interna di 512 byte. Altri clienti che utilizzano AIX hanno occasionalmente creato file system JFS2 con dimensioni blocco di 512 o 1, 024 byte. Anche se il file system potrebbe essere allineato a un limite di 4K, i file creati all'interno di tale file system non lo sono e le prestazioni ne risentono.

FLI non deve essere usato in queste circostanze. Anche se i dati sono accessibili dopo la migrazione, il risultato sono file system con gravi limitazioni delle prestazioni. In linea di principio, qualsiasi file system che supporti un carico di lavoro di sovrascrittura casuale su ONTAP dovrebbe utilizzare una dimensione del blocco di 4K KB. Ciò è applicabile principalmente a workload come file di dati di database e implementazioni di VDI. La dimensione del blocco può essere identificata utilizzando i comandi del sistema operativo host pertinente.

Ad esempio, su AIX, la dimensione del blocco può essere visualizzata con `lsfs -q`. Con Linux, `xfstool` e `tune2fs` può essere utilizzato per `ext3/ext4`, rispettivamente. Con `zfs`, il comando è `zdb -C`.

Il parametro che controlla la dimensione del blocco è `ashift` e generalmente il valore predefinito è 9, che significa 2^9 , o 512 byte. Per prestazioni ottimali, la `ashift` il valore deve essere 12 ($2^{12}=4K$). Questo valore viene impostato al momento della creazione di `zpool` e non può essere modificato, il che significa che i data `zpool` con un `ashift` oltre a 12 deve essere eseguita la migrazione copiando i dati in uno `zpool` appena creato.

Oracle ASM non ha dimensioni dei blocchi fondamentali. L'unico requisito è che la partizione su cui è stato creato il disco ASM sia allineata correttamente.

7-Mode Transition Tool

7-Mode Transition Tool (7MTT) è un'utility di automazione utilizzata per migrare configurazioni 7- Mode di grandi dimensioni a ONTAP. La maggior parte dei clienti che gestiscono i database trovano altri metodi più semplici, in parte perché eseguono di solito la migrazione dei database piuttosto che trasferire l'intero footprint dello storage. Inoltre, i database sono spesso solo una parte di un ambiente di storage più ampio. Pertanto, spesso i database vengono migrati singolarmente, quindi l'ambiente rimanente può essere spostato con 7MTT.

Alcuni clienti con sistemi di storage dedicati a ambienti di database complicati hanno un numero limitato ma significativo di essi. Questi ambienti potrebbero contenere molti volumi, snapshot e numerosi dettagli di configurazione, come autorizzazioni di esportazione, gruppi iniziatori LUN, autorizzazioni utente e configurazione del protocollo Lightweight Directory Access Protocol. In questi casi, le capacità di automazione di 7MTT possono semplificare una migrazione.

7MTT può funzionare in una delle due modalità seguenti:

- **Copy- Based Transition (CBT).** 7MTT con CBT imposta i volumi SnapMirror da un sistema 7- Mode

esistente nel nuovo ambiente. Una volta sincronizzati i dati, 7MTT orchestra il processo di cutover.

- **Copy- Free Transition (CFT).** 7MTT con CFT si basa sulla conversione in-place degli shelf di dischi 7-Mode esistenti. I dati non vengono copiati e gli shelf di dischi esistenti possono essere riutilizzati. La configurazione esistente di data Protection ed efficienza dello storage viene preservata.

La differenza principale tra queste due opzioni consiste nel fatto che la transizione senza copie è un approccio a big-bang, in cui tutti gli shelf di dischi collegati alla coppia ha 7-Mode originale devono essere ricollocati nel nuovo ambiente. Non esiste alcuna opzione per spostare un sottoinsieme di shelf. L'approccio basato sulla copia consente lo spostamento dei volumi selezionati. Esiste anche potenzialmente una finestra di cutover più lunga con transizione priva di copie a causa del legame necessario per la riselectone degli shelf di dischi e la conversione dei metadati. In base all'esperienza sul campo, NetApp consiglia di lasciare trascorrere 1 ora per il riposizionamento e il ripristino degli shelf di dischi e tra 15 minuti e 2 ore per la conversione dei metadati.

Migrazione dei file dati Oracle

È possibile spostare singoli file di dati Oracle con un singolo comando.

Ad esempio, il comando seguente sposta il file dati IOPST.dbf dal filesystem /oradata2 al filesystem /oradata3.

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

Lo spostamento di un file dati con questo metodo può essere lento, ma in genere non dovrebbe produrre i/o sufficienti da interferire con i carichi di lavoro del database quotidiani. Al contrario, la migrazione tramite il ribilanciamento di ASM può essere eseguita molto più rapidamente, ma con il rischio di rallentare il database globale durante lo spostamento dei dati.

È possibile misurare facilmente il tempo necessario per spostare i file di dati creando un file di dati di test e spostandolo. Il tempo trascorso per l'operazione viene registrato nei dati di v\$session:

```
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 questo esempio, il file spostato era datafile 8, della dimensione di 21GB GB e della durata di 6 minuti per la migrazione. Il tempo necessario dipende ovviamente dalle funzionalità del sistema di storage, della rete di storage e dall'attività complessiva del database che si verifica al momento della migrazione.

Migrazione del database Oracle tramite log shipping

L'obiettivo di una migrazione utilizzando la distribuzione dei log è creare una copia dei file di dati originali in una nuova posizione e quindi stabilire un metodo per la distribuzione delle modifiche nel nuovo ambiente.

Una volta stabiliti, è possibile automatizzare la spedizione e la riproduzione dei log per mantenere il database di replica ampiamente sincronizzato con l'origine. Ad esempio, un job cron può essere programmato per (a) copiare i log più recenti nella nuova posizione e (b) riprodurli ogni 15 minuti. In questo modo si riduce al minimo l'interruzione al momento del cutover, in quanto è necessario riprodurre non più di 15 minuti dei registri di archivio.

La procedura illustrata di seguito è essenzialmente un'operazione di clonazione del database. La logica illustrata è simile al motore all'interno di NetApp SnapManager per Oracle (SMO) e al plug-in NetApp SnapCenter per Oracle. Alcuni clienti utilizzano la procedura indicata negli script o nei workflow Wfa per le operazioni di cloning personalizzate. Sebbene questa procedura sia più manuale che non utilizzi SMO o SnapCenter, viene comunque rapidamente script e le API di gestione dei dati all'interno di ONTAP semplificano ulteriormente il processo.

Log shipping - dal file system al file system

In questo esempio viene illustrata la migrazione di un database denominato WAFFLE da un normale file system a un altro normale file system situato su un server diverso. Illustra anche l'utilizzo di SnapMirror per eseguire una copia rapida dei file di dati, ma questa non è parte integrante della procedura generale.

Creare il backup del database

Il primo passo consiste nel creare un backup del database. In particolare, questa procedura richiede una serie di file di dati che possono essere utilizzati per la riproduzione del log di archivio.

Ambiente

In questo esempio, il database di origine si trova su un sistema ONTAP. Il metodo più semplice per creare un backup di un database consiste nell'utilizzare uno snapshot. Il database viene messo in modalità di backup a caldo per alcuni secondi mentre un `snapshot create` l'operazione viene eseguita sul volume che ospita i file di dati.

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

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

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

Il risultato è un'istantanea sul disco chiamata `hotbackup` che contiene un'immagine dei file di dati in modalità

di backup a caldo. Se combinati con i log di archivio appropriati per rendere i file di dati coerenti, i dati di questa snapshot possono essere utilizzati come base di un ripristino o di un clone. In questo caso, viene replicato sul nuovo server.

Ripristino in un nuovo ambiente

Ora il backup deve essere ripristinato nel nuovo ambiente. Questa operazione può essere eseguita in vari modi, tra cui Oracle RMAN, ripristino da un'applicazione di backup come NetBackup o semplice operazione di copia dei file di dati inseriti in modalità hot backup.

In questo esempio, SnapMirror viene utilizzato per replicare l'hot backup dello snapshot in una nuova posizione.

1. Creare un nuovo volume per ricevere i dati dello snapshot. Inizializzare il mirroring da `jfsc1_oradata` a `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
-destination-path /vol_oradata
Cluster01::*>
```

2. Una volta impostato lo stato da SnapMirror, a indicare che la sincronizzazione è completa, aggiornare il mirror in base allo snapshot desiderato,

```
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. La sincronizzazione può essere verificata visualizzando `newest-snapshot` sul volume speculare.

```
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. Lo specchio può quindi essere rotto.

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

5. Montare il nuovo file system. con i file system basati su blocchi, le procedure precise variano in base al LVM in uso. È necessario configurare lo zoning FC o le connessioni iSCSI. Dopo aver stabilito la connettività ai LUN, comandi come Linux `pvscan` Potrebbe essere necessario per rilevare quali gruppi di volumi o LUN devono essere configurati correttamente per essere rilevati da ASM.

In questo esempio viene utilizzato un semplice file system NFS. Questo file system può essere montato direttamente.

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

Creare un modello di creazione controlfile

Successivamente, è necessario creare un modello controlfile. Il `backup controlfile to trace` comando crea comandi di testo per ricreare un controlfile. In alcuni casi, questa funzione può risultare utile per ripristinare un database da un backup e viene spesso utilizzata con script che eseguono attività come la clonazione dei database.

1. L'output del comando seguente viene utilizzato per ricreare i file di controllo per il database migrato.

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

2. Una volta creati i file di controllo, copiarli nel nuovo server.


```
[oracle@jfsc3 tmp]$ scp oracle@jfsc1:/tmp/waffle.ctrl /tmp/  
oracle@jfsc1's password:  
waffle.ctrl                                100% 5199  
5.1KB/s   00:00
```

File dei parametri di backup

Nel nuovo ambiente è necessario anche un file di parametri. Il metodo più semplice consiste nel creare un pfile dal file spfile o pfile corrente. In questo esempio, il database di origine utilizza un spfile.

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

Crea voce oratab

La creazione di una voce oratab è necessaria per il corretto funzionamento di utility come oraenv. Per creare una voce oratab, completare il passaggio seguente.

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

Preparare la struttura delle directory

Se le directory richieste non sono già presenti, è necessario crearle oppure la procedura di avvio del database non riesce. Per preparare la struttura di directory, completare i seguenti requisiti minimi.

```
[oracle@jfsc3 ~]$ . oraenv  
ORACLE_SID = [oracle] ? WAFFLE  
The Oracle base has been set to /orabin  
[oracle@jfsc3 ~]$ cd $ORACLE_BASE  
[oracle@jfsc3 orabin]$ cd admin  
[oracle@jfsc3 admin]$ mkdir WAFFLE  
[oracle@jfsc3 admin]$ cd WAFFLE  
[oracle@jfsc3 WAFFLE]$ mkdir adump dpdump pfile scripts xdb_wallet
```

Aggiornamenti del file dei parametri

1. Per copiare il file dei parametri nel nuovo server, eseguire i seguenti comandi. La posizione predefinita è \$ORACLE_HOME/dbs directory. In questo caso, il pfile può essere posizionato ovunque. Viene utilizzata solo come fase intermedia del processo di migrazione.

```

[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. Modificare il file come richiesto. Ad esempio, se la posizione del log di archivio è stata modificata, il file pfile deve essere modificato per riflettere la nuova posizione. In questo esempio, vengono ricollocati solo i file di controllo, in parte per distribuirli tra i file system di log e di dati.

