

# SQL03 – Come ottenere il massimo da SQL Server virtuale

**Danilo Dominici**

**Data Platform Specialist**

**[ddominici@solidq.com](mailto:ddominici@solidq.com)**

**Twitter: [@danilo\\_dominici](https://twitter.com/danilo_dominici)**

**#CDays13 – 24, 25 e 26 settembre 2013, Roma**

# Grazie a

Sponsor



COMMUNITY DAYS 

# Chi è Danilo Dominici

- Data Platform Specialist @ SolidQ
- SQL Server DBA & VMWare Administrator @ Regione Marche
- Lavoro con SQL Server dalla versione 6.5
- Consulente, docente, speaker, autore
  - SQL Server 2012 Upgrade Guide



# Sondaggio

- ☐ SQL Server fisico
- ☐ SQL Server virtuale
  - ☐ VMWare
  - ☐ Hyper-V
  - ☐ KVM
  - ☐ XEN
  - ☐ VirtualBox
  - ☐ Oracle VM

# Agenda

- Perché virtualizzare SQL Server?
- Scenari
- Best practices

# Perché virtualizzare SQL Server

- Maggiore flessibilità ed efficienza
  - Nessuna dipendenza dall'hardware
  - Spostamento trasparente delle VM tra i nodi
- Maggiore uptime via vMotion/Live Migration
- Disaster Recovery semplificato
- Semplicità di deployment
  - Creazione delle VM da template
  - Clonazione delle VM (per test, sviluppo, supporto tecnico, etc)

# Perché virtualizzare SQL Server

- Consolidamento dei server
  - Minor numero di server = minor costo complessivo (alimentazione, condizionamento, manutenzione)
- Necessità di mantenere in esercizio versioni di SQL Server fuori supporto
  - SQL Server 2000 è fuori supporto dal 9 aprile 2013  
<http://bit.ly/bph7e4>
- Rapido adattamento alle esigenze
  - Più vRAM, più vCPU, anche *hot-add*
- Sostituzione rapida del vecchio host via P2V

# Quando virtualizzare?

- Aggiornamento dell'hardware o delle applicazioni
- Upgrade della versione di SQL Server
  - Versione obsoleta
  - Necessità di utilizzo delle funzionalità Enterprise



# Miti e leggende

- “SQL Server, se virtuale, è molto più lento”
- “Microsoft non supporta SQL Server su VMWare/Hyper-V”
- “Database molto grandi non possono essere virtualizzati”

In realtà...

# Performance

- Esistono casi limite in cui le prestazioni di un SQL Server virtuale sono più lente di una macchina fisica...
- Ma con le ultime generazioni di hypervisor le prestazioni sono praticamente equivalenti
  - Qualche punto percentuale di overhead sulla CPU
  - microsecondi di latenza per l'I/O

# Supporto

- VMWare
  - <http://support.microsoft.com/kb/897615>
  - Server Virtualization Validation Program (SVVP)
  - VMware Customer Support Statement and TSANet
    - [http://vmware.com/support/policies/ms\\_support\\_statement.html](http://vmware.com/support/policies/ms_support_statement.html)
- Hyper-V
  - <http://support.microsoft.com/kb/956893>

# Quando NON virtualizzare

- Se il sistema è già saturo, i benefici della virtualizzazione sono limitati a:
  - Manutenibilità del server virtuale
  - HA/DR del sistema di virtualizzazione (Vmotion o Live migration, DRS)
- Ma le prestazioni in questo caso non migliorano virtualizzando ☹️

# SCENARI

# Scenari possibili

- Il mio SQL Server è ancora fisico
  - Come scegliere l'hypervisor più adatto?
  - Di che cosa tener conto nella progettazione dell'ambiente di virtualizzazione?
  - Come e quando virtualizzare?
- Ho già un sistema di virtualizzazione e SQL Server virtuale
  - Come posso ottimizzarlo?

# Quale hypervisor per SQL Server?

## VMWARE vSphere 5.5

- 64 vCPU per VM (ESXi=8)
- 1TB RAM per VM (ESXi=32GB)
- Cluster → 32 nodi (ESXi=N/A)
- ESXi è gratuito
- vCenter Std + vSphere 5 Ent  
circa € 7000
- Vanno aggiunte le licenze del S.O.

