March-27-2021

# Welcome

# To


08:30am – 5pm EDT-US

# Haitian

# Data Saturday

The Greatest Island

# Who We Are?

- **Support Community Data Platform.**
- **Deliver Free contents related to:**
  - **Enterprise Database Administration & Deployment**
  - **Cloud Application Development & Deployment**
  - **Advanced Analysis Techniques  and Visualization**
  - **BI Information Delivery**
  - **Big Data**
  - **Data Sciences**
  - **Professional Development**
  - **Others**

# Thank You To Our Organizer, Contributors, and Speakers

**Organizer**

- Jean Joseph

**Volunteer**

- Armando Lacerda
- Pragati Jain

**Contributors**

- Anie Dulipre
- André M.D. Melancia
- Gianluca Sartori

## Speakers

- Cédric Charlier
- Seddryck@gmail.com
- Christophe LAPORTE
- Charles-Henri Sauget
- Matthieu Roy
- Charles-Henri Sauget
- Matthieu Roy
- Fabien Adato
- Mikey Bronowski
- John Miner
- Jean Joseph
- *Anna Hoffman*
- Kathi Kellenberger
- Amit Bansal
- Hugo Kornelis
- Benni De Jagere
- Dennes Torres
- Erik Monchen
- Ahmad Osama
- Tracy Boggiano
- Markus Ehrenmüller-Jensen
- Andrew Pruski
- James Serra

## Thank You Microsoft Employees

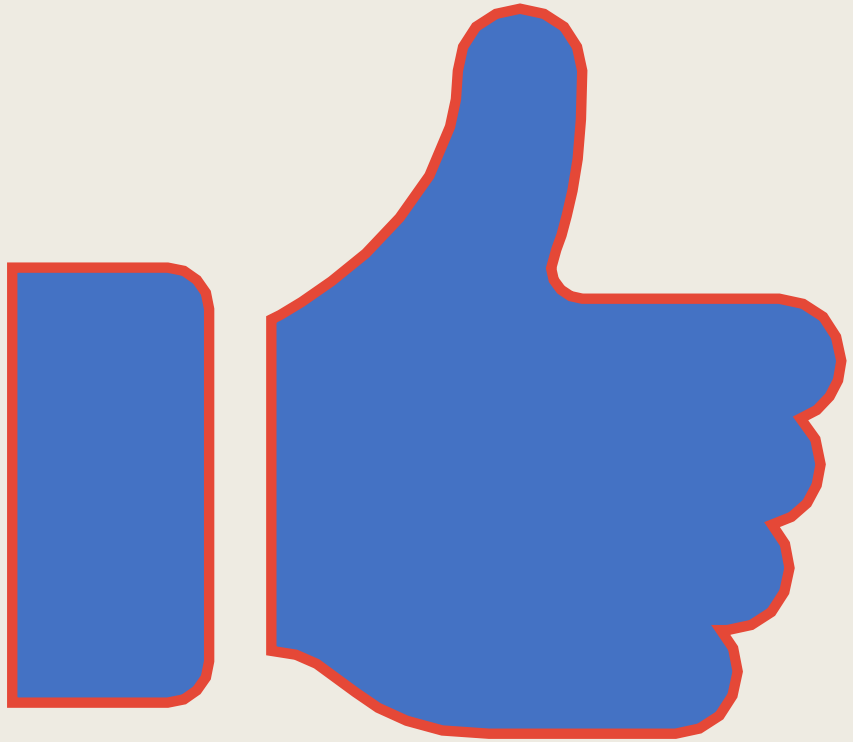
***SPEAKERS***

- **Anna Hoffman** – *Data Scientist*
- **James Serra** – *Former recent Technical Architect*