```

[root@jpsc1 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. Al termine delle modifiche, creare un file spfile basato su questo file pfile.

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

Ricreare i file di controllo

In una fase precedente, l'output di `backup controlfile to trace` è stato copiato nel nuovo server. La parte specifica dell'uscita richiesta è la `controlfile recreation` comando. Queste informazioni si trovano nel file sotto la sezione contrassegnata `Set #1. NORESETLOGS`. Inizia con la linea `create controlfile reuse database` e dovrebbe includere la parola `noresetlogs`. Termina con il punto e virgola (;).

1. In questa procedura di esempio, il file viene letto come segue.

```
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. Modificare lo script come desiderato per riflettere la nuova posizione dei vari file. Ad esempio, alcuni file di dati noti per supportare un i/o elevato potrebbero essere reindirizzati a un file system su un Tier di storage dalle performance elevate. In altri casi, le modifiche possono essere apportate solo per motivi di amministrazione, ad esempio isolando i file di dati di un PDB in volumi dedicati.
3. In questo esempio, il `DATAFILE` stanza viene lasciata invariata, ma i log di redo vengono spostati in una nuova posizione in `/redo` piuttosto che condividere lo spazio con i log di archivio `/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>

```

Se i file sono posizionati in modo errato o i parametri non sono configurati correttamente, vengono generati errori che indicano ciò che deve essere corretto. Il database è montato, ma non è ancora aperto e non può essere aperto perché i file di dati in uso sono ancora contrassegnati come in modalità di backup a caldo. Per rendere il database coerente, è necessario applicare prima i registri di archiviazione.

Replica iniziale del registro

Per rendere coerenti i file di dati è necessaria almeno un'operazione di risposta del registro. Sono disponibili molte opzioni per la riproduzione dei registri. In alcuni casi, la posizione originale del log di archivio sul server originale può essere condivisa tramite NFS e la risposta del log può essere effettuata direttamente. In altri casi, è necessario copiare i registri di archivio.

Ad esempio, un semplice `scp` l'operazione può copiare tutti i log correnti dal server di origine al server di migrazione:

```

[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

```

Riproduzione del registro iniziale

Una volta che i file si trovano nella posizione del log di archivio, è possibile riprodurli inviando il comando `recover database until cancel` seguito dalla risposta `AUTO` per riprodurre automaticamente tutti i registri disponibili.

```

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

```

La risposta finale del log di archivio riporta un errore, ma questo è normale. Il registro indica che sqlplus stava cercando un particolare file di registro e non lo ha trovato. Il motivo è, molto probabilmente, che il file di registro non esiste ancora.

Se il database di origine può essere arrestato prima di copiare i registri di archivio, questa operazione deve essere eseguita una sola volta. I log di archivio vengono copiati e riprodotti, quindi il processo può continuare direttamente con il processo di cutover che replica i log di ripristino critici.

Replica e riproduzione incrementale dei log

Nella maggior parte dei casi, la migrazione non viene eseguita immediatamente. Il completamento del processo di migrazione potrebbe richiedere alcuni giorni o addirittura settimane, pertanto i log devono essere inviati continuamente al database di replica e riprodotti. Pertanto, quando arriva il cutover, occorre trasferire e riprodurre minimi dati.

In questo modo è possibile eseguire script in molti modi diversi, ma uno dei metodi più diffusi è l'utilizzo di rsync, un'utilità comune di replica dei file. Il modo più sicuro per usare questa utility è configurarla come demone. Ad esempio, il `rsyncd.conf` file che segue mostra come creare una risorsa chiamata `waffle.arch` a cui si accede con le credenziali utente Oracle e a cui è mappato `/logs/WAFFLE/arch`. Soprattutto, la risorsa è impostata su sola lettura, consentendo la lettura dei dati di produzione, ma non l'alterazione.

```
[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
```

Il seguente comando sincronizza la destinazione del log di archivio del nuovo server con la risorsa `rsync waffle.arch` sul server originale. Il `t` argomento in `rsync -p` fa sì che l'elenco di file venga confrontato in base alla data e all'ora e che vengano copiati solo i nuovi file. Questo processo fornisce un aggiornamento incrementale del nuovo server. Questo comando può anche essere programmato in cron per essere eseguito regolarmente.


```

[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

```

Una volta ricevuti i registri, è necessario riprodurli. Gli esempi precedenti mostrano l'uso di sqlplus per l'esecuzione manuale `recover database until cancel`, un processo che può essere facilmente automatizzato. Nell'esempio illustrato viene utilizzato lo script descritto nella ["Riproduci i registri sul database"](#). Gli script accettano un argomento che specifica il database che richiede un'operazione di riproduzione. Ciò consente di utilizzare lo stesso script in una migrazione di più database.

```
[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
```

Cutover

Quando si è pronti per il passaggio al nuovo ambiente, è necessario eseguire una sincronizzazione finale che includa sia i registri di archivio che i registri di ripristino. Se la posizione originale del log di ripristino non è già nota, è possibile identificarla come segue:

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

1. Arrestare il database di origine.
2. Eseguire una sincronizzazione finale dei registri di archivio sul nuovo server con il metodo desiderato.
3. I log di ripristino di origine devono essere copiati nel nuovo server. In questo esempio, i log di ripristino sono stati spostati in una nuova directory all'indirizzo `/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 questa fase, il nuovo ambiente di database contiene tutti i file necessari per portarlo nello stesso stato dell'origine. I registri di archivio devono essere riprodotti una volta finale.

```

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. Al termine, i log di ripristino devono essere riprodotti. Se il messaggio `Media recovery complete` viene restituito, il processo è riuscito e i database sono sincronizzati e possono essere aperti.

```

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

```

Log shipping - da ASM a file system

In questo esempio viene illustrato l'utilizzo di Oracle RMAN per la migrazione di un database. È molto simile all'esempio precedente di distribuzione del log del file system, ma i file su ASM non sono visibili all'host. Le uniche opzioni per la migrazione dei dati presenti sui dispositivi ASM sono il riposizionamento del LUN ASM o l'utilizzo di Oracle RMAN per eseguire le operazioni di copia.

Sebbene RMAN sia un requisito per la copia dei file da Oracle ASM, l'utilizzo di RMAN non è limitato a ASM. RMAN può essere utilizzato per migrare da qualsiasi tipo di storage a qualsiasi altro tipo.

Questo esempio mostra il trasferimento di un database chiamato PANCAKE dallo storage ASM a un file system normale situato su un server diverso nei percorsi `/oradata` e `/logs`.

Creare il backup del database

Il primo passo consiste nel creare un backup del database da migrare su un server alternativo. Poiché l'origine utilizza Oracle ASM, è necessario utilizzare RMAN. Un semplice backup RMAN può essere eseguito come segue. Questo metodo crea un backup con tag che può essere facilmente identificato da RMAN più avanti nella procedura.

Il primo comando definisce il tipo di destinazione per il backup e la posizione da utilizzare. Il secondo avvia il

backup dei soli file di dati.

```

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

```

Backup controlfile

Un controlfile di backup è necessario più avanti nella procedura per duplicate database operazione.

```
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
```

File dei parametri di backup

Nel nuovo ambiente è necessario anche un file di parametri. Il metodo più semplice consiste nel creare un pfile dal file spfile o pfile corrente. In questo esempio, il database di origine utilizza un spfile.

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

Script di ridenominazione file ASM

Diverse posizioni dei file attualmente definite nei file di controllo cambiano quando il database viene spostato. Lo script seguente crea uno script RMAN per semplificare il processo. Questo esempio mostra un database con un numero molto ridotto di file di dati, ma in genere i database contengono centinaia o addirittura migliaia di file di dati.

Questo script si trova in "[Conversione da ASM a nome file system](#)" e fa due cose.

In primo luogo, viene creato un parametro per ridefinire le posizioni del log di ripristino chiamate `log_file_name_convert`. Si tratta essenzialmente di un elenco di campi alternati. Il primo campo rappresenta la posizione di un registro di ripristino corrente, mentre il secondo campo rappresenta la posizione sul nuovo server. Il modello viene quindi ripetuto.

La seconda funzione consiste nel fornire un modello per la ridenominazione dei file di dati. Lo script esegue il ciclo dei file di dati, estrae le informazioni sul nome e sul numero del file e lo formatta come uno script RMAN. Quindi fa lo stesso con i file temporanei. Il risultato è un semplice script rman che può essere modificato come desiderato per assicurarsi che i file vengano ripristinati nella posizione desiderata.

```

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.

```

Acquisire l'output di questa schermata. Il `log_file_name_convert` il parametro viene inserito nel file pfile come descritto di seguito. Il file di dati RMAN rinominato e lo script duplicato devono essere modificati di conseguenza per posizionare i file di dati nelle posizioni desiderate. In questo esempio, sono tutti inseriti `/oradata/pancake`.

```

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';
}

```

Preparare la struttura delle directory

Gli script sono quasi pronti per l'esecuzione, ma prima la struttura di directory deve essere in posizione. Se le directory richieste non sono già presenti, è necessario crearle oppure la procedura di avvio del database non riesce. L'esempio riportato di seguito riflette i requisiti minimi.

```

[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

```

Crea voce oratab

Il seguente comando è necessario per il corretto funzionamento di utility come oraenv.

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

Aggiornamenti dei parametri

Il file pfile salvato deve essere aggiornato per riflettere eventuali modifiche di percorso sul nuovo server. Le modifiche al percorso del file di dati vengono modificate dallo script di duplicazione RMAN e quasi tutti i database richiedono modifiche al `control_files` e `log_archive_dest` parametri. Potrebbero inoltre essere presenti posizioni dei file di controllo che devono essere modificate e parametri quali `db_create_file_dest` Potrebbe non essere rilevante al di fuori di ASM. Prima di procedere, un DBA esperto deve esaminare attentamente le modifiche proposte.

In questo esempio, le modifiche principali sono le posizioni controlfile, la destinazione di archivio del registro e l'aggiunta di `log_file_name_convert` parametro.


```

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'

```

Dopo la conferma dei nuovi parametri, i parametri devono essere applicati. Esistono diverse opzioni, ma la maggior parte dei clienti crea un file spfile basato sul file pfile di testo.

```
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.
```

Nomount di avvio

Il passaggio finale prima della replica del database consiste nel visualizzare i processi del database ma non nel montare i file. In questa fase, potrebbero manifestarsi problemi con spfile. Se il `startup nomount` comando non riesce a causa di un errore di parametro, è semplice chiudere, correggere il modello pfile, ricaricarlo come spfile, e riprovare.

