



## WS2012R2 Pianificazione e configurazione del CSV

### Introduzione

Il Cluster Shared Volumes (CSVs) è stato introdotto in Windows Server 2008 R2 per permettere di superare il limite di dover dedicare una LUN/Volume ad ogni VM che si intende avere in alta disponibilità utilizzando un Failover cluster di Hyper-V in quanto consente di condividere una LUN a più VM.

Il CSV basa il suo funzionamento sull'uso di un **“Coordinator Node”** che si occupa di gestire le scritture dei metadati NTFS sui Volumi CSV, nel caso si verifichi un failure che interessa il Coordinator Node le VM verranno gestite da altri nodi che insistono sui medesimi volumi senza che vi siano disservizi.

Inoltre il CSV è in grado anche di garantire la **“SAN Fault Tolerance”** in quanto è possibile accedere ad un Volume CSV da un nodo che a causa di un failure ha perso la connessione diretta con la SAN. In questo caso, infatti, i **“Redirect I/O”** passano tramite la rete tra i nodi dedicata al CSV e vengono gestiti dal **“Coordinator Node”** che effettua le operazioni per conto del Nodo richiedente. Ovviamente la comunicazione non avrà le prestazioni di un accesso diretto alla SAN via doppia HBA, ma consente di avere un punto di accesso in situazioni d'emergenza permettendo di risolvere il failure verso la SAN senza dare disservizi.

In questo articolo analizzeremo quali sono le procedure consigliate per la configurazione del CSV con particolare attenzione per gli ambienti virtuali anche se a partire da Windows Server 2012 il CSV è supportato anche per le Scale-out file shares e non solo col ruolo Hyper-V.

### Argomenti

Numero di volumi CSV e di LUN .....	2
Configurazione del volumi CSV .....	3
Conclusioni .....	4
Riferimenti .....	4

## Numero di volumi CSV e di LUN

Uno dei primi punti su cui è necessari prendere una decisione quando si implementa un Hyper-V Failover Cluster è quello di pianificare quanti volumi CSV implementare.

Nel documento [Use Cluster Shared Volumes in a Failover Cluster](#) viene data l'indicazione di gestire la pianificazione del numero di volume come se si dovesse gestire l'implementazione delle VM in ambiente fisico. Questo significa che dal momento che in un'implementazione fisica è consigliabile mantenere i file di sistema e il page file su un disco separato rispetto a quello dei dati, la stessa cosa in ambiente virtuale si traduce dedicando **un CSV per i VHD contenenti i file di sistema e page file e un CSV dedicato ai VHD dei dati.**

**Per quanto riguarda il numero di LUN viene suggerito di fare riferimento alle indicazioni del produttore della SAN** che potrebbe suggerire di configurare ogni LUN con una partizione su cui posizionare un volume CSV.

Dal punto di vista del CSV in ogni caso **non vi sono limitazioni sul numero di VM supportate sul singolo volume CSV**, ma **per garantire performance adeguate occorre pianificare il cluster in base ai workload svolti dalle VM e quindi in base alle operazioni di I/O per secondo eseguite sullo storage.** Ne consegue che se si sta implementando un'infrastruttura VDI (Virtual Desktop Infrastructure) dal momento che le VM eseguono workload leggeri sullo storage è possibile assegnare più VM allo stesso volume CSV, mentre se si devono gestire VM che elaborazioni su dati mediante database è consigliabile assegnare poche VM allo stesso volume CSV. Ovviamente queste valutazioni vanno fatte tenendo conto delle performance e del numero di dischi della SAN, nel caso non vi siano problemi di performance è sempre consigliabile scegliere l'implementazione più semplice ovvero quella che prevede un minor numero di volumi CSV soprattutto se ad ogni volume CSV corrisponde una LUN in quanto si ha anche un miglior utilizzo dello spazio di archiviazione.