***Contributor & Ass Organizer***

- **Jaimie Fox** – *Sr. Account Technology Strategist*

# Haitian



# Data Saturday

## Follow Us

At **#HaitianDataSaturday** &
**#HaitianDataDriven**

**Join our Meetup Group:**
https://www.meetup.com/haitian-data-driven/

**Facebook Group:**
https://www.facebook.com/groups/713234249372103/

**LinkedIn Group:**
https://www.linkedin.com/groups/12515982/

**Like Our Facebook Page:**

https://www.facebook.com/Haitiandatasaturday-107853578011523


# Who Am I?

Christophe Laporte

Consultant & Trainer

24 years plus in IT

Located in Toulouse - France

Your host for today

**The Greatest Island**

# *Christophe LAPORTE*

**/in/christophelaporte**

**conseilit.wordpress.com**

**@conseilit**

**conseilit@outlook.com**

- Audit
- Conseil
  - Infrastructure / Architecture
  - Virtualisation / Cloud
  - Haute disponibilité
  - Performance / Optimisation
  - Dépannage / Migrations
- Formations
- Remote DBA

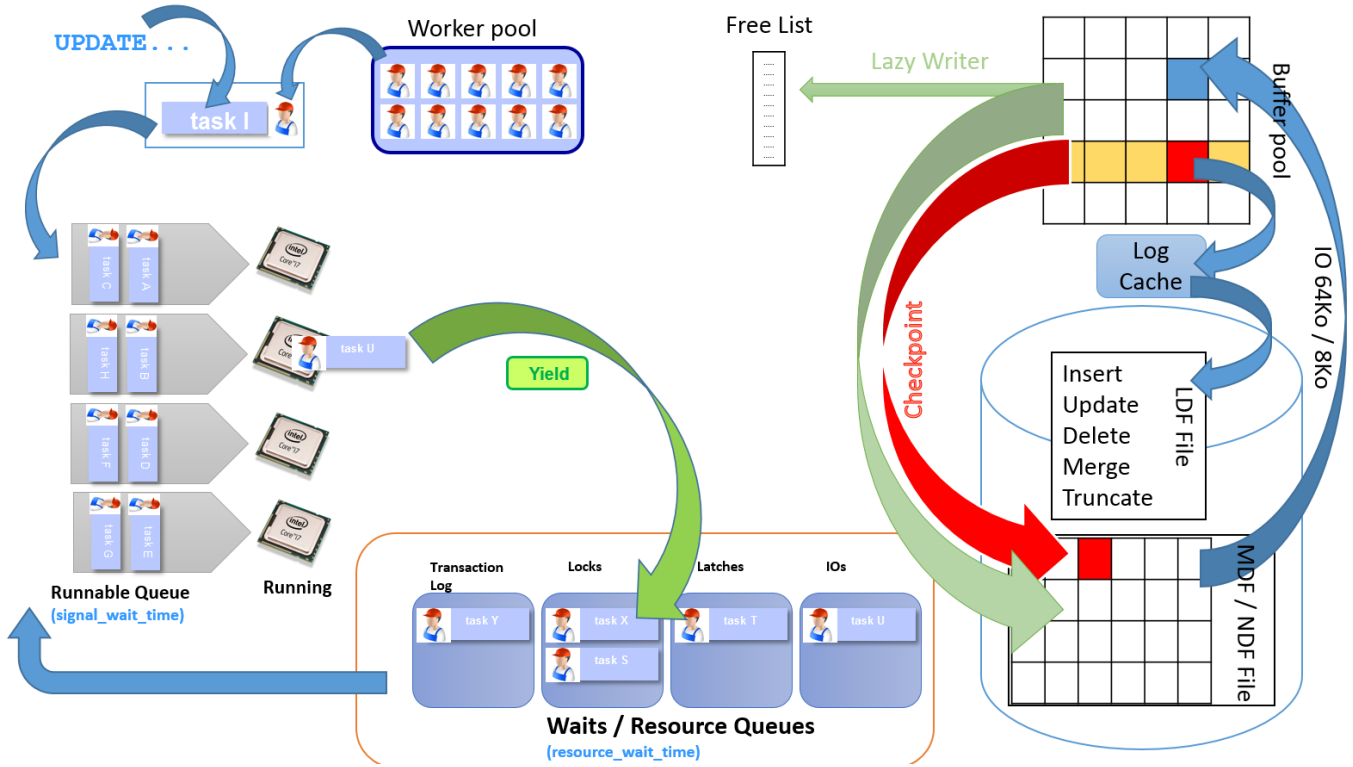


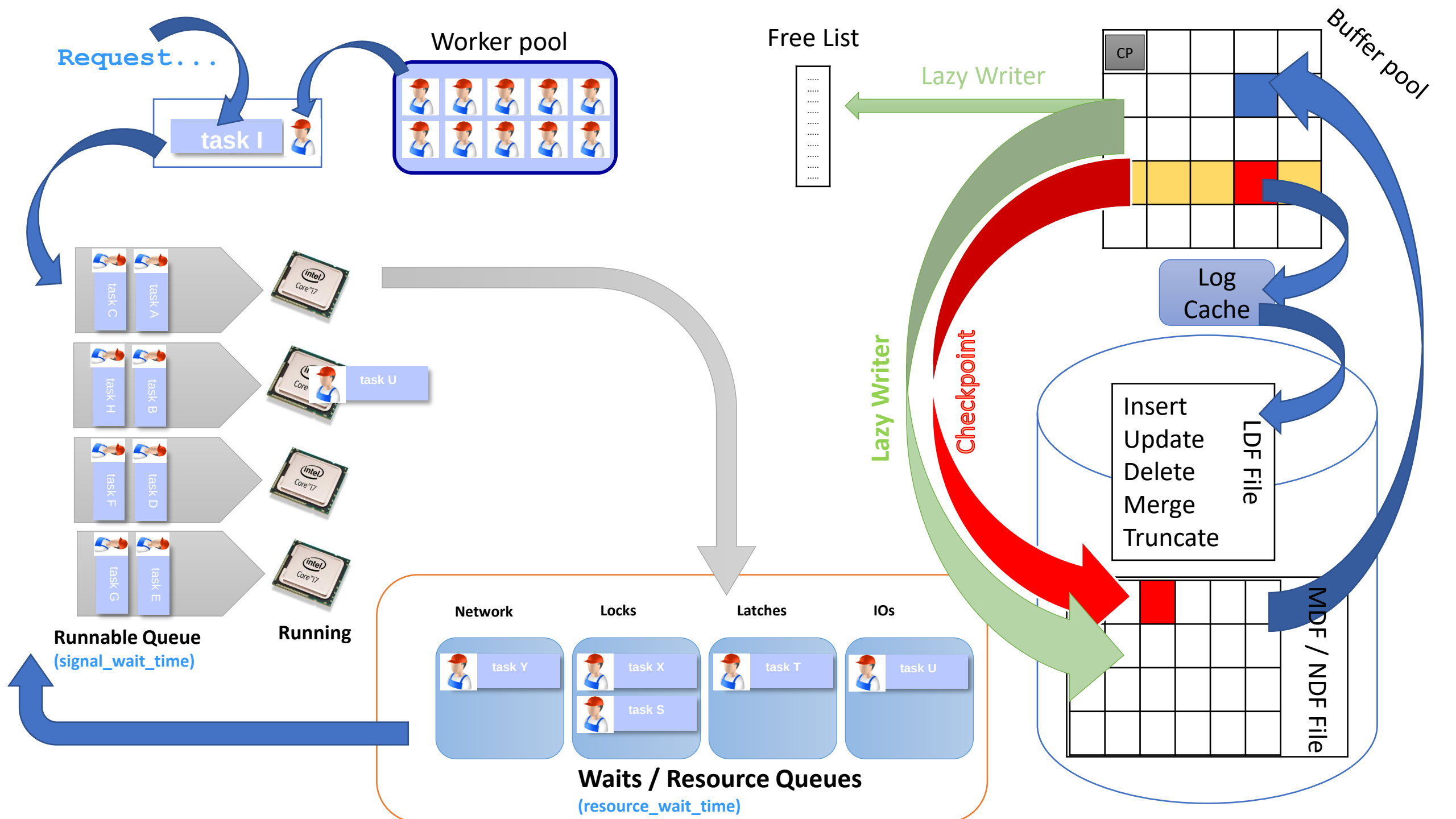
# Agenda

- Fonctionnement de SQL Server
- Configurer SQL Server pour un dépannage efficace
- Technique des Wait And Queues
- Les DMVs indispensables