```
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
```

Duplicare il database

Il ripristino del backup RMAN precedente nella nuova posizione richiede più tempo rispetto ad altre fasi di questo processo. Il database deve essere duplicato senza modificare l'ID del database (DBID) o reimpostare i registri. Ciò impedisce l'applicazione dei registri, operazione necessaria per la sincronizzazione completa delle copie.

Connettersi al database con RMAN come aux ed eseguire il comando duplicato del database utilizzando lo script creato in un passaggio precedente.

```
[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
executing command: SET NEWNAME
executing command: SET NEWNAME
executing command: SET NEWNAME
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

```

Replica iniziale del registro

A questo punto è necessario inviare le modifiche dal database di origine a una nuova posizione. In tal caso, potrebbe essere necessario eseguire una combinazione di operazioni. Il metodo più semplice sarebbe fare in modo che RMAN nel database di origine scriva i log di archivio in una connessione di rete condivisa. Se una posizione condivisa non è disponibile, un metodo alternativo consiste nell'utilizzare RMAN per scrivere su un file system locale e quindi utilizzare rcp o rsync per copiare i file.

In questo esempio, il `/rman` Directory è una condivisione NFS disponibile sia per il database originale che per quello migrato.

Una questione importante in questo caso è la `disk format` clausola. Il formato del disco del backup è `%h_%e_%a.dbf`, Che significa che è necessario utilizzare il formato del numero di thread, il numero di sequenza e l'ID di attivazione per il database. Anche se le lettere sono diverse, questa corrisponde alla `log_archive_format='%t_%s_%r.dbf` parametro nel pfile. Questo parametro specifica inoltre i log di archivio nel formato di numero di thread, numero di sequenza e ID di attivazione. Il risultato finale è che i backup del file di registro sull'origine utilizzano una convenzione di denominazione prevista dal database. In questo modo, vengono eseguite operazioni come `recover database` molto più semplice perché sqlplus anticipa correttamente i nomi dei log di archivio da riprodurre.

```

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

```

Riproduzione del registro iniziale

Una volta che i file si trovano nella posizione del log di archivio, è possibile riprodurli inviando il comando `recover database until cancel` seguito dalla risposta `AUTO` per riprodurre automaticamente tutti i registri disponibili. Il file dei parametri sta attualmente indirizzando i log di archivio a `/logs/archive`, Ma non corrisponde alla posizione in cui RMAN è stato utilizzato per salvare i registri. La posizione può essere reindirizzata temporaneamente come segue prima di ripristinare il database.

```

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

```

La risposta finale del log di archivio riporta un errore, ma questo è normale. L'errore indica che sqlplus stava cercando un particolare file di registro e non lo ha trovato. Il motivo è molto probabile che il file di registro non esista ancora.

Se il database di origine può essere arrestato prima di copiare i registri di archivio, questa operazione deve essere eseguita una sola volta. I log di archivio vengono copiati e riprodotti, quindi il processo può continuare direttamente con il processo di cutover che replica i log di ripristino critici.

Replica e riproduzione incrementale dei log

Nella maggior parte dei casi, la migrazione non viene eseguita immediatamente. Il completamento del processo di migrazione potrebbe richiedere alcuni giorni o addirittura settimane, pertanto i log devono essere inviati continuamente al database di replica e riprodotti. In questo modo si assicura che i dati minimi debbano essere trasferiti e riprodotti all'arrivo del cutover.

Questo processo può essere facilmente gestito tramite script. Ad esempio, è possibile pianificare il seguente comando nel database originale per assicurarsi che la posizione utilizzata per la spedizione dei log venga

aggiornata continuamente.

```
[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.
```

Una volta ricevuti i registri, è necessario riprodurli. Gli esempi precedenti hanno mostrato l'uso di sqlplus per l'esecuzione manuale `recover database until cancel`, che può essere facilmente automatizzato. Nell'esempio illustrato viene utilizzato lo script descritto nella ["Replay Logs on Standby Database"](#). Lo script accetta un argomento che specifica il database che richiede un'operazione di riproduzione. Questo processo consente di utilizzare lo stesso script in una migrazione di più database.

```
[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
```

Cutover

Quando si è pronti a passare al nuovo ambiente, è necessario eseguire una sincronizzazione finale. Quando si lavora con i normali file system, è facile assicurarsi che il database migrato sia sincronizzato al 100% rispetto all'originale, poiché i log di ripristino originali vengono copiati e riprodotti. Con ASM non esiste un buon modo per farlo. È possibile recuperare facilmente solo i registri di archivio. Per assicurarsi che i dati non vadano persi, è necessario eseguire con attenzione l'arresto finale del database originale.

1. In primo luogo, la base di dati deve essere chiusa, garantendo che non vengano apportate modifiche. Questa chiusura potrebbe includere la disattivazione delle operazioni pianificate, la chiusura dei listener e/o la chiusura delle applicazioni.
2. Una volta eseguita questa operazione, la maggior parte dei DBA crea una tabella fittizia da utilizzare come indicatore dell'arresto.
3. Forzare l'archiviazione di un registro per assicurarsi che la creazione della tabella fittizia sia registrata nei registri di archivio. A tale scopo, eseguire i seguenti comandi:

```
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. Per copiare l'ultimo dei registri di archivio, eseguire i seguenti comandi. Il database deve essere disponibile ma non aperto.

```
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. Per copiare i log di archivio, eseguire i seguenti comandi:

```

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. Infine, riprodurre i log di archivio rimanenti sul nuovo server.

```

[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 questa fase, replicare tutti i dati. Il database è pronto per essere convertito da un database di standby a un database operativo attivo e quindi aperto.

```

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

```

8. Verificare la presenza della tabella fittizia e poi rilasciarla.

```

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.

```

Migrazione dei log di ripristino senza interruzioni

A volte, un database è organizzato correttamente in generale, ad eccezione dei registri di ripristino. Questo può accadere per molte ragioni, la più comune delle quali è correlata agli snapshot. Prodotti come SnapManager per Oracle, SnapCenter e il framework di gestione dello storage NetApp Snap Creator consentono il ripristino quasi istantaneo di un database, ma solo se vengono ripristinati i volumi dei file di dati. Se i log di redo condividono lo spazio con i file di dati, non è possibile eseguire la reversione in modo sicuro, poiché causerebbe la distruzione dei log di redo, probabilmente la perdita di dati. Pertanto, i log di ripristino devono essere spostati.

Questa procedura è semplice e può essere eseguita senza interruzioni.

Configurazione corrente del log di ripristino

1. Identificare il numero di gruppi di log di ripristino e i rispettivi numeri di gruppo.

```

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. Immettere le dimensioni dei registri di ripristino.

```

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

```

Creare nuovi registri

1. Per ogni log di ripristino, creare un nuovo gruppo con dimensioni e numero di membri corrispondenti.

```

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. Verificare la nuova configurazione.

```

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.

```

Rilasciare i vecchi registri

1. Rilasciare i vecchi registri (gruppi 1, 2 e 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. Se si verifica un errore che impedisce di rilasciare un registro attivo, forzare un passaggio al registro successivo per rilasciare il blocco e forzare un checkpoint globale. Fare riferimento al seguente esempio di questo processo. Il tentativo di rilasciare il gruppo di file di registro 2, che si trovava nella vecchia posizione, è stato negato perché in questo file di registro erano ancora presenti dati attivi.

```

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. Un'archiviazione dei log seguita da un punto di verifica consente di rilasciare il file di log.

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

4. Quindi, eliminare i log dal file system. Questo processo deve essere eseguito con estrema attenzione.

Copia dei dati host del database Oracle

Come per la migrazione a livello di database, la migrazione nel layer host fornisce un approccio indipendente dal vendor di soluzioni di storage.

In altre parole, talvolta "basta copiare i file" è l'opzione migliore.

Sebbene questo approccio a bassa tecnologia possa sembrare troppo semplice, offre comunque vantaggi significativi in quanto non è richiesto alcun software speciale e i dati originali rimangono intatti in tutta sicurezza durante il processo. Il limite principale è rappresentato dal fatto che la migrazione dei dati di una copia file causa interruzioni, poiché il database deve essere arrestato prima dell'inizio dell'operazione di copia. Non esiste un buon modo per sincronizzare le modifiche all'interno di un file, quindi i file devono essere completamente disattivati prima che la copia abbia inizio.

Se l'arresto richiesto da un'operazione di copia non è desiderabile, l'opzione successiva migliore basata su host è sfruttare un Logical Volume Manager (LVM). Esistono molte opzioni LVM, tra cui Oracle ASM, tutte con funzionalità simili, ma anche con alcune limitazioni che è necessario tenere in considerazione. Nella maggior parte dei casi, la migrazione può essere eseguita senza downtime e interruzioni.

Copia da filesystem a filesystem

L'utilità di una semplice operazione di copia non deve essere sottovalutata. Si tratta di un processo altamente affidabile che non richiede particolari competenze su sistemi operativi, database o sistemi storage. Inoltre, è molto sicuro perché non influisce sui dati originali. In genere, un amministratore di sistema modifica i file system di origine in modo che vengano montati in sola lettura e quindi riavvia un server per garantire che nessun elemento possa danneggiare i dati correnti. Il processo di copia può essere eseguito tramite script per garantire che venga eseguito il più rapidamente possibile senza il rischio di errori dell'utente. Poiché il tipo di i/o è un semplice trasferimento sequenziale dei dati, risulta estremamente efficiente in termini di larghezza di banda.

Nell'esempio seguente viene illustrata un'opzione per una migrazione sicura e rapida.

Ambiente

L'ambiente da migrare è il seguente:

- File system attuali

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

- Nuovi file system

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

Panoramica

Un DBA può migrare il database chiudendo semplicemente il database e copiando i file. Tuttavia, se occorre migrare molti database o ridurre al minimo il downtime, il processo può essere facilmente gestito tramite script. L'utilizzo di script riduce inoltre la possibilità di errori da parte dell'utente.

Gli script di esempio illustrati automatizzano le seguenti operazioni:

- Chiusura del database in corso
- Conversione dei file system esistenti in uno stato di sola lettura
- Copiare tutti i dati dai file system di origine a quelli di destinazione, mantenendo tutte le autorizzazioni dei file
- Smontaggio dei file system vecchi e nuovi
- Rimontaggio dei nuovi file system negli stessi percorsi dei file system precedenti

Procedura

1. Arrestare il database.

```

[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

```

- Convertire i file system in sola lettura. Questa operazione può essere eseguita più rapidamente utilizzando uno script, come illustrato nella ["Convertire il file system in sola lettura"](#).

```

[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

```

- Verificare che i file system siano ora di sola lettura.

```

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

```

- Sincronizzare il contenuto del file system con `rsync` comando.

```

[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. Smontare i vecchi file system e riposizionare i dati copiati. Questa operazione può essere eseguita più rapidamente utilizzando uno script, come illustrato nella ["Sostituire il file system"](#).

```
[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. Verificare che i nuovi file system siano in posizione.

```
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. Avviare il database.

