DataFrogs

Stockage SQL, virtualisation et performances

@mikedavem

Microsoft®
SQL Server®

> whoami

David Barbarin

Senior Database Administrator

DBA dans un monde de DevOps

 **@mikedavem**

 David Barbarin

 Blog

Agenda

- Dissection des IO SQL Server
- Fichiers / Groupes de fichiers et IO
- Configuration stockage et performance
- Virtualisation des IO

Dissection des IO SQL Server

Page ?

Fichier ?

IO asynchrones ?

Extent ?

Groupe de fichiers ?

IO synchrones ?

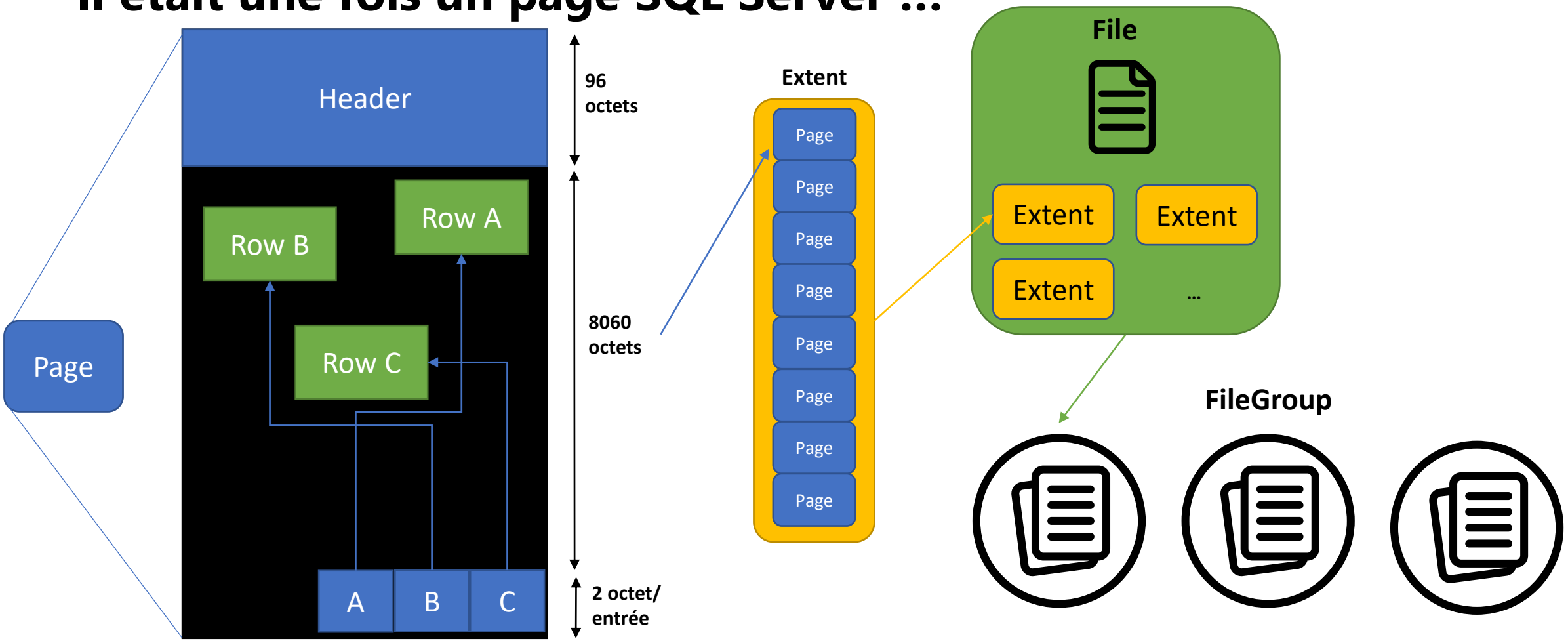
Scatter / Gather ?

IO Completion ports ?

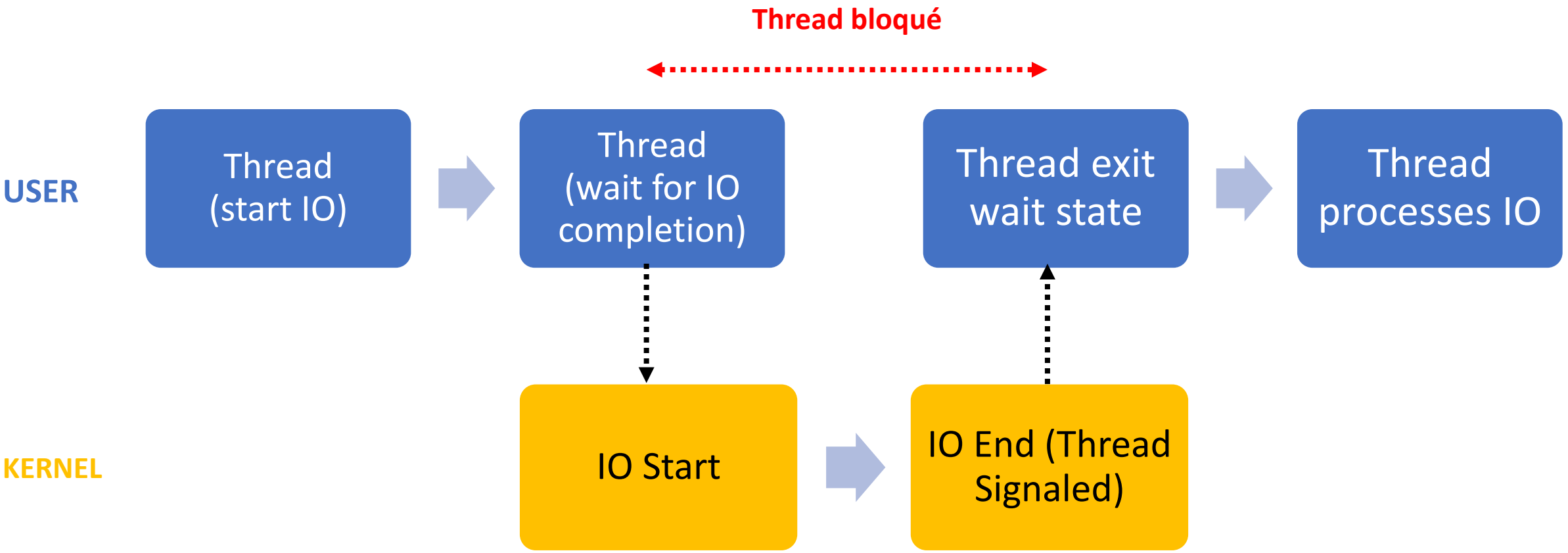
Instant file initialization?