## HYPER-V 3.0

- 64 vCPU per VM
- 1TB RAM per VM
- Cluster → 64 nodi
- Hyper-V Server è gratuito
- SCCM 2012 + Windows  
Server 2012 Datacenter circa €  
7000

Il 95% dei casi richiede 1-2 vCPU, <= 4GB di picco, <= 2.5 Mb/s, < 10.000 IOPS

# Se SQL Server è ancora fisico

- Analizzare il workload corrente
  - Per conoscere come va il sistema PRIMA
  - Per capire come sta andando il sistema DOPO
- Progettare il virtuale
- Scegliere il percorso di virtualizzazione più adatto
  - Nuova installazione
  - Physical-to-Virtual (P2V)



# Partire con il piede giusto

- **LEGGERE** attentamente la documentazione fornita dai produttori di tutto ciò che è coinvolto
  - hypervisor, storage, apparati di rete, etc
- Documentarsi nei principali siti/blog
  - Vedi la sezione “Risorse” in fondo alla presentazione
    - Links utili per saperne di più

# Pianificare in ogni dettaglio

- Cosa prevedere nella pianificazione
  - ✧ Requisiti “business”: SLA, RPO, RTO
  - ✧ Definire una baseline dell’attuale carico di lavoro (fisico o virtuale)
  - ✧ Stimare la crescita del carico di lavoro in un certo periodo di tempo
  - ✧ Identificare i requisiti del sistema di I/O
  - ✧ Caratteristiche dello storage
  - ✧ Versioni del software
  - ✧ Licenze (possono determinare l’architettura generale)
  - ✧ Tipo di carico di lavoro (OLTP, Batch, Data warehouse)
  - ✧ Account necessari per l’installazione e di servizio
  - ✧ Strategie High Availability
  - ✧ Strategie per il backup e recovery

# Analizzare i diversi tipi di workload

## OLTP

- Numero elevato di piccole query
- Utilizzo elevato della CPU
- Sensibile ai picchi di utilizzo (riducono gli SLA)

## BATCH /ETL

- Scritture intensive
- Tipico degli orari notturni, bassa o nessuna attività di CPU durante il resto del giorno
- Attività sostenuta, ma basso impatto sugli SLA

## DSS

- Numero ridotto di grandi query
- Uso intensivo di CPU, memoria e dischi
- Picchi durante i periodi di consolidamento dei dati
- Beneficia del parallelismo

# Definire una baseline

- Quali caratteristiche prestazionali deve avere il mio vSQL (CPU, memoria, IO, rete) ?
  - Media giornaliera di utilizzo
  - Picchi giornalieri di utilizzo
  - Picchi orari
  - Picchi periodici (mensili, semestrali o annuali)
    - Legati ad esempio al ciclo di business

# Definire una baseline (continua)

- Strumenti per il monitoraggio
  - Performance monitor di Windows
    - Processor(\*) %Processor Time
    - Process(sqlservr) %Processor Time
    - SQLServer:Memory Manager Total Server Memory (KB)
    - PhysicalDisk(\*) Disk Reads/Sec, Disk Writes/Sec
    - PhysicalDisk(\*) Disk Reads Bytes/Sec, Disk Write Bytes/Sec
    - Network Interface(\*) Bytes Received/Sec, Bytes Sent/Sec
  - VMWare
    - Capacity Planner
    - ESXTOP
  - Hyper-V Performance Counter
    - Estratte da PAL (<http://pal.codeplex.com>)

# Definire una baseline

vSphere: ESXTOP

| Risorsa | Parametro  | Descrizione   | Valore ottimale |
|---------|------------|---|-----------------|
| CPU     | %USED      | % CPU nell'intervallo di campionamento  |                 |
|         | %RDY       | Tempo di CPU in stato di attesa   | < 10            |
| Memoria | MCTLSZ     | Dimensione in MB della memoria richiesta al resource pool utilizzando il ballooning | 0               |
|         | Swapin/out | Dimensione della memoria che ESX scambia da/per il disco                            |                 |
| Dischi  | CMDS/s     | Numero di IOPS  |                 |
|         | DAVG/cmd   | Latenza media (ms) del disco (LUN)  | <= 10ms         |
|         | KAVG/cmd   | Latenza media (ms) nel VMKernel   | <= 1ms          |

<http://kb.vmware.com/kb/1006797>

# Definire una baseline

Monitorare le prestazioni dell'I/O

- Quali indicatori utilizzare?

| Contatore                         | Descrizione           |
|-----------------------------------|-----------------------|
| LogicalDisk\Avg Disk sec/Read     | Latenza di lettura    |
| LogicalDisk\Avg Disk sec/Write    | Latenza di scrittura  |
| LogicalDisk\Disk Read Bytes /sec  | Capacità in lettura   |
| LogicalDisk\Disk Write Bytes /sec | Capacità di scrittura |
| LogicalDisk\Disk Reads/sec        | IOPS in lettura       |
| LogicalDisk\Disk Writes/sec       | IOPS in scrittura     |
| LogicalDisk\Disk Transfers/sec    | Totale IOPS           |