```
[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
```

Cutover completamente automatizzato

Questo script di esempio accetta argomenti del SID del database seguiti da coppie di file system delimitate in comune. Per l'esempio sopra illustrato, il comando viene inviato come segue:

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

Quando viene eseguito, lo script di esempio tenta di eseguire la seguente sequenza. Termina se incontra un errore in qualsiasi fase:

1. Arrestare il database.
2. Convertire i file system correnti in stato di sola lettura.
3. Utilizzare ciascuna coppia di argomenti del file system delimitati da virgole e sincronizzare il primo file system con il secondo.
4. Smontare i file system precedenti.
5. Aggiornare `/etc/fstab` archiviare come segue:
 - a. Creare un backup in `/etc/fstab.bak`.
 - b. Annotare le voci precedenti per i file system precedenti e nuovi.
 - c. Creare una nuova voce per il nuovo file system che utilizza il vecchio punto di montaggio.
6. Montare i file system.
7. Avviare il database.

Il testo seguente fornisce un esempio di esecuzione per questo script:

```
[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
Successfully 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]#

```

Migrazione Oracle ASM spfile e passwd

Una difficoltà nel completare la migrazione che coinvolge ASM è rappresentata dallo spfile specifico per ASM e dal file delle password. Per impostazione predefinita, questi file di metadati critici vengono creati nel primo gruppo di dischi ASM definito. Se un particolare gruppo di dischi ASM deve essere evacuato e rimosso, il file spfile e la password che governano l'istanza ASM deve essere riposizionato.

Un altro caso d'utilizzo in cui potrebbe essere necessario trasferire questi file è durante una distribuzione di software di gestione del database, come SnapManager per Oracle o il plug-in SnapCenter Oracle. Una delle funzionalità di questi prodotti è il ripristino rapido di un database ripristinando lo stato dei LUN ASM che ospitano i file di dati. Per eseguire questa operazione, è necessario portare il gruppo di dischi ASM offline prima di eseguire un ripristino. Questo non è un problema, purché i file di dati di un determinato database siano isolati in un gruppo di dischi ASM dedicato.

Quando il gruppo di dischi contiene anche il file ASM spfile/passwd, l'unico modo per mettere il gruppo di dischi in modalità non in linea è arrestare l'intera istanza ASM. Si tratta di un processo di interruzione, il che significa che il file spfile/passwd dovrebbe essere riposizionato.

Ambiente

1. SID database = TOAST
2. File di dati correnti su +DATA
3. File di log e file di controllo correnti attivati +LOGS
4. Nuovi gruppi di dischi ASM stabiliti come +NEWDATA e. +NEWLOGS

Posizioni dei file spfile/passwd ASM

Il trasferimento di questi file può essere eseguito senza interruzione delle attività. Tuttavia, per motivi di sicurezza, NetApp consiglia di arrestare l'ambiente del database in modo da poter essere certi che i file siano stati spostati e che la configurazione sia stata aggiornata correttamente. Questa procedura deve essere ripetuta se su un server sono presenti più istanze ASM.

Identificare le istanze ASM

Identificare le istanze ASM in base ai dati registrati in oratab file. Le istanze di ASM sono indicate dal simbolo +.

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

Su questo server è presente un'istanza ASM denominata +ASM.

Assicurarsi che tutti i database siano chiusi

L'unico processo di smon visibile dovrebbe essere quello per l'istanza ASM in uso. La presenza di un altro processo di smon indica che un database è ancora in esecuzione.

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

L'unico processo di smon è l'istanza ASM stessa. Ciò significa che nessun altro database è in esecuzione ed è sicuro procedere senza il rischio di interrompere le operazioni del database.

Individuare i file

Identificare la posizione corrente del file spfile e della password di ASM utilizzando spget e pwget comandi.

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

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

I file si trovano entrambi alla base di +DATA gruppo di dischi.

Copiare i file

Copiare i file nel nuovo gruppo di dischi ASM con `spcopy` e `pwcopy` comandi. Se il nuovo gruppo di dischi è stato creato di recente ed è attualmente vuoto, potrebbe essere necessario montarlo per primo.

```
ASMCMD> mount NEWDATA
```

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

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

I file sono stati copiati da +DATA a. +NEWDATA.

Aggiornare l'istanza ASM

L'istanza ASM deve ora essere aggiornata per riflettere la modifica della posizione. Il `spset` e `pwset` i comandi aggiornano i metadati ASM richiesti per l'avvio del gruppo di dischi ASM.

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

Attivare ASM utilizzando i file aggiornati

A questo punto, l'istanza ASM utilizza ancora le posizioni precedenti di questi file. L'istanza deve essere riavviata per forzare una rilettura dei file dalle nuove posizioni e per rilasciare i blocchi sui file precedenti.

```
-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
```

Rimuovere i vecchi file spfile e password

Se la procedura è stata eseguita correttamente, i file precedenti non sono più bloccati e possono essere rimossi.

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

Copia da Oracle ASM a ASM

Oracle ASM è essenzialmente un volume manager e un file system combinati e leggeri. Poiché il file system non è facilmente visibile, è necessario utilizzare RMAN per eseguire operazioni di copia. Sebbene il processo di migrazione basato sulle copie sia sicuro e semplice, si traduce in un'interruzione. È possibile ridurre al minimo le interruzioni, ma non eliminarle completamente.

Se si desidera eseguire la migrazione senza interruzioni di un database basato su ASM, l'opzione migliore è sfruttare la capacità di ASM di riequilibrare le estensioni ASM nei nuovi LUN, eliminando al contempo i vecchi LUN. In genere, questo tipo di operazioni è sicuro e senza interruzioni, ma non offre alcun percorso di back-out. Se si riscontrano problemi di funzionamento o di prestazioni, l'unica opzione è quella di trasferire nuovamente i dati all'origine.

Questo rischio può essere evitato copiando il database nella nuova posizione piuttosto che spostare i dati, in modo che i dati originali non vengano toccati. Il database può essere completamente testato nella sua nuova posizione prima di entrare in funzione e il database originale è disponibile come opzione di fallback se vengono rilevati problemi.

Questa procedura è una delle numerose opzioni che interessano RMAN. È progettato per consentire un processo in due fasi in cui viene creato il backup iniziale e quindi sincronizzato successivamente tramite la riproduzione del registro. Questo processo è auspicabile per ridurre al minimo i tempi di inattività, in quanto consente al database di rimanere operativo e di distribuire i dati durante la copia di base iniziale.

Copia database

Oracle RMAN crea una copia di livello 0 (completa) del database di origine attualmente presente nel gruppo di dischi ASM +DATA alla nuova posizione su +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

```

Forzare l'interruttore del registro di archiviazione

È necessario forzare un'opzione del log di archivio per assicurarsi che i log di archivio contengano tutti i dati necessari per rendere la copia completamente coerente. Senza questo comando, i dati chiave potrebbero essere ancora presenti nei log di ripristino.

```

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

```

Arrestare il database di origine

L'interruzione inizia in questa fase perché il database viene arrestato e inserito in una modalità di sola lettura ad accesso limitato. Per arrestare il database di origine, eseguire i seguenti comandi:

```

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 ControlFile

È necessario eseguire il backup di controlfile nel caso in cui sia necessario interrompere la migrazione e ripristinare la posizione di archiviazione originale. Una copia del controlfile di backup non è richiesta al 100%, ma rende più semplice il processo di ripristino delle posizioni dei file di database nella posizione originale.

```

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

```

Aggiornamenti dei parametri

Il file spfile corrente contiene riferimenti ai file di controllo nelle posizioni correnti all'interno del vecchio gruppo di dischi ASM. Deve essere modificato, il che è fatto facilmente modificando una versione pfile intermedia.

```

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

```

Aggiornare pfile

Aggiornare tutti i parametri che fanno riferimento ai vecchi gruppi di dischi ASM per riflettere i nuovi nomi dei gruppi di dischi ASM. Quindi salvare il file pfile aggiornato. Assicurarsi che il db_create parametri presenti.

Nell'esempio seguente, i riferimenti a `+DATA` che sono stati modificati in `+NEWDATA` sono evidenziati in giallo. Due parametri chiave sono `db_create` parametri che creano nuovi file nella posizione corretta.

```
*.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'
```

Aggiorna il file `init.ora`

La maggior parte dei database basati su ASM utilizza un `init.ora` file che si trova in `$ORACLE_HOME/dbs` Directory, che è un punto di spfile sul gruppo di dischi ASM. Questo file deve essere reindirizzato a una posizione sul nuovo gruppo di dischi ASM.

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

Modificare questo file come segue:

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

Ricreazione del file dei parametri

spfile è ora pronto per essere popolato dai dati nel pfile modificato.

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

Avviare il database per iniziare a utilizzare il nuovo spfile

Avviare il database per assicurarsi che utilizzi ora il nuovo spfile creato e che eventuali ulteriori modifiche ai parametri di sistema siano registrate correttamente.

```

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

```

Ripristina controlfile

Il controlfile di backup creato da RMAN può anche essere ripristinato da RMAN direttamente nella posizione specificata nel nuovo spfile.

```

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

```

Montare il database e verificare l'uso del nuovo controlfile.

```

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

```

Riproduzione del registro

Il database utilizza attualmente i file di dati nella vecchia posizione. Prima di poter utilizzare la copia, è necessario sincronizzarla. È trascorso del tempo durante il processo di copia iniziale e le modifiche sono state registrate principalmente nei registri di archivio. Queste modifiche vengono replicate come segue:

1. Eseguire un backup incrementale RMAN, che contiene i registri di archivio.

```
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. Riprodurre nuovamente il registro.


```

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

```

Attivazione

Il controlfile ripristinato fa ancora riferimento ai file di dati nella posizione originale e contiene anche le informazioni di percorso per i file di dati copiati.

1. Per modificare i file di dati attivi, eseguire `switch database to copy` comando.

```

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"

```

I file di dati attivi sono ora i file di dati copiati, ma potrebbero comunque essere presenti modifiche nei log di ripristino finali.

2. Per riprodurre tutti i registri rimanenti, eseguire il `recover database` comando. Se il messaggio `media recovery complete` il processo è stato eseguito correttamente.

```

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

```

Questo processo ha modificato solo la posizione dei file di dati normali. I file di dati temporanei devono essere rinominati, ma non devono essere copiati perché sono solo temporanei. Il database è attualmente inattivo, pertanto non sono presenti dati attivi nei file di dati temporanei.

3. Per spostare i file di dati temporanei, identificarne prima la posizione.

```

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

```

4. Spostare i file di dati temporanei utilizzando un comando RMAN che imposta il nuovo nome per ciascun file di dati. Con Oracle Managed Files (OMF), il nome completo non è necessario; il gruppo di dischi ASM è sufficiente. Quando il database viene aperto, OMF si collega alla posizione appropriata nel gruppo di dischi ASM. Per spostare i file, eseguire i seguenti comandi:

```

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

```

Migrazione dei log di ripristino

Il processo di migrazione è quasi completo, ma i log di ripristino si trovano ancora nel gruppo di dischi ASM originale. I log di ripristino non possono essere spostati direttamente. Viene invece creata una nuova serie di log di ripristino che viene aggiunta alla configurazione, seguita da una rimozione dei log precedenti.