Dissection des IO – Stockage interne

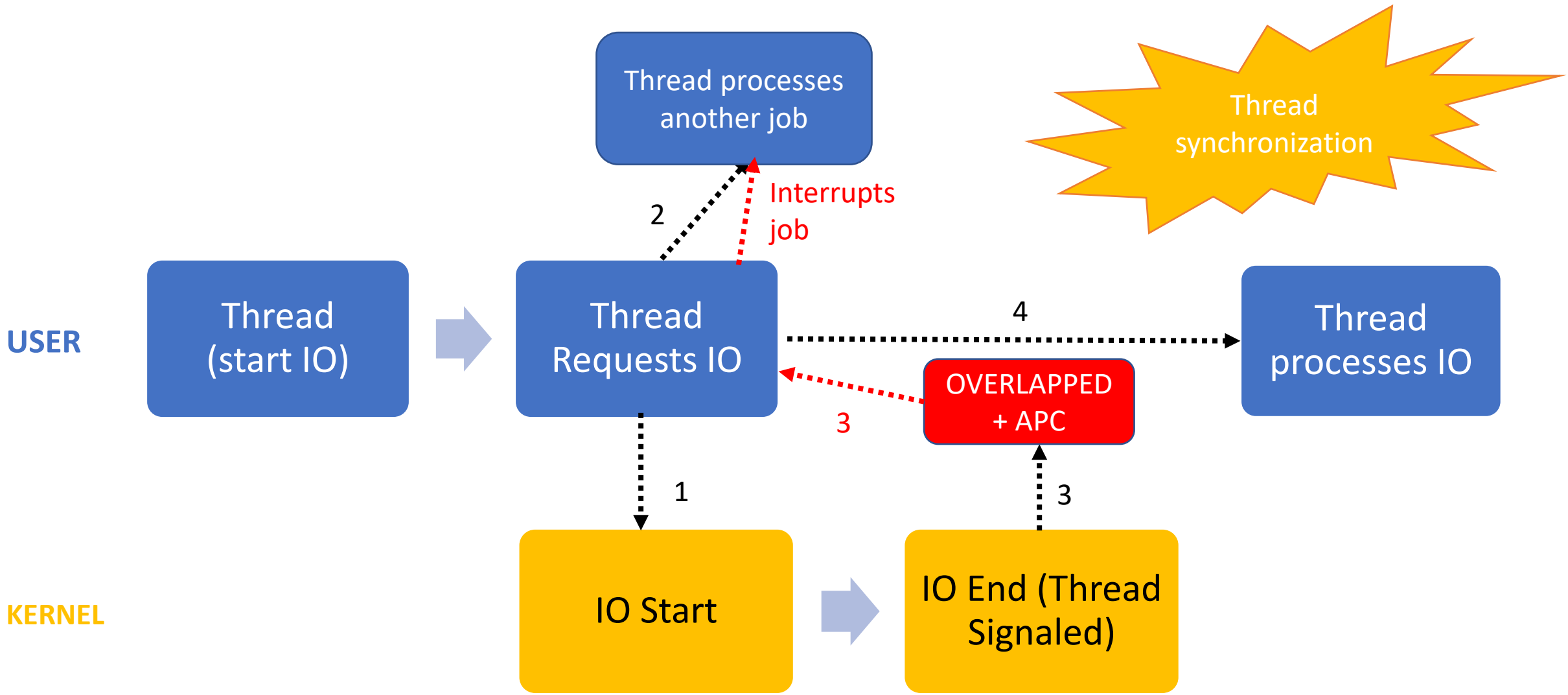
Il était une fois un page SQL Server …



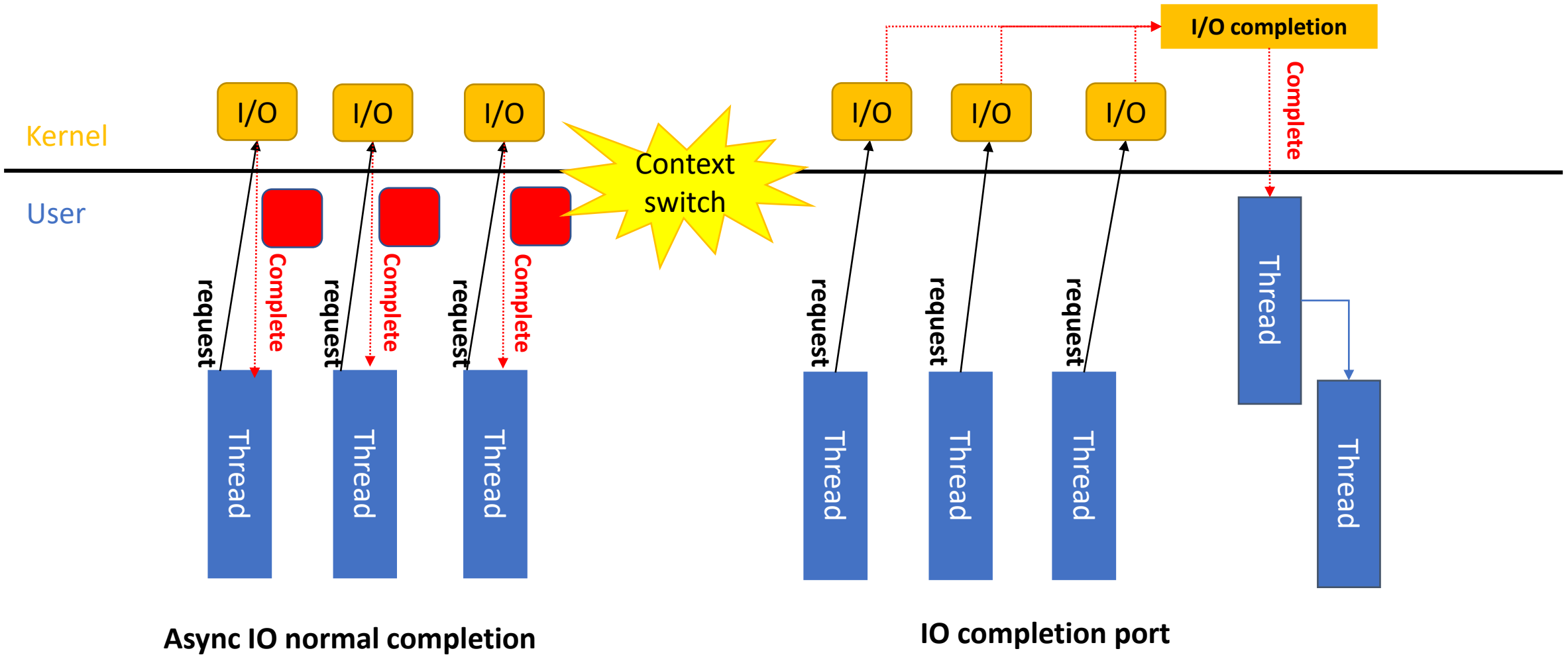
Dissection des IO – IO Synchrone



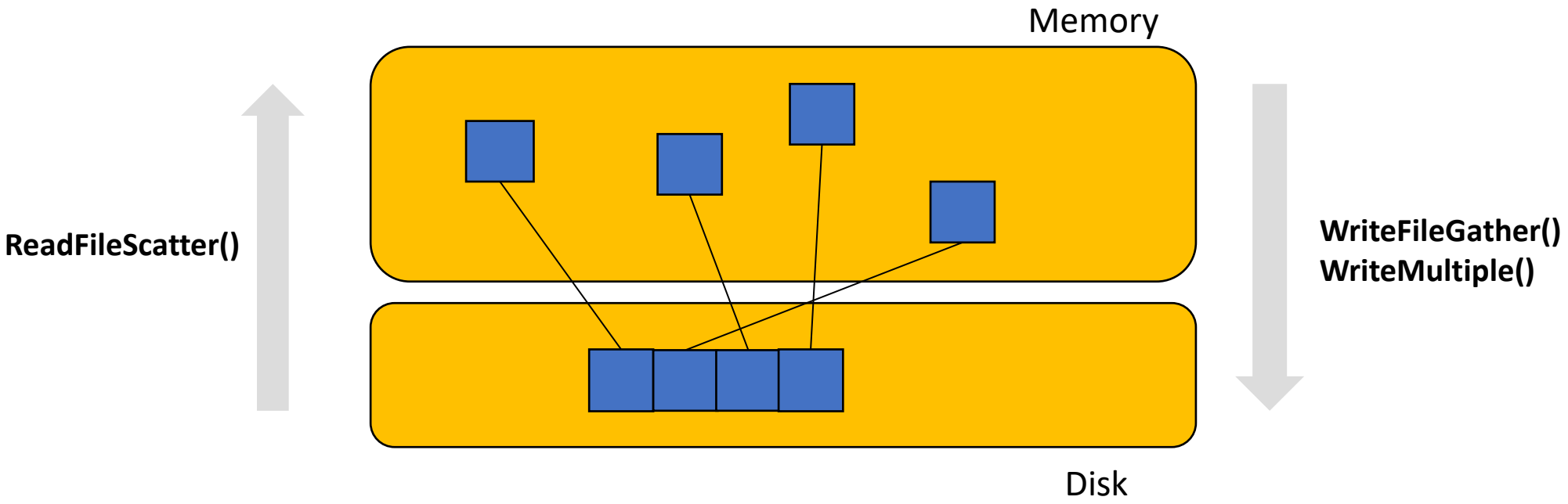
Dissection des IO – IO asynchrone



Dissection des IO – IOCP



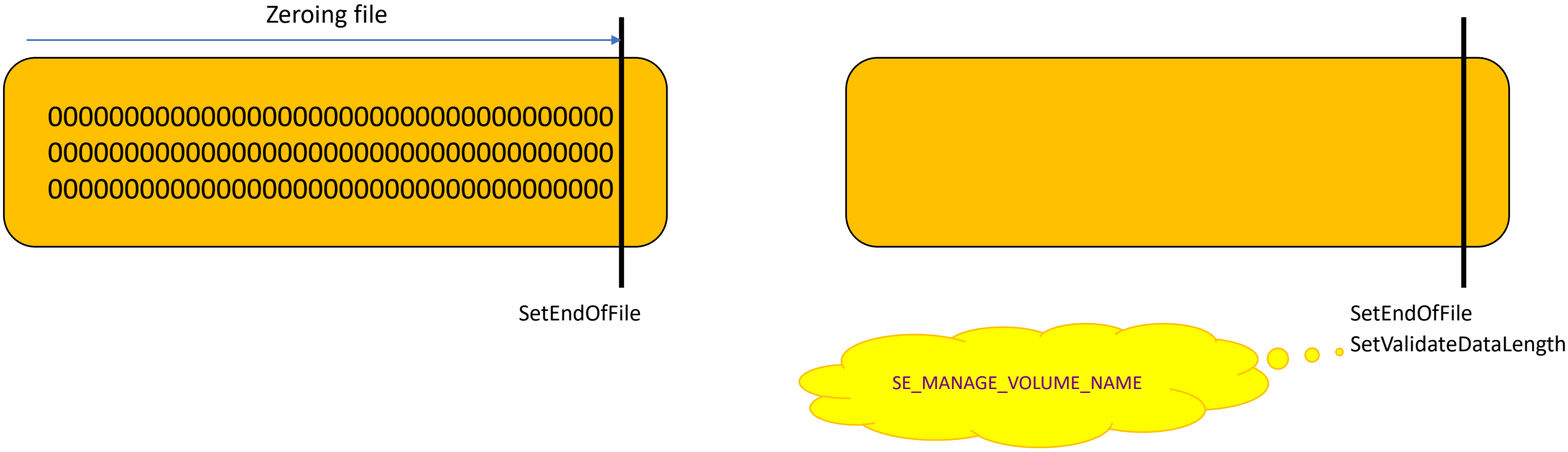
Dissection des IO – Scatter / Gather API



Prerequisites:

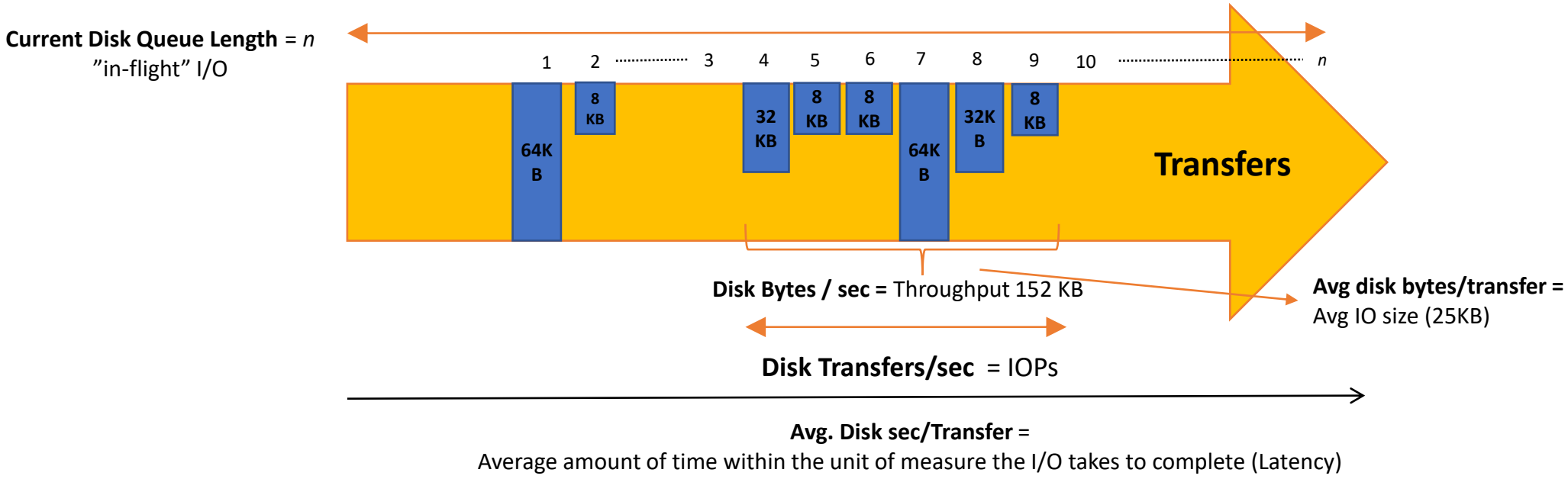
- Non cached IO
- Async IO
- Page aligned (device sector size / length multiple of sector size)

Dissection des IO – Instant File Initialization



Dissection des IO – TL/DR

- SQL Server utilise principalement des IO asynchrones non bufferisés
  - Ecrire ou lire des pages de données et maximiser l'utilisation CPU
  - Eviter qu'un worker reste bloqué à attendre le traitement d'une IO
- Nb IO > Longueur de file d'attente recommandée (2)



Dissection des IO- TL/DR

- Allocation unit size (64K) <> Taille d'une IO SQL Server
- Taille des IO SQL Server variable

Opération	Taille des IO	Type IO
Recherche d'index	8Ko – 64Ko	Aléatoire
Transaction log	512 octets – 60Ko	Séquentiel
Checkpoint/Lazywriter/ EagerWriter	8Ko – 1Mo	Séquentiel
Read-Ahead Scans	128Ko – 512Ko (EE) (8Mo pour CC)	Séquentiel
Bulk Loads	256Ko	Séquentiel
Backup/Restore	1Mo	Séquentiel
File Initialization	8Mo	Séquentiel