# Definire una baseline

Monitorare le prestazioni dell'I/O

- Fare i test sui pattern di utilizzo del disco più comuni per SQL Server

| Tipo di I/O              | Dimensione blocco | Threads/coda    | Che cosa simula |
|--------------------------|-------------------|-----------------|-----------------|
| Lettura/scrittura random | 8K                | # cores / files | File dati OLTP  |
| Scrittura sequenziale    | 60K               | 1 / 32          | Transaction log |
| Lettura sequenziale      | 512K              | 1 / 16          | Table scan      |
| Scrittura sequenziale    | 256K              | 1 / 16          | Bulk load       |
| Lettura random           | 32K               | # cores / 1     | SSAS workload   |
| Lettura sequenziale      | 1MB               | 1 / 32          | Backup          |
| Scrittura random         | 64K-256K          | # cores / files | Checkpoint      |

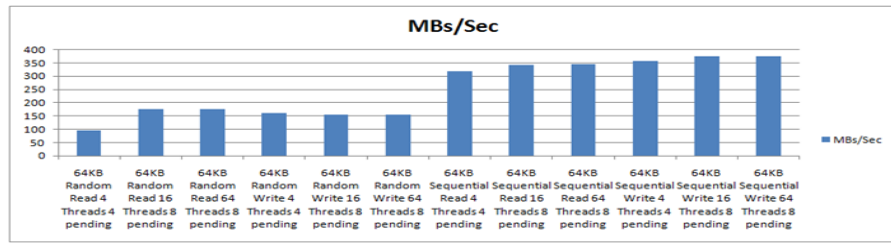


# Definire una baseline

Monitorare le prestazioni dell'I/O

- SQLIO  
(vedi sez. Risorse)
- Esportare su Excel e visualizzare in forma grafica
  - Vari tools *free* disponibili
  - Es. script Powershell di Jonathan Kehayias

```
sqlio vl.5.SG
using system counter for latency timings, 14318180 counts per second
16 threads writing for 60 secs to file d:\MSSQL\testfile1.dat
  using 8KB random I/Os
  enabling multiple I/Os per thread with 8 outstanding
  buffering set to use hardware disk cache (but not file cache)
using current size: 2048 MB for file: d:\MSSQL\testfile1.dat
initialization done
CUMULATIVE DATA:
throughput metrics:
IOs/sec: 18527.91
MBs/sec: 144.74
latency metrics:
Min_Latency(ms): 0
Avg_Latency(ms): 6
Max_Latency(ms): 4900
histogram:
ms: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24+
%: 78 6 8 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2
```



# SQLIO

- E' un tool per determinare la capacità dell'I/O, tipicamente della SAN
- Può generare diversi tipi di I/O
  - Lettura/Scrittura
  - Random/Sequenziale
  - Thread multipli
  - Diversa dimensione dei blocchi letti/scritti
- Modificare il file param.txt e impostare la dimensione del file di test almeno a 2/4 volte la dimensione della cache della SAN

# DEMO

COMMUNITY DAYS 

# **BEST PRACTICES: STORAGE**

# Block alignment

- VMWare: è automatico se lo storage viene configurato da vCenter
- A partire da Windows 2008 l'allineamento è automatico

– <http://bit.ly/BYUg4>

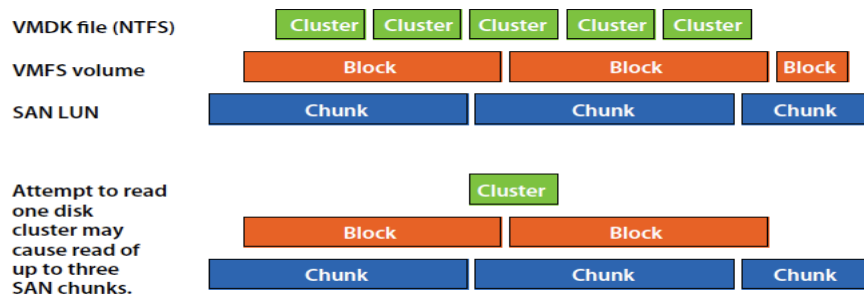


Figure 1: Unaligned partitions result in additional I/O

# Thin/thick provisioning & Block zeroing

## VMWARE

- Il thin provisioning permette di risparmiare spazio nello storage allocando solo lo spazio occupato
  - ...ma le performance degradano progressivamente
- Utilizzare dischi «Thick Eager Zeroed» per il massimo delle performance
- Impostare in questo modo i dischi per:
  - Database
  - Log
  - Tempdb