1. Identificare il numero di gruppi di log di ripristino e i rispettivi numeri di gruppo.

```

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. Immettere le dimensioni dei registri di ripristino.

```

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

```

3. Per ogni log di ripristino, creare un nuovo gruppo con una configurazione corrispondente. Se non si utilizza OMF, è necessario specificare il percorso completo. Questo è anche un esempio che utilizza `db_create_online_log` parametri. Come mostrato in precedenza, questo parametro era impostato su `+NEWLOGS`. Questa configurazione consente di utilizzare i seguenti comandi per creare nuovi registri online senza dover specificare un percorso di file o un gruppo di dischi ASM specifico.

```

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. Aprire il database.

```

SQL> alter database open;
Database altered.

```

5. Rilasciare i vecchi registri.

```

RMAN> alter database drop logfile group 1;
Statement processed

```

6. Se si verifica un errore che impedisce di rilasciare un registro attivo, forzare un passaggio al registro

successivo per rilasciare il blocco e forzare un checkpoint globale. Di seguito è riportato un esempio. Il tentativo di rilasciare il gruppo di file di registro 3, che si trovava nella vecchia posizione, è stato negato perché in questo file di registro erano ancora presenti dati attivi. L'archiviazione di un registro dopo un punto di verifica consente di eliminare il file di registro.

```

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. Esaminare l'ambiente per assicurarsi che tutti i parametri basati sulla posizione siano aggiornati.

```

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. Nello script seguente viene illustrato come semplificare questo processo:

```

[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. Se i gruppi di dischi ASM sono stati completamente evacuati, è possibile smontarli con `asmcmd`. Tuttavia, in molti casi i file appartenenti ad altri database o al file ASM `spfile/passwd` potrebbero essere ancora presenti.

```

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

```

Copia da Oracle ASM al file system

La procedura di copia da Oracle ASM a file system è molto simile alla procedura di copia da ASM a ASM, con vantaggi e restrizioni simili. La differenza principale è la sintassi dei vari comandi e parametri di configurazione quando si utilizza un file system visibile anziché un gruppo di dischi ASM.

Copia database

Oracle RMAN viene utilizzato per creare una copia di livello 0 (completa) del database di origine attualmente presente nel gruppo di dischi ASM `+DATA` alla nuova posizione su `/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

```

Forzare l'interruttore del registro di archiviazione

È necessario forzare lo switch del log di archivio per assicurarsi che i log di archivio contengano tutti i dati necessari per rendere la copia completamente coerente. Senza questo comando, i dati chiave potrebbero essere ancora presenti nei log di ripristino. Per forzare un'opzione del log di archivio, eseguire il comando seguente:

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

Arrestare il database di origine

L'interruzione inizia in questa fase perché il database viene arrestato e inserito in una modalità di sola lettura ad accesso limitato. Per arrestare il database di origine, eseguire i seguenti comandi:

```
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 ControlFile

Eseguire il backup dei file di controllo nel caso in cui sia necessario interrompere la migrazione e ripristinare la posizione di archiviazione originale. Una copia del controlfile di backup non è richiesta al 100%, ma rende più semplice il processo di ripristino delle posizioni dei file di database nella posizione originale.

```
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
```

Aggiornamenti dei parametri

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

Aggiornare pfile

Tutti i parametri che fanno riferimento ai vecchi gruppi di dischi ASM devono essere aggiornati e, in alcuni casi, eliminati quando non sono più rilevanti. Aggiornarli per riflettere i nuovi percorsi del file system e salvare il file pfile aggiornato. Assicurarsi che sia elencato il percorso di destinazione completo. Per aggiornare questi parametri, eseguire i seguenti comandi:

```
*.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'
```

Disattivare il file init.ora originale

Questo file si trova in \$ORACLE_HOME/dbs Ed è in genere in un pfile che funge da puntatore a spfile sul gruppo di dischi ASM. Per assicurarsi che spfile originale non sia più utilizzato, rinominarlo. Non eliminarlo, tuttavia, perché questo file è necessario se la migrazione deve essere interrotta.

```
[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]$
```

Ricreazione del file dei parametri

Questa è la fase finale del trasferimento di spfile. Il file spfile originale non viene più utilizzato e il database viene avviato (ma non montato) utilizzando il file intermedio. Il contenuto di questo file può essere scritto nella nuova posizione spfile come segue:


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

Avviare il database per iniziare a utilizzare il nuovo spfile

È necessario avviare il database per rilasciare i blocchi sul file intermedio e avviare il database utilizzando solo il nuovo file spfile. L'avvio del database dimostra inoltre che la nuova posizione di spfile è corretta e che i suoi dati sono validi.

```
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
```

Ripristina controlfile

È stato creato un controlfile di backup nel percorso `/tmp/TOAST.ctrl` nelle fasi precedenti della procedura. Il nuovo spfile definisce le posizioni controlfile come `/logfs/TOAST/ctrl/controlfile1.ctrl` e `/logfs/TOAST/redo/controlfile2.ctrl`. Tuttavia, tali file non esistono ancora.

1. Questo comando ripristina i dati controlfile nei percorsi definiti in spfile.

```
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.ctrl  
output file name=/logs/TOAST/redo/control02.ctrl  
Finished restore at 13-MAY-16
```

2. Eseguire il comando mount in modo che i file di controllo vengano rilevati correttamente e contengano dati validi.

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

Per convalidare `control_files` eseguire il seguente comando:

```
SQL> show parameter control_files;
NAME                                TYPE                                VALUE
-----                                -
control_files                       string
/logs/TOAST/arch/control01.ctl
,
/logs/TOAST/redo/control02.c
tl
```

Riproduzione del registro

Il database sta attualmente utilizzando i file di dati nella vecchia posizione. Prima di poter utilizzare la copia, è necessario sincronizzare i file di dati. È trascorso del tempo durante il processo di copia iniziale e le modifiche sono state registrate principalmente nei registri di archivio. Queste modifiche vengono replicate nei due passaggi seguenti.

1. Eseguire un backup incrementale RMAN, che contiene i registri di archivio.

```
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. Riprodurre i registri.

```

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

```

Attivazione

Il controlfile ripristinato fa ancora riferimento ai file di dati nella posizione originale e contiene anche le informazioni di percorso per i file di dati copiati.

1. Per modificare i file di dati attivi, eseguire `switch database to copy` comando:

```

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. Sebbene i file di dati debbano essere completamente coerenti, è necessario eseguire un passaggio finale per riprodurre le modifiche rimanenti registrate nei registri di ripristino online. Utilizzare `recover database` comando per riprodurre queste modifiche e rendere la copia identica al 100% all'originale. Tuttavia, la copia non è ancora aperta.

```

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

```

Spostare i file di dati temporanei

1. Identificare la posizione dei file di dati temporanei ancora in uso sul gruppo di dischi originale.

```

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

```

2. Per spostare i file di dati, eseguire i seguenti comandi. Se ci sono molti tempfile, utilizzare un editor di testo per creare il comando RMAN e quindi tagliarlo e incollarlo.

```

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

```

Migrazione dei log di ripristino

Il processo di migrazione è quasi completo, ma i log di ripristino si trovano ancora nel gruppo di dischi ASM originale. I log di ripristino non possono essere spostati direttamente. Al contrario, viene creata e aggiunta alla configurazione una nuova serie di log di ripristino, in seguito a una perdita dei vecchi log.

1. Identificare il numero di gruppi di log di ripristino e i rispettivi numeri di gruppo.

```

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. Immettere le dimensioni dei registri di ripristino.

```

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

```

3. Per ogni log di ripristino, creare un nuovo gruppo utilizzando le stesse dimensioni del gruppo di log di ripristino corrente utilizzando la nuova posizione del file system.

```

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. Rimuovere i vecchi gruppi di file di registro che si trovano ancora nell'archivio precedente.

```

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. Se si verifica un errore che blocca l'eliminazione di un registro attivo, forzare un passaggio al registro successivo per rilasciare il blocco e forzare un punto di verifica globale. Di seguito è riportato un esempio. Il tentativo di rilasciare il gruppo di file di registro 3, che si trovava nella vecchia posizione, è stato negato

perché in questo file di registro erano ancora presenti dati attivi. L'archiviazione dei log seguita da un punto di verifica consente l'eliminazione dei file di log.

```

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. Esaminare l'ambiente per assicurarsi che tutti i parametri basati sulla posizione siano aggiornati.

```

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. Nel seguente script viene illustrato come semplificare questo processo.

```

[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. Se i gruppi di dischi ASM sono stati completamente evacuati, è possibile smontarli con `asmcmd`. In molti casi, i file appartenenti ad altri database o al file ASM `spfile/passwd` possono essere ancora presenti.

```

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

```

Procedura di pulizia del file di dati

Il processo di migrazione potrebbe generare file di dati con sintassi lunga o criptica, a seconda del modo in cui è stato utilizzato Oracle RMAN. Nell'esempio illustrato, il backup è stato eseguito con il formato file di `/oradata/TOAST/%U. %U` Indica che RMAN deve creare un nome univoco predefinito per ciascun file di dati. Il risultato è simile a quanto illustrato nel testo seguente. I nomi tradizionali dei file di dati sono incorporati nei nomi. Questo può essere ripulito utilizzando l'approccio basato su script illustrato nella ["Pulitura della migrazione ASM"](#).

```

[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;

```

Ribilanciamento di Oracle ASM

Come indicato in precedenza, è possibile eseguire la migrazione trasparente di un gruppo di dischi Oracle ASM in un nuovo sistema di storage utilizzando il processo di ribilanciamento. Riassumendo, il processo di ribilanciamento richiede l'aggiunta di LUN di dimensioni uguali al gruppo esistente di LUN, seguita da un'operazione di disgregazione del LUN precedente. Oracle ASM riposiziona automaticamente i dati sottostanti nel nuovo storage in un layout ottimale e, al termine, rilascia i vecchi LUN.

Il processo di migrazione utilizza un i/o sequenziale efficiente e non causa generalmente un'interruzione delle performance, ma la velocità di migrazione può essere rallentata quando necessario.

Identificazione dei dati da migrare

```

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

```


Creazione di nuovi LUN

Creare nuovi LUN delle stesse dimensioni e impostare l'appartenenza a utenti e gruppi come richiesto. I LUN devono essere visualizzati come CANDIDATE dischi.

```
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
```

Aggiungere nuovi LUN

Anche se è possibile eseguire tutte le operazioni di aggiunta e rilascio, in genere è più semplice aggiungere nuovi LUN in due passaggi. Innanzitutto, aggiungere i nuovi LUN al gruppo di dischi. Questo passaggio comporta la migrazione di metà delle estensioni dai LUN ASM correnti ai nuovi LUN.

La potenza di riequilibrio indica la velocità di trasferimento dei dati. Più alto è il numero, più alto è il parallelismo del trasferimento dei dati. La migrazione viene eseguita con efficienti operazioni di i/o sequenziali che hanno scarse probabilità di causare problemi di performance. Tuttavia, se lo si desidera, il potere di riequilibrio di una migrazione in corso può essere regolato con `alter diskgroup [name] rebalance power [level]` comando. Le migrazioni tipiche utilizzano un valore di 5.