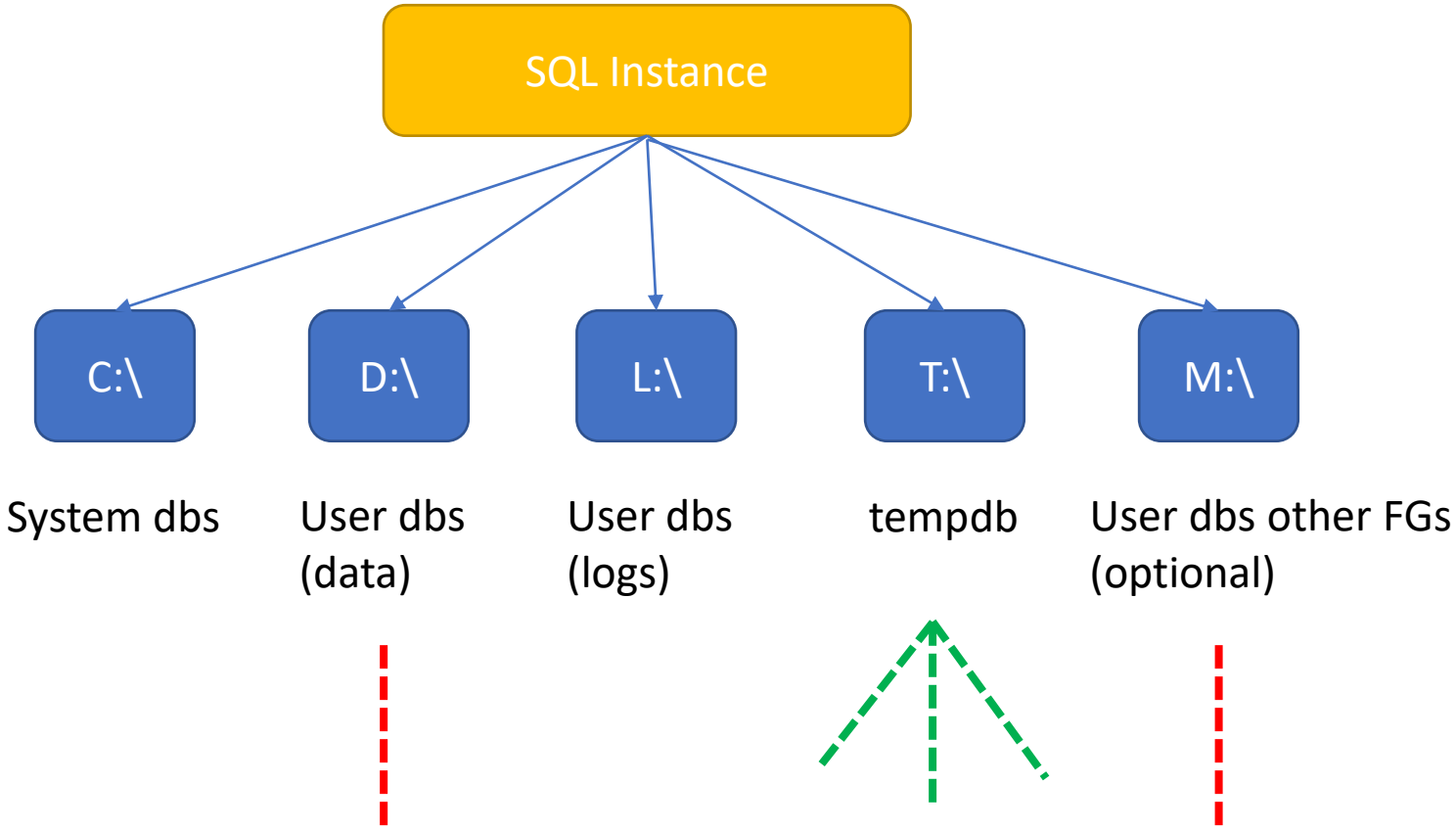
Fichiers / Groupe de fichiers et IO

Fichiers multiples ou groupes de fichiers multiples ?

- Groupe de fichier primaire + Journal des transactions
- Fichiers multiples dans un groupe de fichiers
  - Allocation des pages de données round robin + remplissage proportionnel
- Groupe de fichiers multiples
  - Parallélisation des IO
    - Jointure de tables
    - Key / RID lookup (peu utilisé)
    - Partitionnement (exception avec ORDER BY)

Fichiers / Groupe de fichiers et IO

Fichiers multiples ou groupes de fichiers multiples



Configuration stockage et performance

Bonnes pratiques Microsoft

- Alignement des partitions (par défaut depuis 2008)
- Formatage volume = 64Ko ~Taille extent (optimise allocation NTFS)
  - Ne correspond pas aux tailles IO SQL Server
  - Paramètre /L (format) ou –UseLargeFRS (Format-Volume) – CHECKDB error 665
- Nb de fichiers pour tempdb (BP Microsoft)
- Nb de fichiers Data / Log
  - Dépend du contexte et performance du stockage
  - Sauvegardes sur un disque distinct
  - Si virtualisation -> voir considérations slides suivants
- IFI (si pas de contrainte de sécurité)
- Exclusion antivirus

Configuration stockage et performance

Identification de la charge de travail

- OLTP = IOPs + Latence => Nombre d'axes disques important …
- DS = Débit et accès séquentiels => Stack contrôleurs + réseau + caches