# Thin/thick provisioning & Block zeroing

## VMWARE

- Configurazione
  - Se la VM è accesa
    - Utilizzando lo Storage Vmotion
    - Formattando il disco SENZA la formattazione veloce
  - Se la VM è spenta
    - Inflate del disco (da vCenter)
    - Utilizzando il comando **vmkfstools**
      - [kb.vmware.com](http://kb.vmware.com) (articolo 1011170)
      - `vmkfstools -D «VM.vmdk»` (se `tbz=0` → `eagerzeroedthick`)
      - `vmkfstools -k «VM.vmdk»` (converte a `eagerzeroedthick`)
      - Occhio alle performance: alto carico sulla san (senza vAAI)

# Paravirtual SCSI (PVSCSI)

VMWARE

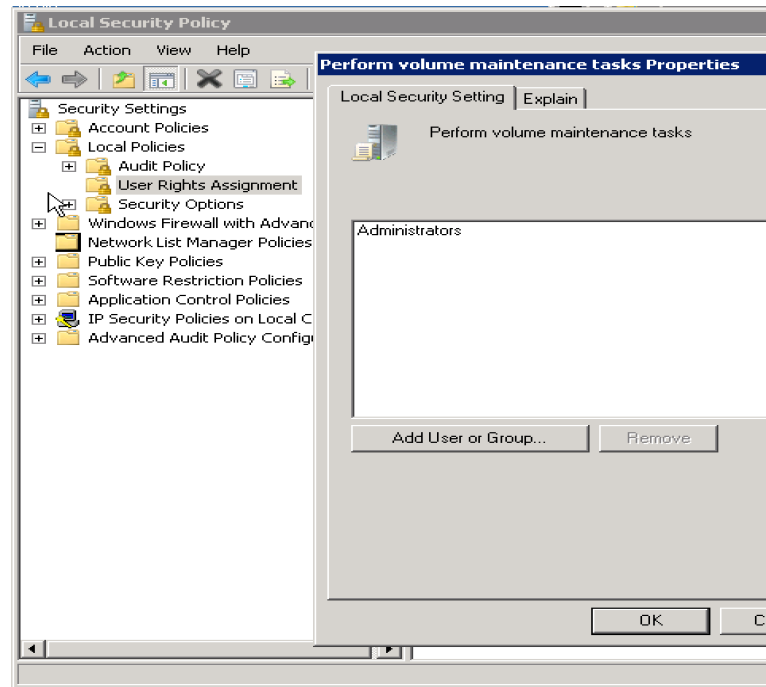
- Gli adapter PVSCSI sono pensati per elevati volumi di I/O
  - Maggiore throughput (fino al 10-15%)
  - Minore carico sulla CPU (fino al 30% in meno)
- Sono più adatti ad ambienti SAN
- NON sono adatti per DAS (Direct-Attached Storage)



# Instant File Initialization

## SQL Server

- Aggiungere al service account di SQL Server il permesso **Perform Volume Maintenance Task** in Local Security Policy
  - Per default solo gli Administrators
- SQL Server riempie con zeri i files durante la creazione/espansione e blocca le scritture finché non ha terminato
  - Richiede il restart quando aggiunto ed il reboot se rimosso