# Fonctionnement SQL Server

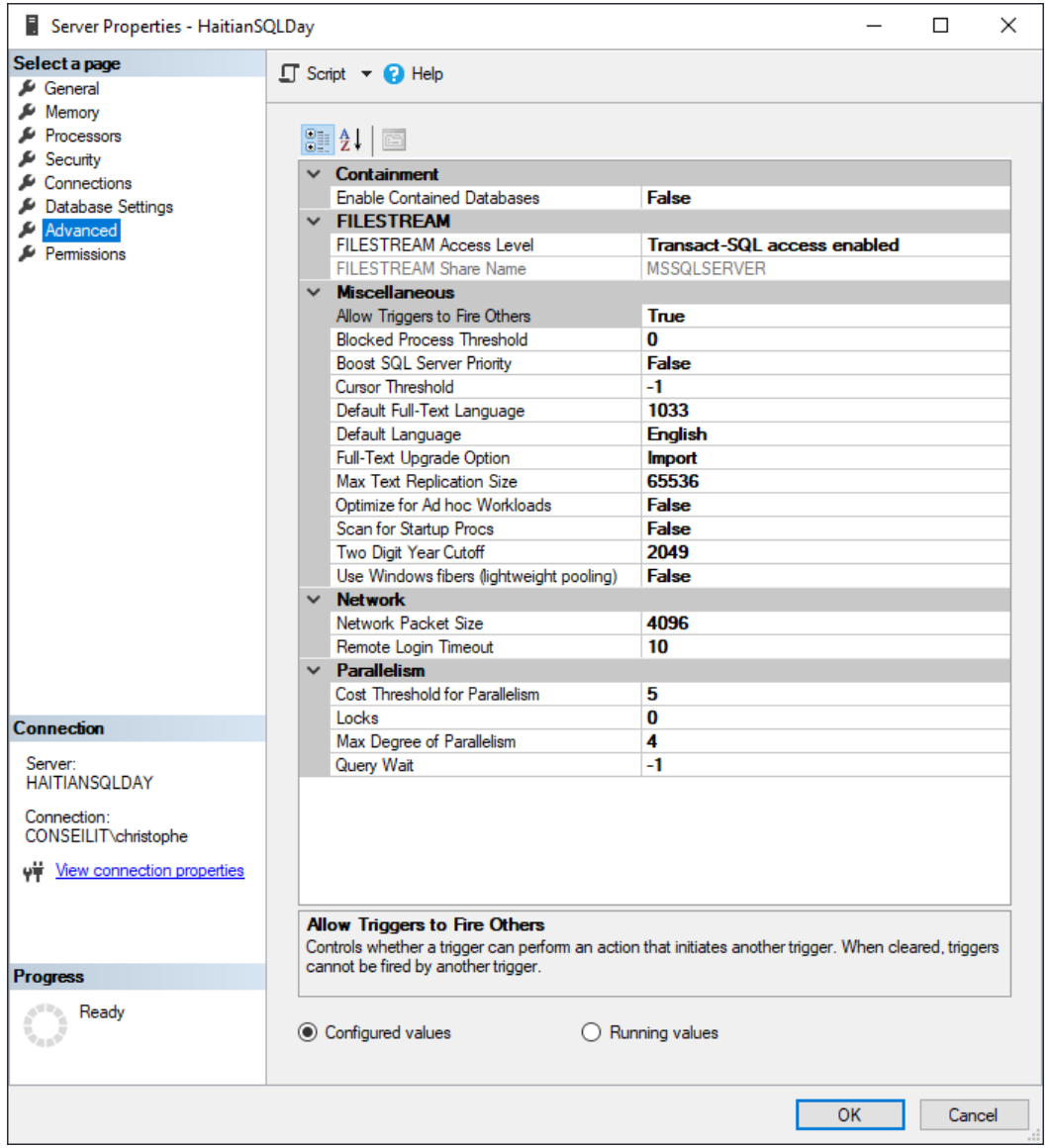
10 000 ft view




Request...
task I
Worker pool
Free List
Lazy Writer
Buffer pool
CP
Log Cache
Lazy Writer
Checkpoint
Insert
Update
Delete
Merge
Truncate
LDF File
MDF / NDF File
task C
task A
task H
task B
task F
task D
task G
task E
task U
Runnable Queue
(signal_wait_time)
Running
Network
Locks
Latches
IOs
task Y
task X
task S
task T
task U
Waits / Resource Queues
(resource_wait_time)