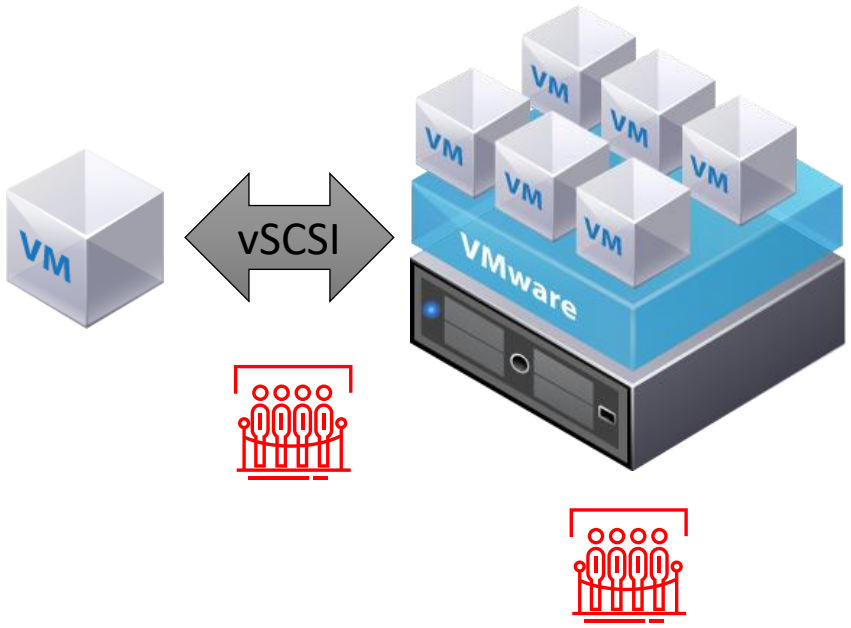
Evolution des technologies de stockage

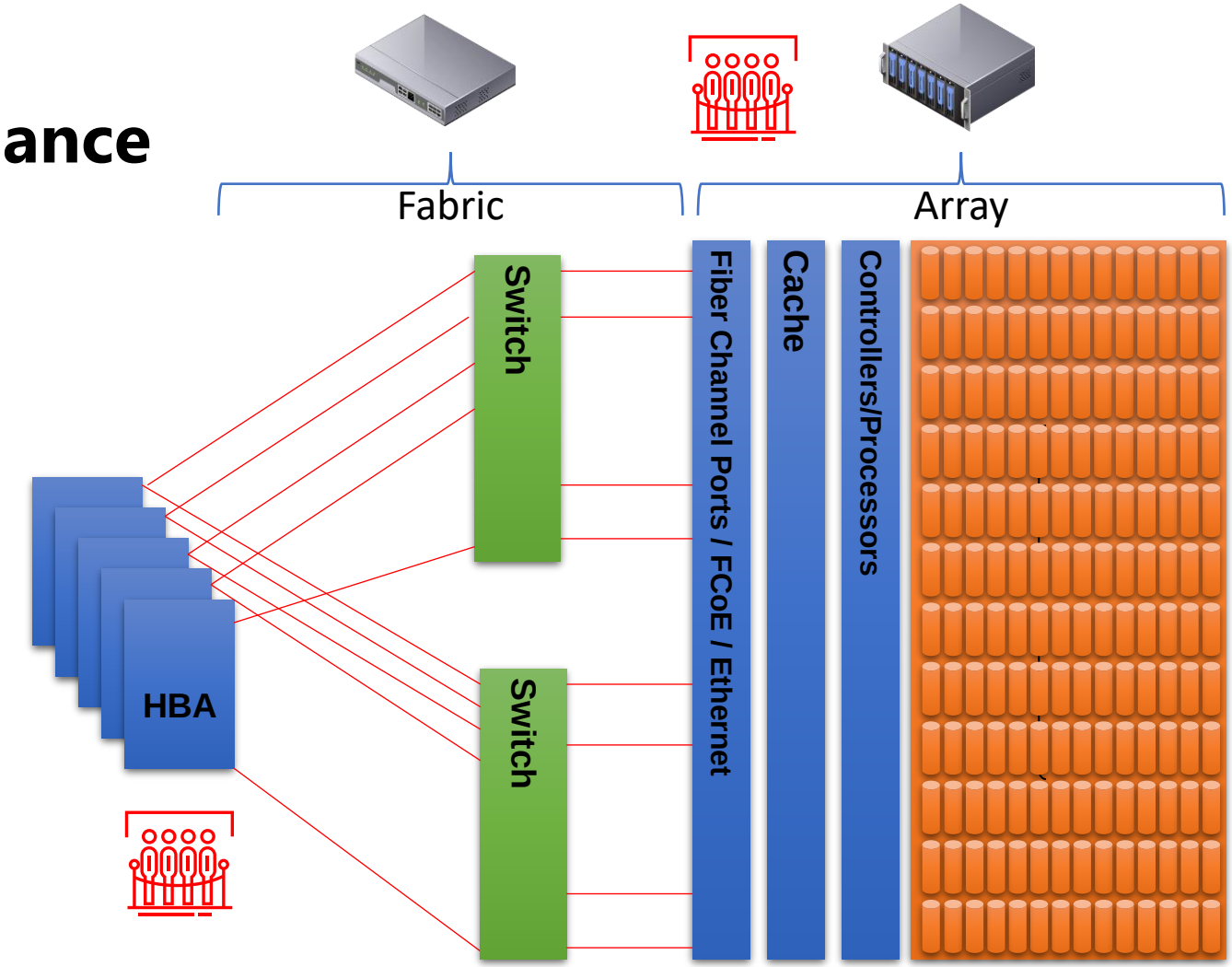
- Disques mécaniques 10K/15K (RAID / Short Strocking …) => Disques flash
- Stockages d'entreprises avec caches intermédiaires en SSD
- Baies full flash

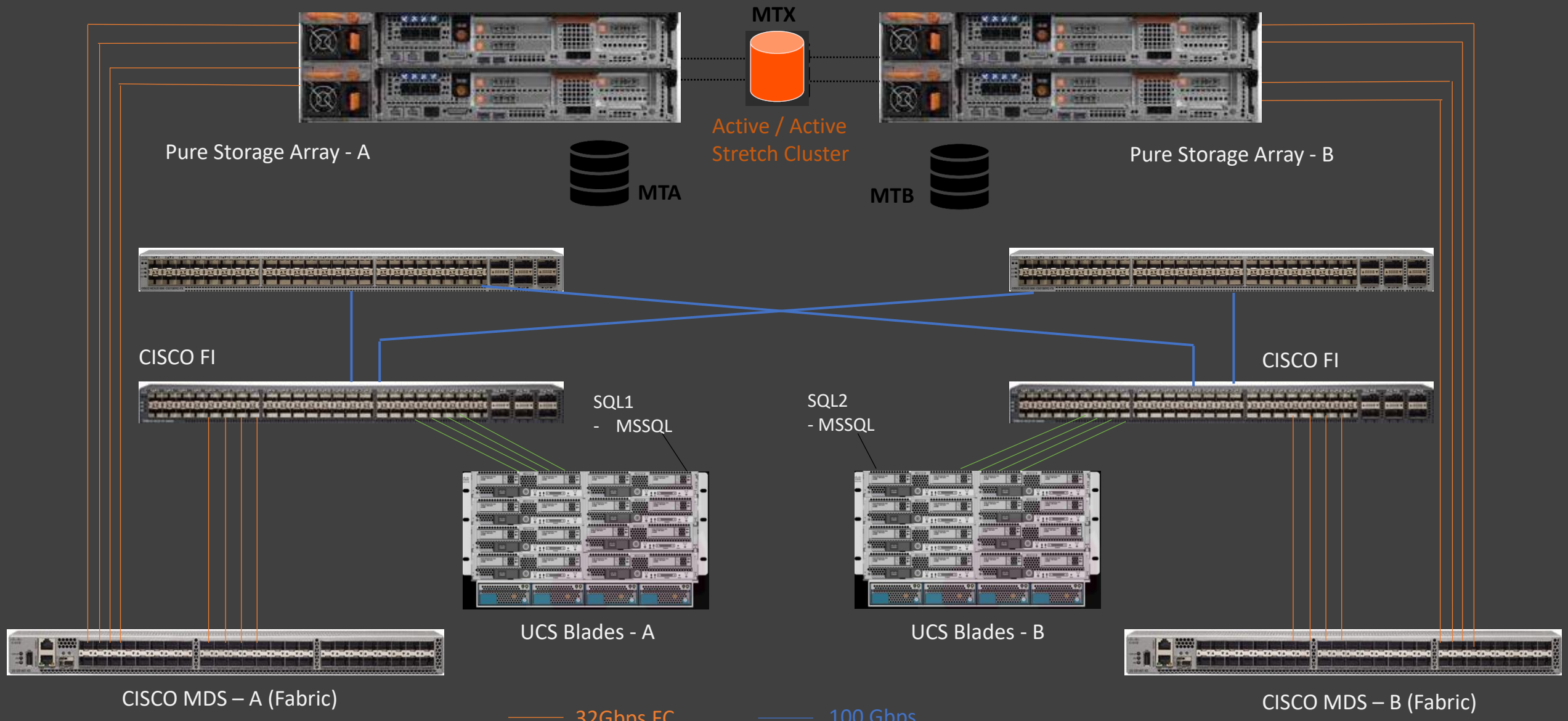
Virtualisation des IO – Stockage d'entreprise