```
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.
```

Funzionamento del monitor

È possibile monitorare e gestire un'operazione di ribilanciamento in più modi. Per questo esempio è stato utilizzato il comando seguente.

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

Una volta completata la migrazione, non vengono segnalate operazioni di ribilanciamento.

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

LUN meno recenti

La migrazione è ormai a metà strada. Potrebbe essere opportuno eseguire alcuni test delle prestazioni di base per assicurarsi che l'ambiente sia sano. Dopo la conferma, è possibile spostare i dati rimanenti eliminando i vecchi LUN. Tenere presente che ciò non determina il rilascio immediato dei LUN. L'operazione di rilascio indica ad Oracle ASM di riposizionare prima le estensioni e quindi rilasciare il LUN.

```
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.
```

Funzionamento del monitor

L'operazione di ribilanciamento può essere monitorata e gestita in più modi. Per questo esempio è stato utilizzato il seguente comando:

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

Una volta completata la migrazione, non vengono segnalate operazioni di ribilanciamento.

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

Rimuovere i vecchi LUN

Prima di rimuovere i vecchi LUN dal gruppo di dischi, è necessario eseguire un controllo finale dello stato dell'intestazione. Dopo il rilascio di un LUN da ASM, non viene più elencato un nome e lo stato dell'intestazione viene elencato come FORMER. Questo indica che questi LUN possono essere rimossi in modo sicuro dal sistema.

```
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.
```

Migrazione LVM

La procedura qui presentata mostra i principi di una migrazione basata su LVM di un gruppo di volumi chiamato `datavg`. Gli esempi sono tratti da Linux LVM, ma i principi si applicano ugualmente a AIX, HP-UX e VxVM. I comandi precisi possono variare.

1. Identificare i LUN attualmente presenti in `datavg` gruppo di volumi.

```
[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. Creazione di nuovi LUN di dimensioni fisiche identiche o leggermente superiori e definizione di volumi fisici.

```

[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. Aggiungere i nuovi volumi al gruppo di volumi.

```

[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. Eseguire il pvmove Comando per spostare le estensioni di ogni LUN corrente nel nuovo LUN. Il - i [seconds] l'argomento controlla l'avanzamento dell'operazione.

```
[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. Una volta completato questo processo, rimuovere i LUN precedenti dal gruppo di volumi utilizzando `vgreduce` comando. Se l'operazione ha esito positivo, è ora possibile rimuovere il LUN dal sistema in modo sicuro.

```
[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"
```

Importazione LUN esterne

Migrazione di Oracle con FLI: Pianificazione

Le procedure per la migrazione delle risorse SAN utilizzando FLI sono documentate in NetApp ["TR-4380: Migrazione SAN con importazione di LUN esterne"](#).

Dal punto di vista del database e dell'host, non sono necessarie operazioni speciali. Dopo l'aggiornamento delle zone FC e la disponibilità dei LUN su ONTAP, LVM dovrebbe essere in grado di leggere i metadati LVM dai LUN. Inoltre, i gruppi di volumi sono pronti per l'uso senza ulteriori passaggi di configurazione. Rari casi, gli ambienti potrebbero includere file di configurazione con hard-code e riferimenti allo storage array precedente. Ad esempio, un sistema Linux che includeva `/etc/multipath.conf` Le regole che fanno riferimento a un WWN di un dato dispositivo devono essere aggiornate per riflettere le modifiche introdotte da FLI.



Fare riferimento alla matrice di compatibilità NetApp per informazioni sulle configurazioni supportate. Se il proprio ambiente non è incluso, contattare il rappresentante NetApp per assistenza.

Questo esempio mostra la migrazione di LUN ASM e LVM ospitati su un server Linux. FLI è supportato su altri sistemi operativi e, sebbene i comandi sul lato host possano differire, i principi sono gli stessi e le procedure ONTAP sono identiche.

Identificare i LUN LVM

La prima fase della preparazione consiste nell'identificare i LUN da migrare. Nell'esempio mostrato qui, due file system basati su SAN sono montati su `/orabin` e `/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
```

Il nome del gruppo di volumi può essere estratto dal nome del dispositivo, che utilizza il formato (nome del gruppo di volumi)-(nome del volume logico). In questo caso, viene chiamato il gruppo di volumi `sanvg`.

Il `pvdisplay` Il comando può essere utilizzato come segue per identificare i LUN che supportano questo gruppo di volumi. In questo caso, sono presenti 10 LUN che compongono il `sanvg` gruppo di volumi.

```
[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
```

Identificare i LUN ASM

Anche i LUN ASM devono essere migrati. Per ottenere il numero di LUN e percorsi LUN da `sqlplus` come utente `sysasm`, eseguire il comando seguente:

```

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>

```

Modifiche alla rete FC

L'ambiente corrente contiene 20 LUN da migrare. Aggiornare la SAN corrente in modo che ONTAP possa accedere ai LUN correnti. I dati non sono ancora stati migrati, ma ONTAP deve leggere le informazioni di configurazione dalle LUN correnti per creare la nuova home page per quei dati.

Almeno una porta HBA sul sistema AFF/FAS deve essere configurata come porta Initiator. Inoltre, le zone FC devono essere aggiornate in modo che ONTAP possa accedere alle LUN sullo storage array esterno. Alcuni storage array hanno configurato il masking dei LUN, che limita i WWN che possono accedere a una determinata LUN. In tal caso, è necessario aggiornare anche il masking dei LUN per garantire l'accesso ai WWN di ONTAP.

Al termine di questa operazione, ONTAP dovrebbe essere in grado di visualizzare l'array di archiviazione esterno con `storage array show` comando. Il campo chiave restituito è il prefisso utilizzato per identificare il LUN esterno sul sistema. Nell'esempio seguente, i LUN dell'array esterno `FOREIGN_1` Appare in ONTAP usando il prefisso di `FOR-1`.

Identificare un array esterno

```

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

```

Identificare i LUN esterni

I LUN possono essere elencati passando l'array-name al `storage disk show` comando. I dati restituiti vengono referenziati più volte durante la procedura di migrazione.


```

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$HuVWBW
FOR-1.4   800DT$HuVWB Y
FOR-1.5   800DT$HuVWB/
FOR-1.6   800DT$HuVWBa
FOR-1.7   800DT$HuVWBd
FOR-1.8   800DT$HuVWBb
FOR-1.9   800DT$HuVWBc
FOR-1.10  800DT$HuVWB e
FOR-1.11  800DT$HuVWBf
FOR-1.12  800DT$HuVWBg
FOR-1.13  800DT$HuVWB i
FOR-1.14  800DT$HuVWBh
FOR-1.15  800DT$HuVWBj
FOR-1.16  800DT$HuVWBk
FOR-1.17  800DT$HuVWBm
FOR-1.18  800DT$HuVWB l
FOR-1.19  800DT$HuVWB o
FOR-1.20  800DT$HuVWB n
20 entries were displayed.
Cluster01::>

```

Registrazione LUN di array esterni come candidati di importazione

Le LUN esterne vengono inizialmente classificate come qualsiasi tipo di LUN specifico. Prima di poter importare i dati, i LUN devono essere contrassegnati come esterni e quindi come candidati al processo di importazione. Questo passaggio viene completato passando il numero di serie a. `storage disk modify`, come illustrato nell'esempio seguente. Si noti che questa procedura etichetta solo il LUN come estraneo all'interno di ONTAP. Nessun dato viene scritto nella LUN esterna stessa.

```

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$HuVWBn} -is
-foreign true
Cluster01::*> storage disk modify {-serial-number 800DT$HuVWB o} -is
-foreign true
Cluster01::*>

```

Creazione di volumi per l'hosting di LUN migrati

Per ospitare le LUN migrate è necessario un volume. La configurazione esatta dei volumi dipende dal piano generale per sfruttare le funzionalità di ONTAP. In questo esempio, i LUN ASM vengono posizionati in un volume e i LUN LVM in un secondo volume. In questo modo, puoi gestire le LUN come gruppi indipendenti per scopi come il tiering, la creazione di snapshot o l'impostazione di controlli della qualità del servizio.

Impostare `snapshot-policy` a `none`. Il processo di migrazione può comportare un notevole ricambio dei dati. Pertanto, potrebbe verificarsi un notevole aumento del consumo di spazio se le istantanee vengono create accidentalmente perché i dati indesiderati vengono acquisiti nelle istantanee.

```
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::>
```

Creare LUN ONTAP

Una volta creati i volumi, è necessario creare i nuovi LUN. In genere, la creazione di un LUN richiede all'utente di specificare tali informazioni come la dimensione LUN, ma in questo caso l'argomento del disco esterno viene passato al comando. Di conseguenza, ONTAP replica i dati di configurazione LUN correnti dal numero di serie specificato. Utilizza inoltre la geometria del LUN e i dati della tabella delle partizioni per regolare l'allineamento delle LUN e stabilire prestazioni ottimali.

In questo passaggio, i numeri di serie devono essere referenziati rispetto all'array esterno per assicurarsi che il LUN esterno corretto corrisponda al nuovo LUN corretto.

```
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)
```

Creare relazioni di importazione

I LUN sono stati creati ma non sono configurati come destinazione di replica. Prima di eseguire questo

passaggio, i LUN devono essere messi offline. Questo passaggio aggiuntivo è progettato per proteggere i dati dagli errori dell'utente. Se ONTAP consentisse di eseguire una migrazione su un LUN online, rischierebbe di provocare la sovrascrittura dei dati attivi con un errore tipografico. Questa fase aggiuntiva, che obbliga l'utente a portare un LUN offline, consente di verificare se viene utilizzato il LUN di destinazione corretto come destinazione della migrazione.

```
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
```

Una volta che i LUN sono offline, è possibile stabilire la relazione di importazione passando il numero di serie del LUN esterno a. `lun import create` comando.

```
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::*>
```

Una volta stabilite tutte le relazioni di importazione, è possibile riportare online i LUN.

```

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::*>

```

Crea gruppo iniziatore

Un gruppo iniziatore (igroup) fa parte dell'architettura di mascheramento LUN di ONTAP. Un LUN appena creato non è accessibile a meno che non venga concesso per la prima volta l'accesso a un host. A tale scopo, creare un igroup in cui siano elencati i nomi WWN FC o iSCSI Initiator a cui è necessario concedere l'accesso. Al momento della scrittura del report, FLI era supportato solo per LUN FC. Tuttavia, la conversione in post-migrazione iSCSI è un'attività semplice, come illustrato nella ["Conversione protocollo"](#).