# Configuration SQL Server

Pour un dépannage efficace



# Configuration SQL Server

- Installation SQL « correcte »
- Rétention trop faible
  - Errorlog, xEvent system_health, Job history

```
Stop-DbaXESession -SqlInstance $Server -Session "system_health"

Invoke-DbaQuery -SqlInstance $Server -Database "master" -Query "
    ALTER EVENT SESSION [system_health] ON SERVER
    DROP TARGET package0.event_file;
    GO
    ALTER EVENT SESSION [system_health] ON SERVER
    ADD TARGET package0.event_file
        (SET FILENAME=N'system_health.xel',
            max_file_size=(25),
            max_rollover_files=(20)
        );
    GO
    ALTER EVENT SESSION [system_health] ON SERVER
    DROP EVENT sqlserver.security_error_ring_buffer_recorded;
"
Start-DbaXESession -SqlInstance $Server -Session "system_health"
```

```
# increase history retention
Set-DbaErrorLogConfig -sqlinstance $Server -LogCount 99
Set-DbaAgentServer -sqlinstance $Server -MaximumHistoryRows 999999 -MaximumJobHistoryRows 999999
```

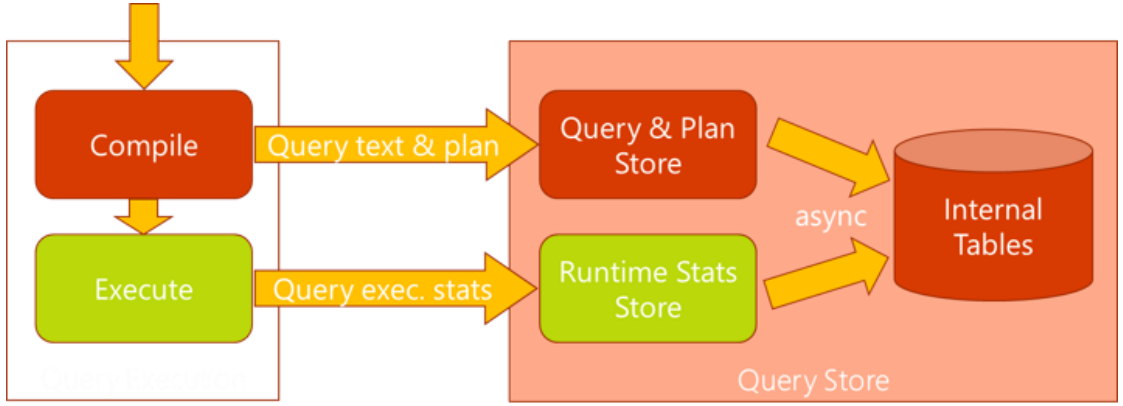
# Session d’évènements étendus

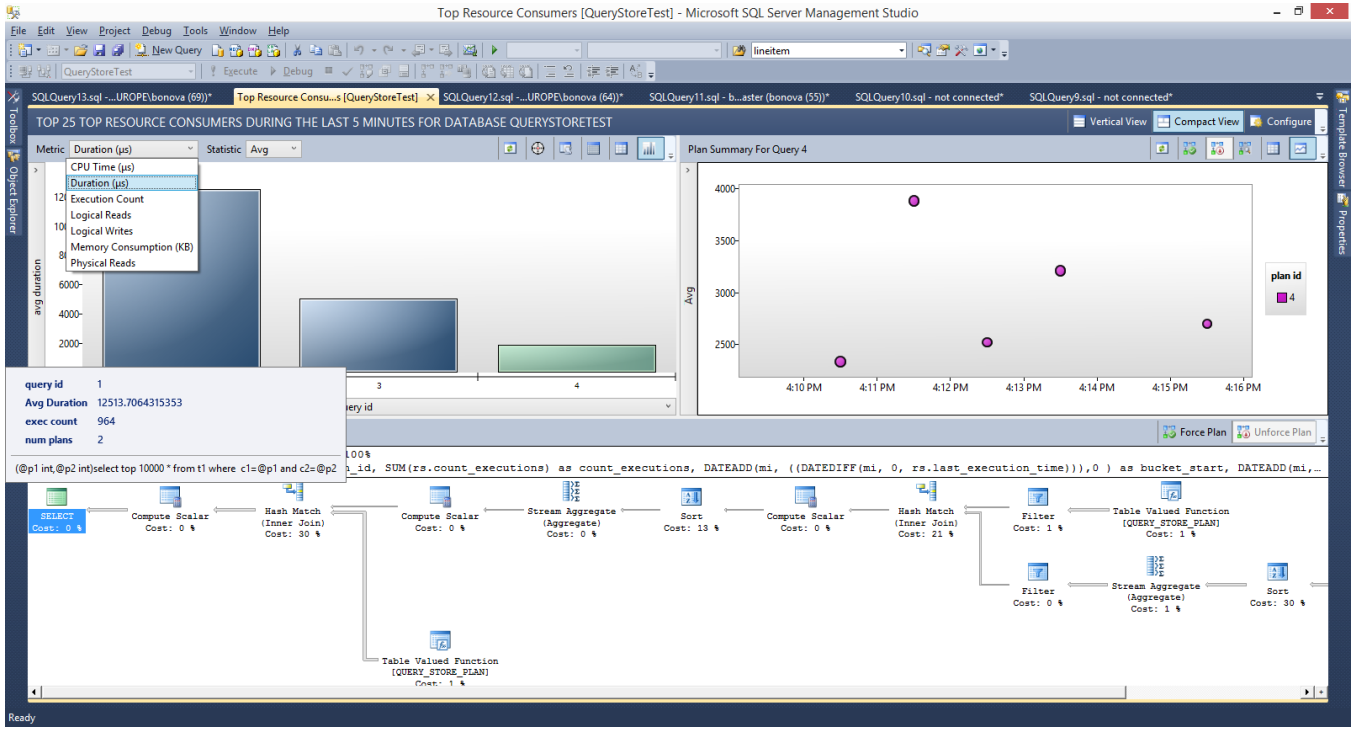

- Créer ses propres sessions
  - Moyen simple de détecter des problèmes de verrouillage & performance
- Evènements importants
  - Blocked Process Report
  - Lock Timeout
  - Lock waits
  - RPC Completed
  - Batch Completed

```
Invoke-DbaQuery -SqlInstance $Server -Database "master" -Query "