Chemin IO et performance

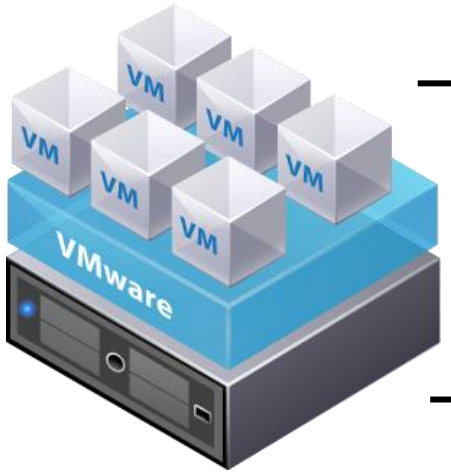
- Files d'attentes
- Débit des liens d'interconnexion
- Contrôleurs et Stockage






MTX
Active / Active
Stretch Cluster
Pure Storage Array - A
Pure Storage Array - B
MTA
MTB
CISCO FI
CISCO FI
SQL1
- MSSQL
SQL2
- MSSQL
UCS Blades - A
UCS Blades - B
CISCO MDS – A (Fabric)
CISCO MDS – B (Fabric)
32Gbps FC
100 Gbps

Virtualisation des IO – Blender effect



Séquentiel

Squentiel

→ Random

Trop de threads séquentiels sur une LUN

Random

Squentiel

→ Random

Mix Séquentiel + Aléatoire

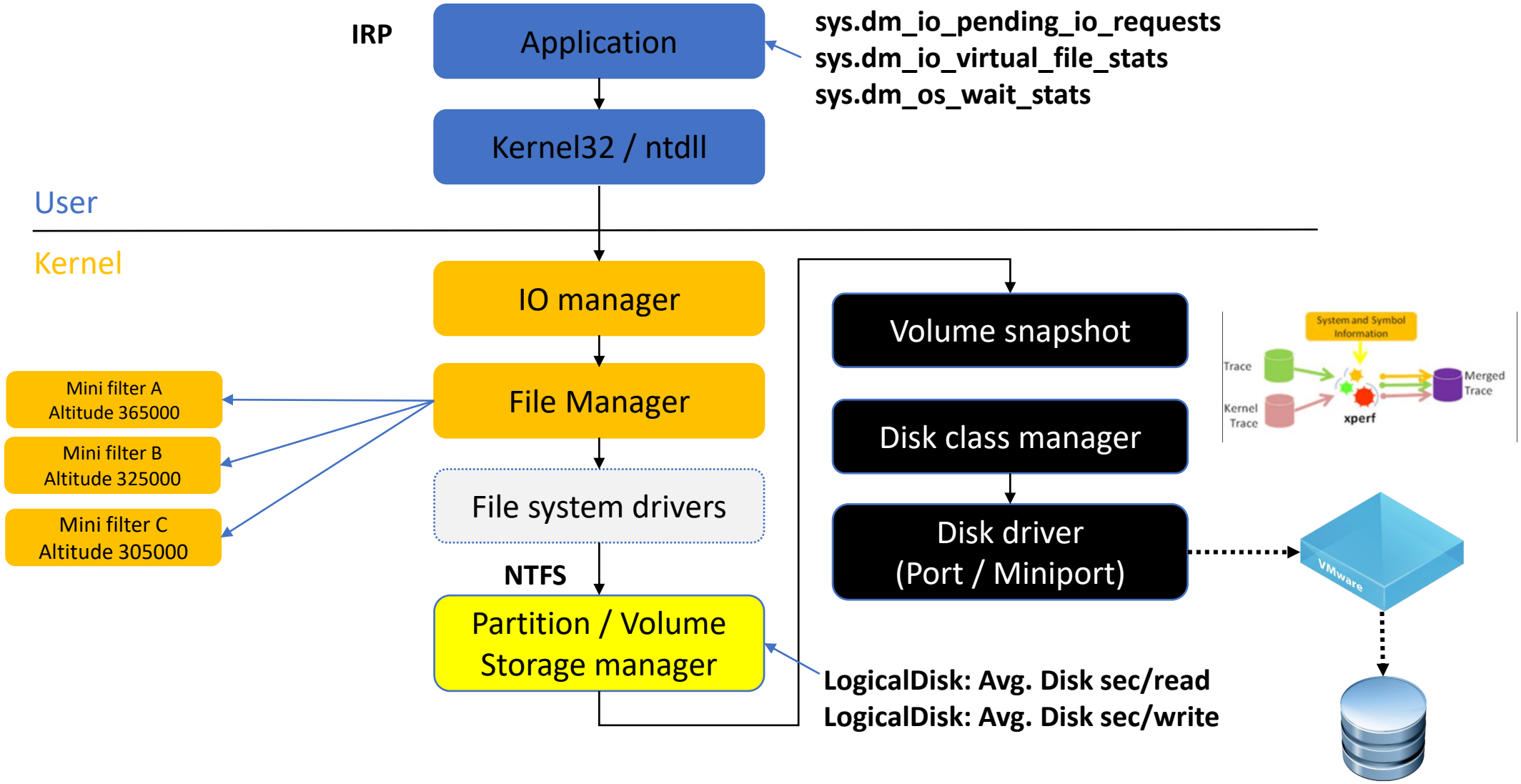
Random

Random

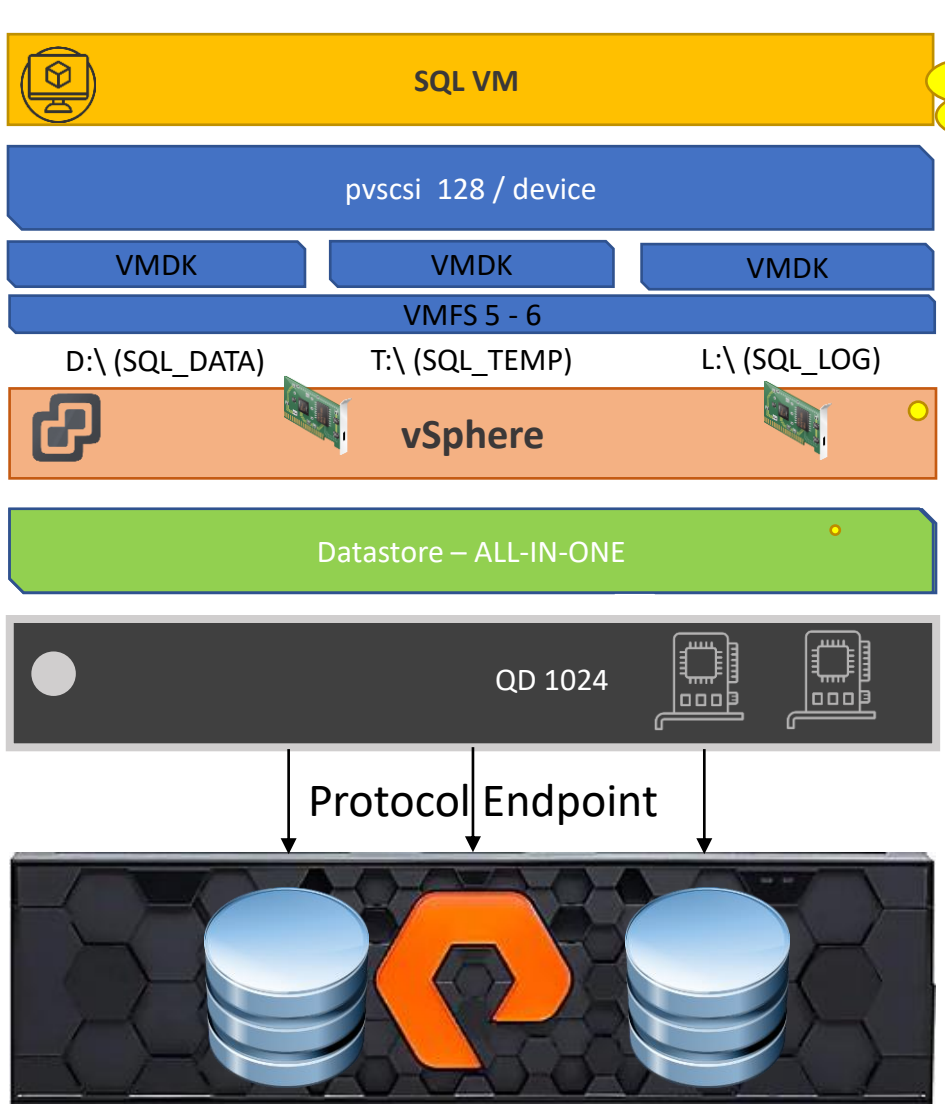
→ Random

Grouper des workloads similaires ?