In questo esempio, viene creato un igroup che contiene due WWN corrispondenti alle due porte disponibili sull'HBA dell'host.

```

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

```

Mappare nuovi LUN all'host

Dopo la creazione di igroup, i LUN vengono quindi mappati all'igroup definito. Questi LUN sono disponibili solo per i WWN inclusi in questo igroup. In questa fase del processo di migrazione, NetApp presume che l'host non sia stato sottoposto a zoning in ONTAP. Questo è importante perché se l'host è contemporaneamente collegato all'array esterno e al nuovo sistema ONTAP, vi è il rischio che su ogni array possano essere rilevati LUN con lo stesso numero di serie. Questa situazione potrebbe causare malfunzionamenti del multipath o danni ai dati.

```

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::*>

```

Migrazione Oracle con FLI: Cutover

Una parte delle interruzioni durante l'importazione di LUN esterne è inevitabile a causa della necessità di modificare la configurazione di rete FC. Tuttavia, l'interruzione non deve durare più a lungo del tempo necessario per riavviare l'ambiente di database e

aggiornare lo zoning FC per passare dalla connettività FC dell'host al ONTAP.

Questo processo può essere riassunto come segue:

1. Quietare di tutta l'attività LUN sui LUN esterni.
2. Reindirizzare le connessioni FC dell'host al nuovo sistema ONTAP.
3. Attivare il processo di importazione.
4. Rilevare nuovamente i LUN.
5. Riavviare il database.

Non è necessario attendere il completamento del processo di migrazione. Non appena inizia la migrazione di un determinato LUN, questo è disponibile su ONTAP e può fornire dati durante il processo di copia dei dati. Tutte le letture vengono passate alla LUN esterna e tutte le scritture vengono scritte in modo sincrono su entrambi gli array. L'operazione di copia è molto veloce e l'overhead del reindirizzamento del traffico FC è minimo, per cui qualsiasi impatto sulle performance deve essere transitorio e minimo. In caso di problemi, è possibile ritardare il riavvio dell'ambiente fino al completamento del processo di migrazione e all'eliminazione delle relazioni di importazione.

Chiudere il database

Il primo passo per chiudere l'ambiente in questo esempio è arrestare il database.

```
[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>
```

Chiudere i servizi di rete

Uno dei file system basati su SAN oggetto della migrazione include anche i servizi Oracle ASM. La disattivazione dei LUN sottostanti richiede lo smontaggio dei file system, il che a sua volta significa l'arresto di tutti i processi con file aperti su questo file system.

```

[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]$

```

Smontare i file system

Se tutti i processi vengono arrestati, l'operazione `umount` ha esito positivo. Se l'autorizzazione viene negata, è necessario che sul file system sia presente un processo con blocco. Il `fuser` command può aiutare a identificare questi processi.

```

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

```

Disattivare i gruppi di volumi

Una volta smontati tutti i file system di un dato gruppo di volumi, è possibile disattivare il gruppo di volumi.

```

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

```

Modifiche alla rete FC

È ora possibile aggiornare le zone FC per rimuovere tutti gli accessi dall'host all'array esterno e stabilire l'accesso a ONTAP.

Avviare il processo di importazione

Per avviare i processi di importazione LUN, eseguire `lun import start` comando.

Se hai bisogno di una migrazione online, procedi con il rilevamento dei LUN nella nuova sede e attiva i servizi.

Eeguire la scansione delle modifiche al dispositivo SCSI

Nella maggior parte dei casi, l'opzione più semplice per ritrovare nuove LUN è riavviare l'host. In questo modo, si rimuovono automaticamente i vecchi dispositivi obsoleti, si rilevano correttamente tutti i nuovi LUN e si creano dispositivi associati come i dispositivi multipathing. L'esempio qui mostra una procedura completamente online a scopo dimostrativo.

Attenzione: Prima di riavviare un host, assicurarsi che tutte le voci in `/etc/fstab` Il riferimento alle risorse SAN migrate verrà commentato. Se questa operazione non viene eseguita e si verificano problemi con l'accesso LUN, il sistema operativo potrebbe non avviarsi. Questa situazione non danneggia i dati. Tuttavia, può essere molto scomodo avviare in modalità rescue o in una modalità simile e correggere `/etc/fstab` In modo che il sistema operativo possa essere avviato per consentire la risoluzione dei problemi.

I LUN della versione di Linux utilizzata in questo esempio possono essere rianalizzati con `rescan-scsi-bus.sh` comando. Se il comando viene eseguito correttamente, nell'output viene visualizzato ogni percorso LUN. L'output può essere difficile da interpretare, ma, se la configurazione di zoning e igroup era corretta, molti LUN dovrebbero apparire che includono un `NETAPP` stringa fornitore.


```

[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.

```

Verificare la presenza di dispositivi multipercorso

Il processo di rilevamento LUN attiva anche la ricreazione dei dispositivi multipath, ma è noto che il driver multipathing Linux presenta problemi occasionali. L'output di `multipath - ll` dovrebbe essere controllato per verificare che l'output sia come previsto. Per esempio, l'uscita seguente mostra dispositivi multipercorso associati a A. NETAPP stringa fornitore. Ciascun dispositivo dispone di quattro percorsi, di cui due con priorità 50 e due con priorità 10. Anche se l'output esatto può variare con diverse versioni di Linux, questo risultato

sembra come previsto.



Fare riferimento alla documentazione delle utilità host per la versione di Linux utilizzata per verificare che `/etc/multipath.conf` le impostazioni sono corrette.

```
[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 sdbx 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
```

Riattivare il gruppo di volumi LVM

Se i LUN LVM sono stati rilevati correttamente, l' `vgchange --activate y` il comando dovrebbe riuscire. Questo è un buon esempio del valore di un volume manager logico. Una modifica del WWN di una LUN o anche di un numero di serie non è importante perché i metadati del gruppo di volumi vengono scritti sul LUN stesso.

Il sistema operativo ha eseguito la scansione dei LUN e ha rilevato una piccola quantità di dati scritti sul LUN che lo identifica come volume fisico appartenente a `sanvg` volumegroup. Successivamente, ha costruito tutti i dispositivi necessari. È sufficiente riattivare il gruppo di volumi.

```
[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
```

Rimontare i file system

Dopo la riattivazione del gruppo di volumi, i file system possono essere montati con tutti i dati originali intatti. Come indicato in precedenza, i file system sono completamente operativi anche se la replica dei dati è ancora attiva nel gruppo back.

```

[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					

Ripetere la scansione per i dispositivi ASM

I dispositivi ASMLib dovrebbero essere stati risSelectedzionati al momento della nuova scansione dei dispositivi SCSI. La riscoperta può essere verificata online riavviando ASMLib e quindi eseguendo la scansione dei dischi.



Questa fase è pertinente solo alle configurazioni ASM in cui viene utilizzato ASMLib.

Attenzione: Se non viene utilizzato ASMLib, il `/dev/mapper` i dispositivi dovrebbero essere stati ricreati automaticamente. Tuttavia, le autorizzazioni potrebbero non essere corrette. È necessario impostare autorizzazioni speciali sui dispositivi sottostanti per ASM in assenza di ASMLib. Questa operazione viene solitamente eseguita tramite voci speciali in entrambi `/etc/multipath.conf` oppure `udev` o eventualmente in entrambi i set di regole. È possibile che questi file debbano essere aggiornati per riflettere le modifiche apportate all'ambiente in termini di numeri WWN o di serie per assicurarsi che i dispositivi ASM dispongano ancora delle autorizzazioni corrette.

In questo esempio, il riavvio di ASMLib e la scansione dei dischi mostrano gli stessi 10 LUN ASM dell'ambiente originale.

```
[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"
```

Riavviare i servizi di rete

Ora che i dispositivi LVM e ASM sono online e disponibili, è possibile riavviare i servizi grid.

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

Riavviare il database

Dopo aver riavviato i servizi di griglia, è possibile avviare il database. Potrebbe essere necessario attendere alcuni minuti affinché i servizi ASM diventino completamente disponibili prima di provare ad avviare il database.

```
[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>
```

Migrazione Oracle con FLI - completamento

Dal punto di vista dell'host, la migrazione è completa, ma l'i/o viene ancora servito dall'array esterno fino a quando le relazioni di importazione non vengono eliminate.

Prima di eliminare le relazioni, è necessario confermare che il processo di migrazione è completo per tutte le LUN.

```

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$HuVWBe /vol/new_asm/LUN9  completed
vserver1  800DT$HuVWBf /vol/new_lvm/LUN0  completed
vserver1  800DT$HuVWBg /vol/new_lvm/LUN1  completed
vserver1  800DT$HuVWBh /vol/new_lvm/LUN2  completed
vserver1  800DT$HuVWBj /vol/new_lvm/LUN3  completed
vserver1  800DT$HuVWBk /vol/new_lvm/LUN4  completed
vserver1  800DT$HuVWBm /vol/new_lvm/LUN5  completed
vserver1  800DT$HuVWBn /vol/new_lvm/LUN6  completed
vserver1  800DT$HuVWBp /vol/new_lvm/LUN7  completed
vserver1  800DT$HuVWBq /vol/new_lvm/LUN8  completed
vserver1  800DT$HuVWBs /vol/new_lvm/LUN9  completed
20 entries were displayed.

```

Elimina relazioni di importazione

Al termine del processo di migrazione, eliminare la relazione di migrazione. Dopo aver fatto ciò, l'i/o viene servito esclusivamente dalle unità su ONTAP.

```

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

```

Annullare la registrazione di LUN esterne

Infine, modificare il disco per rimuovere is-foreign designazione.

```

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::*>

```

Migrazione Oracle con FLI: Conversione del protocollo

La modifica del protocollo utilizzato per accedere a un LUN è un requisito comune.

In alcuni casi, fa parte di una strategia globale di migrazione dei dati nel cloud. TCP/IP è il protocollo del cloud e il passaggio da FC a iSCSI facilita la migrazione in vari ambienti cloud. In altri casi, iSCSI potrebbe essere desiderabile per sfruttare i costi ridotti di un IP SAN. A volte, una migrazione potrebbe utilizzare un protocollo diverso come misura temporanea. Ad esempio, se un array esterno e LUN basati su ONTAP non possono coesistere sugli stessi HBA, è possibile utilizzare LUN iSCSI abbastanza a lungo da copiare i dati dal vecchio array. Dopo la rimozione dei vecchi LUN dal sistema, è possibile riconvertirli in FC.

La seguente procedura illustra la conversione da FC a iSCSI, ma i principi generali si applicano a una conversione da iSCSI a FC inversa.

Installare iSCSI Initiator

La maggior parte dei sistemi operativi include un iniziatore iSCSI software per impostazione predefinita, ma se non è incluso, può essere facilmente installato.

```

[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.e17 will be
updated
--> Processing Dependency: iscsi-initiator-utils = 6.2.0.873-32.e17 for
package: iscsi-initiator-utils-iscsiuio-6.2.0.873-32.e17.x86_64
---> Package iscsi-initiator-utils.x86_64 0:6.2.0.873-32.0.2.e17 will be
an update
--> Running transaction check
---> Package iscsi-initiator-utils-iscsiuio.x86_64 0:6.2.0.873-32.e17 will
be updated
---> Package iscsi-initiator-utils-iscsiuio.x86_64 0:6.2.0.873-32.0.2.e17
will be an update

```


--> Finished Dependency Resolution

Dependencies Resolved

=====
===

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

Size

=====
===

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_64 | 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_64
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_64
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 ~]#
```

Identificare il nome dell'iniziatore iSCSI

Durante il processo di installazione viene generato un nome iSCSI initiator univoco. Su Linux, si trova in `/etc/iscsi/initiatorname.iscsi` file. Questo nome viene utilizzato per identificare l'host sulla SAN IP.