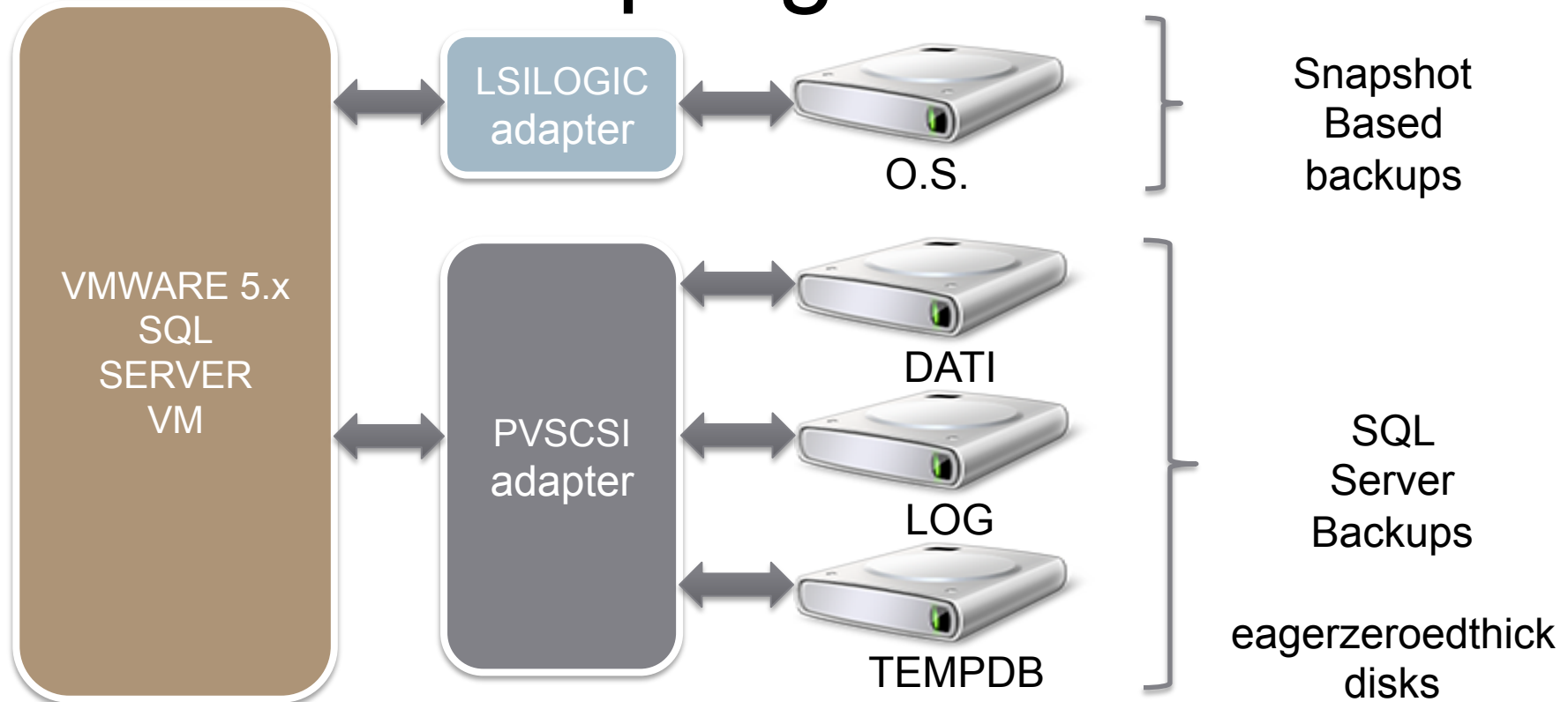


# Posizionamento TempDB

## SQL Server

- Isolare il tempdb dagli altri database
- Creare files multipli per il tempdb
  - Microsoft consiglia 1 file dati per CPU
    - <http://technet.microsoft.com/library/Cc966534>
  - Meglio ancora:
    - Fino a 8 core: un file dati per core
    - Più di 8 core: partire da 8 file dati ed incrementare se c'è contesa
      - <http://technet.microsoft.com/library/Cc966545>
  - TF 1118 per allocare full extent agli oggetti del tempdb

# Riepilogando



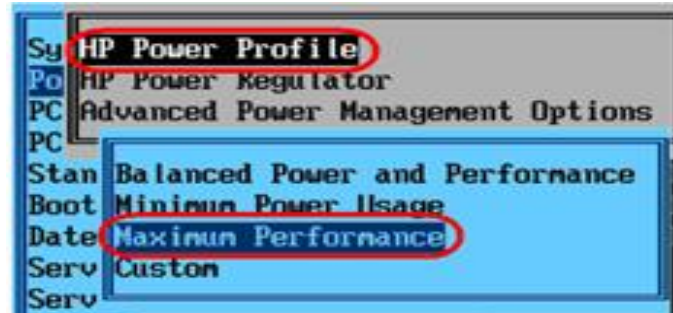
# DEMO

COMMUNITY DAYS 

# BEST PRACTICES: CPU

# Profilo risparmio energetico

- Per default, viene impostato a “bilanciato” o “risparmio energetico”
- In un host che ospita SQL Server va impostato così →



# Ottimizzare le vCPU

- In sistemi di produzione evitare l'overcommitment del processore
  - Mantenere un rapporto di 1:1 tra i cores fisici e quelli virtuali
- Aggiungere host se CPU fisica > 65%
- Aggiungere vCPU se CPU virtuale > 65%