Virtualisation des IO – Couche OS



Virtualisation des IO – Couches de stockage



Blender
+ Funnel

vSCSI Adapter

ESX Kernel

DataStores

Driver (fnic)

HBA

- LSI Logic SAS versus PVSCSI (HWv7 + VMTools)
  - Coût CPU plus faible par IO et plus faible latence
  - File d'attente par défaut => 32 vs 64 – 256
  - Clé de registre Windows
- Disques virtuels
  - Thin vs Thick Eager Zero => Support VAAI / VASA
  - Shares
- Nb de contrôleurs virtuels + Datastore
  - >= 2 (SQL DATA et SQL LOG)
  - 1 VMDK ⇔ 1 Datastore => Files d'attentes //
  - Performance vs management (ex. snapshots)
  - Attention à l'overcommitment -> Cf Slide suivant
- HBA
  - Valeur par défaut dépend du modèle de la carte
  - Qlogic (64) / Brocade (32) / Emulex (32) / Cisco UCS (32)
- Storage
  - Spécifications vendeur
  - Ex. Pure Storage PE >2048

Virtualisation des IO - Surchargement

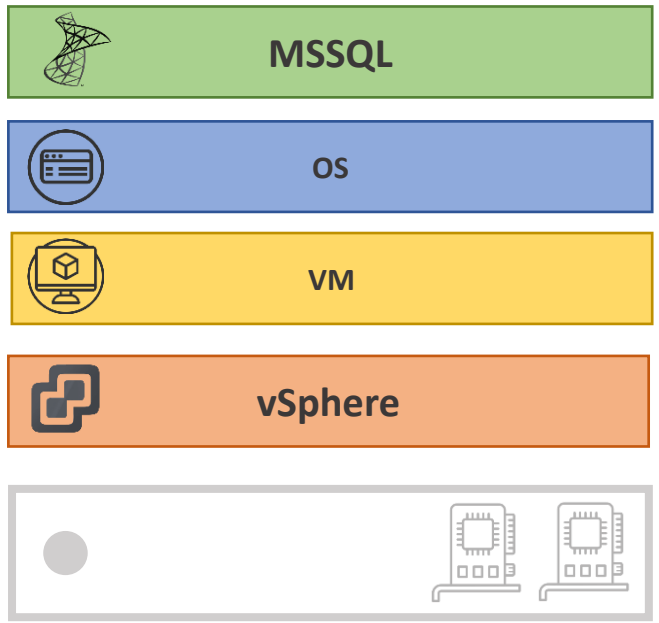
Files d'attentes en environment partagé

Max outstanding IO per LUN (n)	Avg. Active IO per VM (a)	Lun Queue Depth (d)	Max. VM per Host m = d/a	Max. VM on Data Store m = n/a
32	16	32	2	2
128	16	128	8	8
128	32	128	4	4
1024	32	256	8	32

Storage IO Control

- Gestion des files d'attentes IO dynamiques (v1 = Stockage, v2 = Disque)
- Basé sur les proportions de partage (importance) et seuil de latence IO