Inoltre come fa giustamente notare Aidan Finn (MVP per Hyper-V) nel suo post [Another Hyper-V Implementation Mistake –Too Many CSVs](#) prestazionalmente parlando è preferibile avere LUN con molti dischi per potere sfruttare più meccanicamente contemporaneamente che più LUN con meno dischi. A questa considerazione si deve aggiungere che più il numero di volumi CSV cresce e maggiore è il numero di iSCSI3 persistent reservations (PR) necessarie che può essere calcolato come segue:  $PR = \text{Numero Hosts} \times \text{Numero di connessioni verso lo Storage} \times \text{Numero di volumi CSV per Host}$  (per esempio nel caso di 2 host con due connessioni verso lo storage ciascuno e 4 volumi CSV le iSCSI3 persistent reservations saranno  $2 \times 2 \times 4 = 16$ ). Si tenga conto che ogni SAN limite di iSCSI3 persistent reservations supportato.

**Concludendo in assenza di indicazioni specifiche del costruttore della SAN si può seguire la best practice suggerita da Microsoft di creare due LUN dedicate a due volumi CSV uno per i VHD dei file di sistema e pagefile e uno per i VHD dei dati.**

Un altro approccio può essere quello indicato da nella white paper [HP Departmental Private Cloud Reference Architecture](#) redatta in modo congiunto da Microsoft e HP in cui viene discussa la realizzazione di un'infrastruttura di esempio con hardware HP e OS WS2012 e WS2008R2 (per maggiori informazioni si veda il post [HP Departmental Private Cloud Reference Architecture](#)). In tale white paper viene proposto un approccio basato sulla **creazione di una LUN per ogni nodo del cluster ed creare in ciascuna LUN un volume CSV** in cui verranno memorizzate le VM preferibilmente assegnate a tale nodo.

Con questo tipo di approccio ogni nodo del cluster risulta coordinator (ovvero l'owner) del proprio volume CSV evitando così redirezioni sulle scritture dei metadati NTFS, inoltre suddividendo le VM sui nodi si ha anche implicitamente una suddivisione dello storage oltre ad un bilanciamento sui nodi in termini di cpu, rete e memoria.

Si tenga conto che vi sono situazioni in cui per determinati servizi e applicazioni vengono indicate specifiche configurazioni per lo storage come nel caso di SAP in viene ad esempio suggerito di distribuire i database su 4 CSV

assegnati ad altrettante LUN. A riguardo si veda il post [Microsoft Private Cloud Solution for SAP: Configuration of Cluster Shared Volumes, VHDX, Disks & LUNs](#)

### Configurazione del volumi CSV

Per quanto riguarda la preparazione delle partizioni che verranno dedicate ai volumi CSV in mancanza di indicazioni specifiche dello storage vendor è possibile attenersi a quanto suggerito nel post [Windows Server 2012 Hyper-V Best Practices \(In Easy Checklist Form\)](#)

- **Formattare i dischi con NTFS**, in Windows Server 2012 R2 è possibile utilizzare anche **ReFS** (Resilient File System) che però non è supportato per uno Scale-Out File Server
- **Formattare i dischi usando GPT dal momento che i volumi CSV potrebbero superare i 4TB** (per alcune informazioni circa l'uso del GPT col [Failover Cluster si veda GPT and Failover Clustering](#))
- **Formattare il disco usando una Allocation Unit Size di 64K se il volume CSV conterrà file VHD/VHDX** (a meno di indicazioni diverse da parte dello storage vendor)
- **Mantenere dello spazio libero nei volumi CSV dedicati a file VHD/VHDX:**
  - 15% di spazio libero per partizioni con dimensione inferiori a 1TB
  - 10% di spazio libero per partizioni con dimensioni comprese tra 1TB e 5TB
  - 5% di spazio libero per partizioni con dimensioni superiori a 5TB
- **Abilitare la CSV Cache sui volumi CSV su cui vengono eseguite molti read-only unbuffered I/O**, generati ad esempio da VM che eseguono frequenti operazioni di lettura e ridotte operazioni di scrittura (un tipico scenario solo i pool di VM dedicate al VDI). A riguardo si veda [How to Enable CSV Cache](#).
- **Escludere la directory %SystemDrive%\ClusterStorage dall'analisi in tempo reale dell'antivirus.**