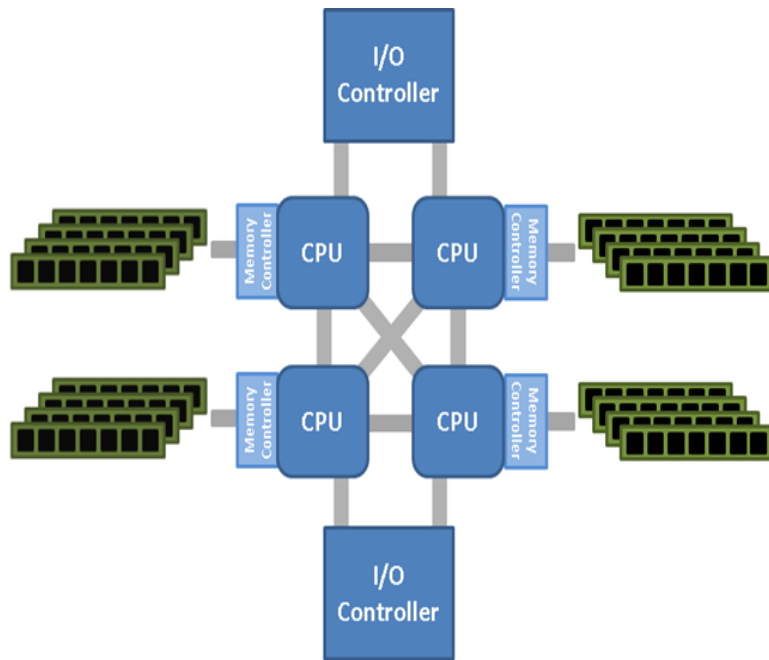
# Hyperthreading

- I processori con Hyper-threading attivo duplicano il numero di processori visti dal sistema
  - Quindi un sistema con 4 core + HT  $\rightarrow$  8 core
  - E quindi 8 core  $\rightarrow$  8 vCPU
- Ma 4 core con HT sono uguali a 8 core ?
  - No, generalmente circa il 120% del numero di core
  - Quindi nel nostro sistema con 4 core  $\rightarrow$  5 vCPU
- Su sistemi critici
  - Il rapporto **deve** essere 1:1 tra core fisici e vCPU



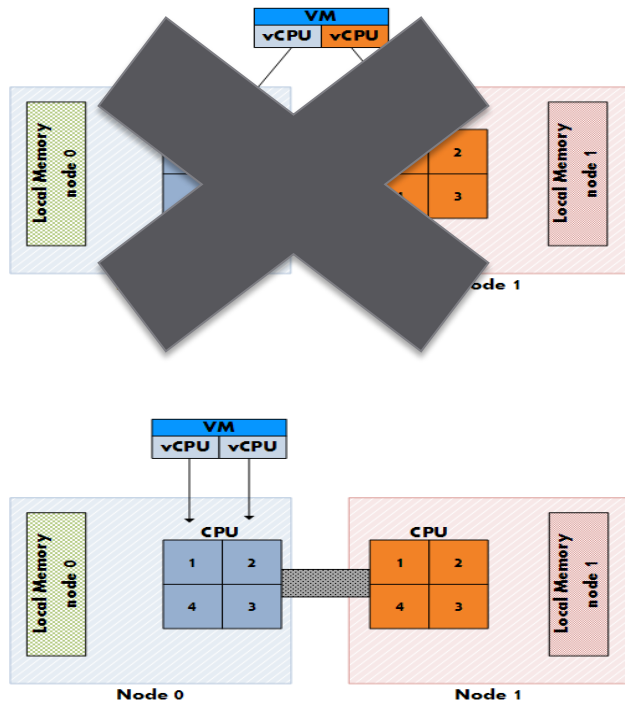
# NUMA

- Non-Uniform Memory Access
  - Ottimizza i problemi di performance quando più processori accedono alla stessa memoria fisica, separando la memoria per ciascun nodo NUMA
  - Ciascun processore ha la sua memoria “dedicata”
  - L’accesso alla memoria “remota” di un altro nodo è possibile, ma costosa



# NUMA

- Per prestazioni ottimali, ciascuna VM non dovrebbe estendersi su più nodi
- Esempio:
  - 2 processori = 2 nodi NUMA
  - 128 GB di RAM
  - Ogni nodo NUMA ha max 64GB di RAM
    - Quindi ciascuna VM non deve superare i **64GB**
- Tools per elenco nodi NUMA
  - [Coreinfo](#) di Sysinternals
  - CPU history nel Task Manager



# NUMA

- In vSphere 5 possiamo usare vNUMA
  - Abilitato su VM con vCPU > 8
  - Occorre modificare il parametro:  
*numa.vcpu.maxPerVirtualNode* = xxx
  - E' disabilitato se attivo *Hot Add*
  - Dipende dalla configurazione socket/core del server
    - [vSphere 5.1 Resource Management Guide](#) (pag.132)