Virtualisation des IO – Métriques de latence



Application

vSCSI Adapter
DataStores

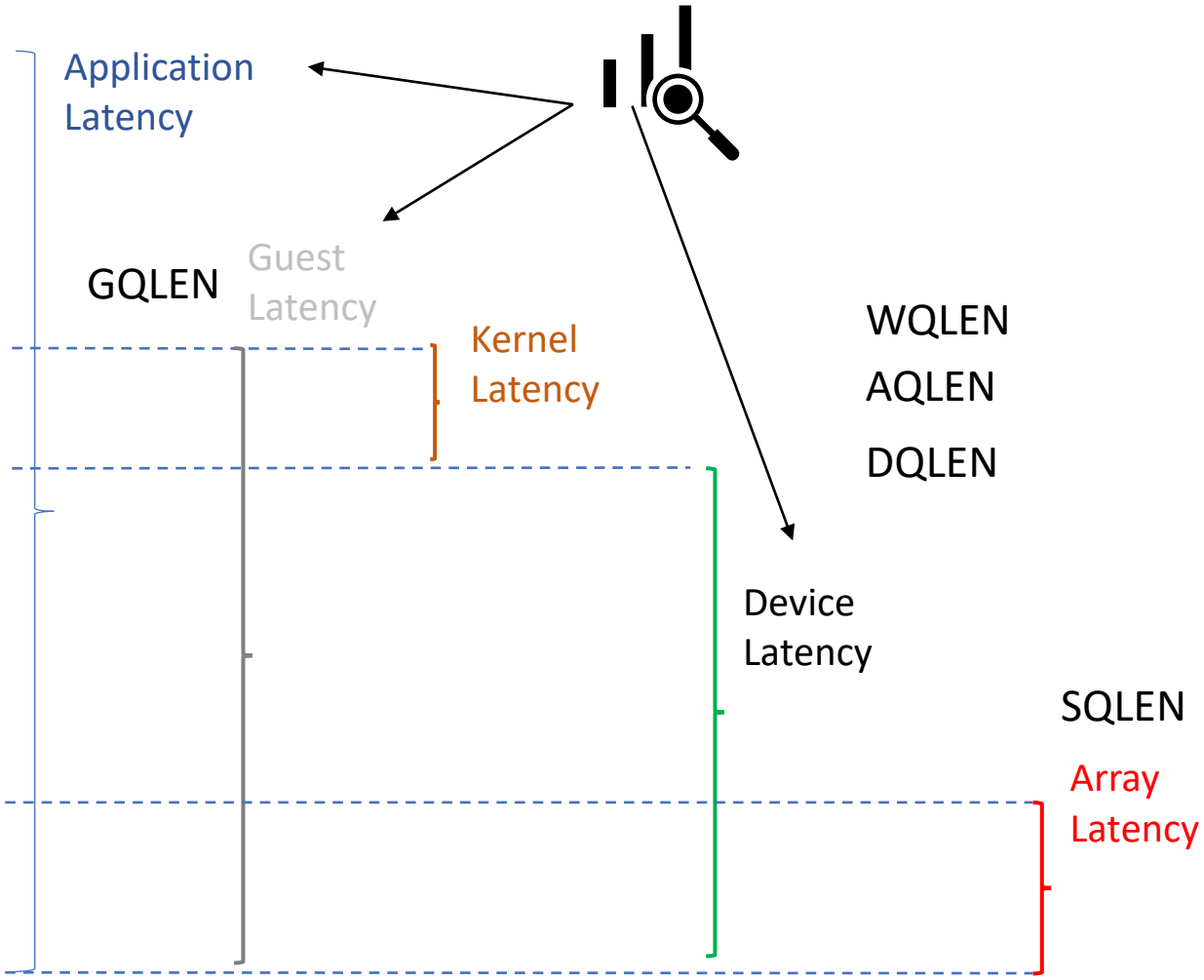
ESX Kernel

Driver

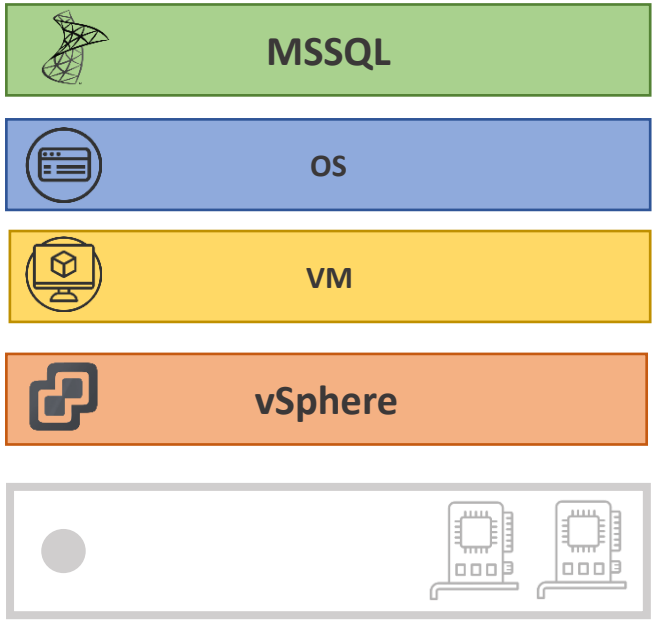
HBA

Fabric

Array



Virtualisation des IO – PVSCI driver



Application

vSCSI Adapter

DataStores

ESX Kernel

Driver

HBA

Fabric

Array

PVSCSI – Configuration – Device –> 128 / Device

```
REG ADD HKLM\SYSTEM\CurrentControlSet\services\pvscsi\Parameters\Device /v DriverParameter /t REG_SZ
/d "RequestRingPages=32,MaxQueueDepth=128"
```

PVSCSI – Configuration – Adapter –> 1024 / Device

```
[root@sqlesxi01:~] esxcli system module parameters list -m nfnic
Name                    Type   Value  Description
----------------------  -----  -----  -----------
ecpu_ka_timeout         ulong         nfnic ecpu keep alive timeout: Default = 0. Range [10 - 120] seconds. 0 to turn off.
log_throttle_count      ulong         nfnic log throttle count: Default = 64
lun_queue_depth_per_path ulong  1024  nfnic lun queue depth per path: Default = 32. Range [1 - 1024]
[root@sqlesxi01:~]
```

```
Device Max Queue Depth: 1024
No of outstanding IOs with competing worlds: 1024
```

Cas d'utilisation: file d'attente PVSCI – QD 64
Diskspd 8 threads, 128 outstanding , block 4K

Virtualisation des IO

\\MTASQLPUREP01

LogicalDisk	D:
Avg. Disk Queue Length	876,188
Avg. Disk Read Queue Length	876,188
Avg. Disk sec/Read	0,005
Disk Read Bytes/sec	787 388 740,538
Disk Reads/sec	192 233,579

Latence vue depuis le guest

File d'attente configurée pour datastore

Latence carte HBA -> Stockage

Latence vue depuis ESX Kernel

Slots actifs concurrents

IO en attente > DQLEN

esxtop command

DEVICE	PATH/WORLD/PARTITION	DQLEN	WQLEN	ACTV	QUED	%USD	LOAD	CMDS/s	READS/s	WRITES/s	MBREAD/s	MBWRTN/s	DAVG/cmd	KAVG/cmd	GAVG/cmd	QAVG/cmd
mpx.vmhba32:C0:T0:L0	-	1	-	0	0	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
naa.6000144000000010e006f6438633d855	-	1024	-	0	0	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
naa.6000144000000010e006f6438633d8bf	-	1024	-	0	0	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
naa.6000144000000010e006f6438633d8c4	-	1024	-	0	0	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
naa.6000144000000010e006f6438633d8c9	-	1024	-	0	0	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
naa.6000144000000010e006f6438633d8eb	-	1024	-	0	0	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
naa.6000144000000010e006f6438633d8f2	-	1024	-	0	0	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
naa.6000144000000010e006f6438633d8f9	-	1024	-	0	0	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
naa.6000144000000010e006f6438633d900	-	1024	-	0	0	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
naa.6000144000000010e006f6438633da49	-	1024	-	0	0	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
naa.6000144000000010e006f6438633da8f	-	1024	-	0	0	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
naa.6000144000000010e006f6438633db2f	-	1024	-	0	0	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
naa.6000144000000010e006f6438633db4d	-	1024	-	0	0	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
naa.600507624a8280c2480000000008000012	-	1024	-	0	0	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
naa.600507624a8280c2480000000009000013	-	1024	-	0	0	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
naa.624a93701ef85d7e4a844b0f0001144a	-	256	-	0	0	0	0.00	0.38	0.00	0.00	0.00	0.00	0.32	0.03	0.35	0.00
naa.624a93701ef85d7e4a844b0f00011b99	-	1024	-	0	0	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
naa.624a93701ef85d7e4a844b0f00011b9a	-	1024	-	0	0	0	0.00	5.15	0.00	4.77	0.00	0.03	0.34	0.02	0.36	0.00
naa.624a93701ef85d7e4a844b0f00011b9b	-	1024	-	0	0	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
naa.624a93701ef85d7e4a844b0f00011d1c	-	1024	-	0	0	0	0.00	0.38	0.00	0.00	0.00	0.00	0.89	0.04	0.93	0.00
naa.624a93701ef85d7e4a844b0f00011e52	-	1024	-	40	0	3	0.04	207217.60	206274.61	936.70	806.06	1.84	0.17	0.01	0.18	0.00
naa.624a93701ef85d7e4a844b0f00013619	-	1024	-	0	0	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
naa.624a9370656d13c3a1e7492800011447	-	23	-	0	0	0	0.00	0.38	0.00	0.00	0.00	0.00	0.41	0.04	0.45	0.00
naa.624a9370656d13c3a1e7492800011b99	-	1024	-	0	0	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
naa.624a9370656d13c3a1e7492800011b9a	-	1024	-	0	0	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
naa.624a9370656d13c3a1e7492800011b9b	-	1024	-	0	0	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
naa.624a9370656d13c3a1e7492800011e52	-	1024	-	0	0	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Conclusion

Configurer le stockage pour la performance c'est …

- Appliquer les bonnes pratiques de configuration Microsoft SQL Server
- Appliquer les bonnes pratiques de configuration VMWare pour SQL Server
- Connaître les spécifications du stockage sous-jacent (virtualisation incluse)
- Configurer un chemin IO optimale depuis SQL Server vers le stockage
- Inclure le «noisy neighbor» dans le design (Environnement partagé)
- Tester les performances du stockage et corriger si besoin
  - Identifier la charge de travail SQL (OLTP/DW/Mixte + Ratio R/W …)
  - Benchmarks versus Tests IO synthétiques

Merci!

Webographie / Bibliographe

- The Guru's Guide to SQL Server Architecture and Internals
- Pure Storage SQL Server blogs
- Windows Internal Books
- Virtualizing SQL Server with VMware: Doing it Right
- VMWare KB 2053145