Il formato **ReFS** nasce per essere più robusto in quanto consente l'auto-detect data corruption e automaticamente eseguire le correzioni necessarie senza necessità di porre offline il volume eliminando quindi la necessità di eseguire CHKDSK. Altra caratteristica del ReFS è la sua scalabilità in quanto un volume può arrivare fino a  $2^{78}$  bytes usando 16 KB cluster sizes (a riguardo si veda [Resilient File System Overview](#)). Al momento però ReFS non supporta tutte le feature supportate da NTFS (un volume CSV formattato con ReFS non supporta TRIM/unmap o ODX.), quindi **può essere indicato solo nel caso sia necessario gestire volumi CSV di grandi dimensioni**. Per ulteriori approfondimenti si vedano [Windows Server 2012: Does ReFS replace NTFS? When should I use it?](#), [Disable the Integrity Bit of VHDs Copied to an ReFS Volume Using PowerShell](#) e la white paper di DELL [Windows Server 2012 R2 Best Practices for Dell Compellent Storage Center](#)

**Il dimensionamento dell'Allocation Unit Size in realtà richiede un'analisi più dettagliata** in quanto la sua impostazione è importata ai fini del Disk Partition Alignment. Per impostazione predefinita l'Allocation Unit Size è di 4K, come indicato in [Default cluster size for NTFS, FAT, and exFAT](#), ma alcuni prodotti richiedono valori diversi per avere performance migliori (ad esempio i DB SQL Server e i file .edb e .log di Exchange 2013 richiedono 64K come best practice). In ogni caso va ricordato che **tutti gli strati in cui vengono eseguite le operazioni di I/O devono essere configurati per ottenere il Disk Partition Alignment** quindi occorre anche aver impostato in maniera adeguata lo **Strip Size** della LUN, il **Partition Offset** e l'**Allocation Unit Size** nelle partizioni contenute all'interno dei VHD/VHDX. Per una linea guida su questo argomento si veda il mio post [SQL Server e il Disk Partition Alignment](#). In conclusione **nel caso in cui si sia scelto di avere un volume CSV dedicato a dati, su tale volume può aver senso impostare le configurazioni per il Disk Partition Alignment considerando un'Allocation Unit Size di 64K.**

Un altro approccio è suggerito nel post [Microsoft Private Cloud Solution for SAP: Configuration of Cluster Shared Volumes, VHDX, Disks & LUNs](#) per la virtualizzazione di SAP, ovvero quello di **formattare i volumi di dischi VHD/VHDX che contengono file di database con allocation uniti di 64K, e di formattare i CSV con l'allocation unit di default di 4K:**

*All Virtual Hard disks (the VHDX file system) that hold SQL Server datafiles or log files must be formatted with 64k NTFS Allocation Unit size.*

*The NTFS allocation unit size on the Cluster Shared Volumes themselves should be formatted with the default NTFS allocation unit size (4k).*

Per quanto riguarda invece la **percentuale di spazio libera su un volume CSV** è possibile controllare tale valore tramite il comando PowerShell **Get-ClusterSharedVolume "Cluster Disk x" | fc \*** leggendo il valore della proprietà **PercentFree**.

### Conclusioni

L'introduzione del CSV è stato sicuramente un passo importante verso una semplificazione ed una maggiore scalabilità e robustezza del Failover Cluster. In scenari di piccole dimensioni il CSV permette di fatto di non occuparsi in modo minuzioso della configurazione dello storage. In ogni caso per massimizzare l'investimento dell'hardware conviene pianificare in maniera corretta il numero di LUN, il numero di volumi CSV e il Disk Partion Alignment in base allo scenario da gestire.

### Riferimenti

- [Informazioni sull'utilizzo dei Cluster Shared Volumes - Parte 1](#)
- [Informazioni sull'utilizzo dei Cluster Shared Volumes - Parte 2](#)
- [What's New in Failover Clustering in Windows Server](#)
- [Hyper-V: Clustering & Resiliency. Step 6: Clustered Shared Volume & Optimize](#)
- [Performance Tuning Guidelines for Windows Server 2012 R2](#)
- [Performance Tuning Guidelines for previous versions of Windows Server](#)