# **BEST PRACTICES: MEMORIA**

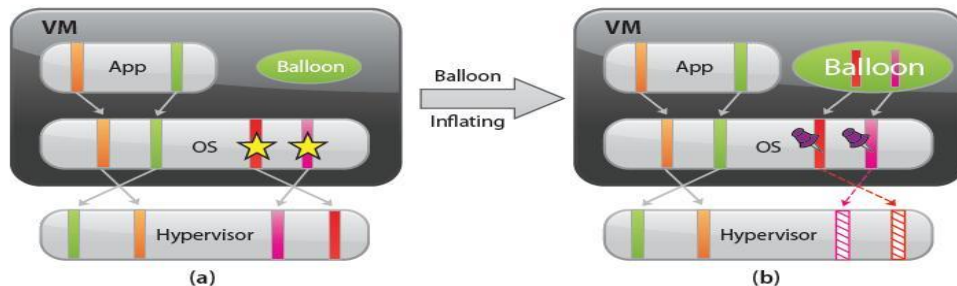
# Memoria

- La quantità di memoria assegnata alla VM dipende da:
  - Memoria totale dell'host
  - Memoria disponibile dell'host
  - Reservations
  - Limiti definiti dall'amministratore
  - DRS (Vmware) / Dynamic Memory (Hyper-V)

# Gestione dinamica della memoria

- VMWare Balloon Driver
  - Installato con i VMWare Tools
  - Serve a comunicare alla VM lo stato di pressione di memoria dell'host
  - In questo modo il sistema operativo guest si occupa di allocare o deallocare le pagine di memoria necessarie

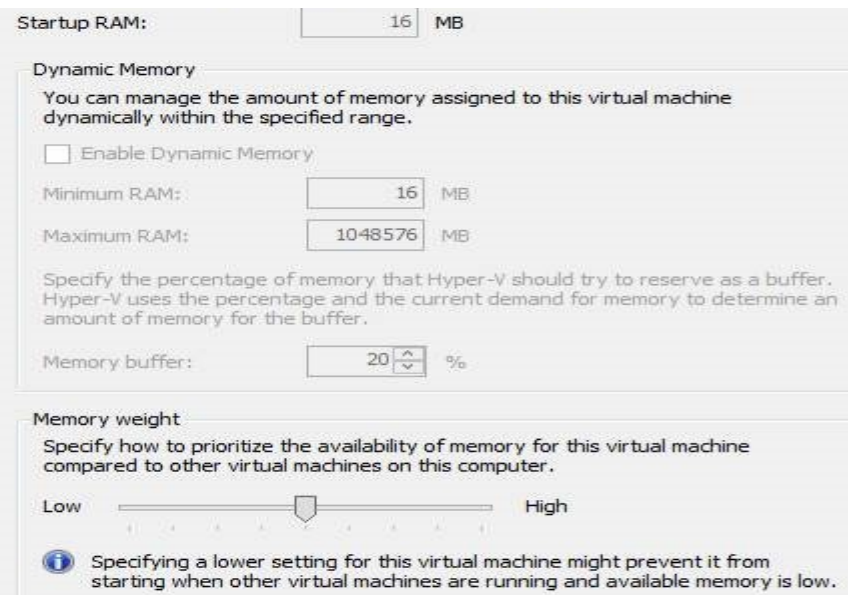
Figure 6: Inflating the balloon in a virtual machine ESX



# Gestione dinamica della memoria

- Hyper-V 3.0 Dynamic Memory

- Installato con gli Integration Services (quelli di Hyper-V 😊)
- Non fa overcommit
  - Tranne durante lo Smart Paging (swapping temporaneo sul disco quando l'host non ha memoria per coprire la *Startup RAM*)



The screenshot shows the 'Startup RAM' and 'Dynamic Memory' settings for a virtual machine. The 'Startup RAM' is set to 16 MB. Under 'Dynamic Memory', the 'Enable Dynamic Memory' checkbox is unchecked. The 'Minimum RAM' is set to 16 MB and the 'Maximum RAM' is set to 1048576 MB. The 'Memory buffer' is set to 20%. The 'Memory weight' section shows a slider between 'Low' and 'High' with a shield icon in the middle. A note at the bottom states: 'Specifying a lower setting for this virtual machine might prevent it from starting when other virtual machines are running and available memory is low.'

Startup RAM: 16 MB

Dynamic Memory

You can manage the amount of memory assigned to this virtual machine dynamically within the specified range.

☐ Enable Dynamic Memory

Minimum RAM: 16 MB

Maximum RAM: 1048576 MB

Specify the percentage of memory that Hyper-V should try to reserve as a buffer. Hyper-V uses the percentage and the current demand for memory to determine an amount of memory for the buffer.

Memory buffer: 20 %

Memory weight

Specify how to prioritize the availability of memory for this virtual machine compared to other virtual machines on this computer.

Low High

Specifying a lower setting for this virtual machine might prevent it from starting when other virtual machines are running and available memory is low.