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

Creare un nuovo gruppo iniziatore

Un gruppo iniziatore (igroup) fa parte dell'architettura di mascheramento LUN di ONTAP. Un LUN appena creato non è accessibile a meno che non venga concesso per la prima volta l'accesso a un host. Questa operazione viene eseguita creando un igroup che elenca i nomi WWN FC o iniziatori iSCSI che richiedono l'accesso.

In questo esempio, viene creato un igroup che contiene l'iniziatore iSCSI dell'host Linux.

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

Chiudere l'ambiente

Prima di modificare il protocollo LUN, è necessario disattivare completamente i LUN. Tutti i database di uno dei LUN da convertire devono essere chiusi, i file system devono essere smontati e i gruppi di volumi devono essere disattivati. Se si utilizza ASM, assicurarsi che il gruppo di dischi ASM sia smontato e chiudere tutti i servizi della griglia.

Rimuovere la mappatura dei LUN dalla rete FC

Una volta terminate completamente le LUN, rimuovere le mappature dall'igroup FC originale.

```
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
```

Eeguire nuovamente il mapping dei LUN alla rete IP

Concedere l'accesso a ogni LUN al nuovo gruppo di iniziatori basati su iSCSI.

```
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::*>
```

Rilevamento delle destinazioni iSCSI

Il rilevamento iSCSI richiede due fasi. La prima è scoprire le destinazioni, che non è la stessa cosa per scoprire un LUN. Il `iscsiadm` il comando mostrato di seguito verifica il gruppo di portali specificato da `-p argument` Memorizza un elenco di tutti gli indirizzi IP e le porte che offrono servizi iSCSI. In questo caso, vi sono quattro indirizzi IP con servizi iSCSI sulla porta predefinita 3260.



Il completamento di questo comando può richiedere alcuni minuti se non è possibile raggiungere uno qualsiasi degli indirizzi IP di destinazione.

```
[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
```

Rilevamento delle LUN iSCSI

Dopo aver rilevato le destinazioni iSCSI, riavviare il servizio iSCSI per rilevare i LUN iSCSI disponibili e creare i dispositivi associati, ad esempio i dispositivi multipath o ASMLib.

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

Riavviare l'ambiente

Riavviare l'ambiente riattivando i gruppi di volumi, rimontando i file system, riavviando i servizi RAC e così via. Per precauzione, NetApp consiglia di riavviare il server al termine del processo di conversione, per assicurarsi che tutti i file di configurazione siano corretti e che tutti i dispositivi obsoleti vengano rimossi.

Attenzione: Prima di riavviare un host, assicurarsi che tutte le voci in `/etc/fstab` Il riferimento alle risorse SAN migrate verrà commentato. Se questa operazione non viene eseguita e si verificano problemi con l'accesso LUN, il risultato può essere un sistema operativo che non si avvia. Questo problema non danneggia i dati. Tuttavia, può essere molto scomodo avviare in modalità rescue o una modalità simile e corretta `/etc/fstab` In modo che il sistema operativo possa essere avviato per consentire l'avvio delle operazioni di risoluzione dei problemi.

Script di esempio della procedura di migrazione Oracle

Gli script presentati sono forniti come esempi di come eseguire lo script di varie attività del sistema operativo e del database. Vengono forniti così come sono. Se è necessario supporto per una procedura particolare, contattare NetApp o un rivenditore NetApp.

Arresto del database

Lo script Perl seguente prende un singolo argomento del SID Oracle e chiude un database. Può essere eseguito come utente Oracle o come root.

```

#!/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;}

```

Avvio del database

Lo script Perl seguente prende un singolo argomento del SID Oracle e chiude un database. Può essere eseguito come utente Oracle o come root.

```

#!/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;}

```

Convertire il file system in sola lettura

Lo script seguente prende un argomento del file system e tenta di smontarlo e rimontarlo in modalità di sola lettura. Questa operazione è utile durante i processi di migrazione in cui un file system deve essere mantenuto disponibile per replicare i dati e deve essere protetto contro danni accidentali.

```

#!/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;

```

Sostituire il file system

L'esempio di script riportato di seguito viene utilizzato per sostituire un file system con un altro. Poiché modifica il file `/etc/fstab`, deve essere eseguito come root. Accetta un singolo argomento delimitato da virgole per i file system vecchi e nuovi.

1. Per sostituire il file system, eseguire lo script seguente:

```

#!/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;

```

Come esempio di utilizzo di questo script, si supponga che i dati in /oradata viene migrato in /neworadata e. /logs viene migrato in /newlogs. Uno dei metodi più semplici per eseguire questa attività consiste nell'utilizzare una semplice operazione di copia dei file per riportare la nuova periferica al punto di montaggio originale.

2. Si supponga che i file system vecchi e nuovi siano presenti in `/etc/fstab` archiviare come segue:

```
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. Quando viene eseguito, questo script smonta il file system corrente e lo sostituisce con il nuovo:

```
[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. Lo script aggiorna anche `/etc/fstab` file di conseguenza. Nell'esempio illustrato, sono incluse le seguenti modifiche:

```
#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
```

Migrazione automatizzata del database

In questo esempio viene illustrato l'utilizzo di script di arresto, avvio e sostituzione del file system per automatizzare completamente la migrazione.

```
#!/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;

```

Visualizzare le posizioni dei file

Questo script raccoglie una serie di parametri critici del database e li stampa in un formato di facile lettura. Questo script può essere utile quando si esaminano i layout dei dati. Inoltre, lo script può essere modificato per altri usi.

```
#!/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";

```

Pulitura della migrazione ASM

```

#!/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";

```

Conversione del nome da ASM a file system

```
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;
/
```

Riprodurre i log sul database

Questo script accetta un singolo argomento di un SID Oracle per un database in modalità mount e tenta di riprodurre tutti i log di archivio attualmente disponibili.

```
#!/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;
```

Riprodurre i registri sul database di standby

Questo script è identico allo script precedente, tranne che è progettato per un database di standby.

```

#!/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;

```


Informazioni sul copyright

Copyright © 2024 NetApp, Inc. Tutti i diritti riservati. Stampato negli Stati Uniti d'America. Nessuna porzione di questo documento soggetta a copyright può essere riprodotta in qualsiasi formato o mezzo (grafico, elettronico o meccanico, inclusi fotocopie, registrazione, nastri o storage in un sistema elettronico) senza previo consenso scritto da parte del detentore del copyright.

Il software derivato dal materiale sottoposto a copyright di NetApp è soggetto alla seguente licenza e dichiarazione di non responsabilità:

IL PRESENTE SOFTWARE VIENE FORNITO DA NETAPP "COSÌ COM'È" E SENZA QUALSIVOGLIA TIPO DI GARANZIA IMPLICITA O ESPRESSA FRA CUI, A TITOLO ESEMPLIFICATIVO E NON ESAUSTIVO, GARANZIE IMPLICITE DI COMMERCIALIZZABILITÀ E IDONEITÀ PER UNO SCOPO SPECIFICO, CHE VENGONO DECLINATE DAL PRESENTE DOCUMENTO. NETAPP NON VERRÀ CONSIDERATA RESPONSABILE IN ALCUN CASO PER QUALSIVOGLIA DANNO DIRETTO, INDIRETTO, ACCIDENTALE, SPECIALE, ESEMPLARE E CONSEGUENZIALE (COMPRESI, A TITOLO ESEMPLIFICATIVO E NON ESAUSTIVO, PROCUREMENT O SOSTITUZIONE DI MERCI O SERVIZI, IMPOSSIBILITÀ DI UTILIZZO O PERDITA DI DATI O PROFITTI OPPURE INTERRUZIONE DELL'ATTIVITÀ AZIENDALE) CAUSATO IN QUALSIVOGLIA MODO O IN RELAZIONE A QUALUNQUE TEORIA DI RESPONSABILITÀ, SIA ESSA CONTRATTUALE, RIGOROSA O DOVUTA A INSOLVENZA (COMPRESA LA NEGLIGENZA O ALTRO) INSORTA IN QUALSIASI MODO ATTRAVERSO L'UTILIZZO DEL PRESENTE SOFTWARE ANCHE IN PRESENZA DI UN PREAVVISO CIRCA L'EVENTUALITÀ DI QUESTO TIPO DI DANNI.

NetApp si riserva il diritto di modificare in qualsiasi momento qualunque prodotto descritto nel presente documento senza fornire alcun preavviso. NetApp non si assume alcuna responsabilità circa l'utilizzo dei prodotti o materiali descritti nel presente documento, con l'eccezione di quanto concordato espressamente e per iscritto da NetApp. L'utilizzo o l'acquisto del presente prodotto non comporta il rilascio di una licenza nell'ambito di un qualche diritto di brevetto, marchio commerciale o altro diritto di proprietà intellettuale di NetApp.

Il prodotto descritto in questa guida può essere protetto da uno o più brevetti degli Stati Uniti, esteri o in attesa di approvazione.

LEGENDA PER I DIRITTI SOTTOPOSTI A LIMITAZIONE: l'utilizzo, la duplicazione o la divulgazione da parte degli enti governativi sono soggetti alle limitazioni indicate nel sottoparagrafo (b)(3) della clausola Rights in Technical Data and Computer Software del DFARS 252.227-7013 (FEB 2014) e FAR 52.227-19 (DIC 2007).

I dati contenuti nel presente documento riguardano un articolo commerciale (secondo la definizione data in FAR 2.101) e sono di proprietà di NetApp, Inc. Tutti i dati tecnici e il software NetApp forniti secondo i termini del presente Contratto sono articoli aventi natura commerciale, sviluppati con finanziamenti esclusivamente privati. Il governo statunitense ha una licenza irrevocabile limitata, non esclusiva, non trasferibile, non cedibile, mondiale, per l'utilizzo dei Dati esclusivamente in connessione con e a supporto di un contratto governativo statunitense in base al quale i Dati sono distribuiti. Con la sola esclusione di quanto indicato nel presente documento, i Dati non possono essere utilizzati, divulgati, riprodotti, modificati, visualizzati o mostrati senza la previa approvazione scritta di NetApp, Inc. I diritti di licenza del governo degli Stati Uniti per il Dipartimento della Difesa sono limitati ai diritti identificati nella clausola DFARS 252.227-7015(b) (FEB 2014).

Informazioni sul marchio commerciale

NETAPP, il logo NETAPP e i marchi elencati alla pagina <http://www.netapp.com/TM> sono marchi di NetApp, Inc. Gli altri nomi di aziende e prodotti potrebbero essere marchi dei rispettivi proprietari.