# Altre considerazioni

- Evitare l'*overcommit* della memoria
  - Se proprio è necessario, monitorarne l'utilizzo
- Per SQL Server critici *riservare* la memoria per evitare ballooning o swapping
- Posizionare server “simili” sullo stesso host, per beneficiare del Transparent Page Sharing (TPS)
- Dimensionare la memoria della VM ad un valore pari o inferiore alla memoria max del nodo NUMA (evitiamo il *remote memory access*)



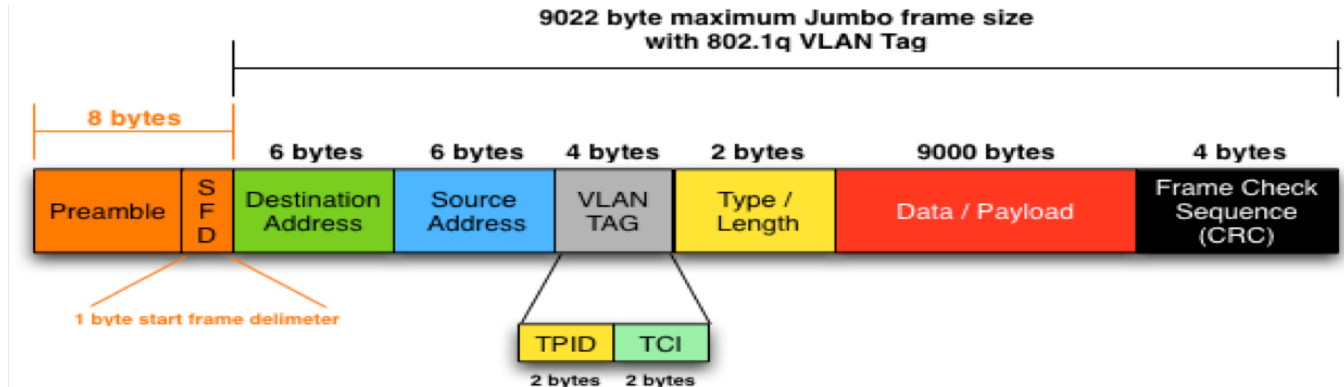
# Altre considerazioni

- Assegnare il permesso “Lock Pages in Memory” all’account di servizio di SQL Server
- Trace flags all’avvio per utilizzare l’allocazione di pagine Large Page Allocations ( /x /T661 /T834 )

# BEST PRACTICES: RETE

# Jumbo frames

- I Jumbo frames sono pacchetti ethernet con carico superiore a 1500 bytes
- Possono trasportare fino a 9000 bytes



# SQL Server: network packet size

- Il default è 4096
  - Si possono utilizzare 8192 bytes se i Jumbo frames sono disponibili per l'intero stack di rete

<http://blogs.msdn.com/b/johnhicks/archive/2008/03/03/sql-server-checklist.aspx>

# Per concludere...

- Utilizzare VMXNET3
  - Riduce l'utilizzo della CPU dell'host fisico
- Separare fisicamente il traffico di rete
  - SQL Server / Vmotion o Live Migration / Service Console
- Utilizzare più schede di rete ed il NIC teaming per ottenere performance e ridondanza
  - Minimo di 4 NICs per host
  - vSphere 5+ supporta più nic per il vMotion

# RISORSE

COMMUNITY DAYS 

# Risorse

- [Hardware and Software Requirements for Installing SQL 2012](#)
- [SQL Server 2012 Upgrade Guide](#)
- [Virtualizing Microsoft SQL Server on VMWare vSphere](#)
- [Achieving 1 Million I/O vSphere 5.0](#)
- [Virtualization articles](#) by David Klee
- [Virtualizing SQL Server with Windows Server 2012 & Hyper-V](#)
- [Hyper-V Dynamic Memory: Explanation and Recommendations](#)
- [Kevin Remde's 20+ days of Server Virtualization](#)

# Risorse (continua)

- SQLIO
  - [SQLIO Disk Subsystem Benchmark Tool](#)
  - [Choosing what SQLIO tests to Run](#)
  - [SAN performance tuning with SQLIO](#)
- [CrystalDiskMark](#)
- [IOmeter](#)
- [SQLIOSim](#)



# Risorse (continua)

- PASS ([www.sqlpass.org](http://www.sqlpass.org))
  - La più grande organizzazione al mondo di professionisti di SQL Server
- C'è una sezione dedicata alla virtualizzazione
  - <http://virtualization.sqlpass.org>
- Non mancate ai prossimi SQL Saturday
  - Eventi gratuiti di una giornata
  - Quest'anno ben 3 appuntamenti !!!

# Risorse (continua)



- Milano, 8 ottobre
- Verona, 9 novembre
- Ancona, 13 dicembre

# Q&A

Tutto il materiale di questa sessione su  
<http://www.communitydays.it/>

#CDays13

**GRAZIE !!!**