    CREATE EVENT SESSION [PerformanceIssues] ON SERVER
    ADD EVENT sqlserver.blocked_process_report(
        ACTION(sqlserver.client_app_name,sqlserver.client_hostname,sqlserver.database_id,sqlserver
    ADD EVENT sqlserver.lock_timeout_greater_than_0(SET collect_database_name=(0)
        ACTION(sqlserver.client_app_name,sqlserver.client_hostname,sqlserver.database_id,sqlserver
    ADD EVENT sqlserver.locks_lock_waits(
        ACTION(sqlserver.client_app_name,sqlserver.client_hostname,sqlserver.database_id,sqlserver
        WHERE ([increment]>=(1000) AND [count]<=(100))),
    ADD EVENT sqlserver.rpc_completed(SET collect_statement=(1)
        ACTION(sqlserver.client_app_name,sqlserver.client_hostname,sqlserver.database_id,sqlserver
        WHERE ([package0].[greater_than_equal_uint64]([duration],(250000)))),
    ADD EVENT sqlserver.sql_batch_completed(
        ACTION(sqlserver.client_app_name,sqlserver.client_hostname,sqlserver.database_id,sqlserver
        WHERE ([package0].[greater_than_equal_uint64]([duration],(250000)))),
    ADD EVENT sqlserver.xml_deadlock_report(
        ACTION(sqlserver.client_app_name,sqlserver.client_hostname,sqlserver.database_id,sqlserver
    ADD TARGET package0.event_file(SET filename=N'PerformanceIssues',max_file_size=(50),max_rollov
    WITH (MAX_MEMORY=4096 KB,EVENT_RETENTION_MODE=ALLOW_SINGLE_EVENT_LOSS,MAX_DISPATCH_LATENCY=30
"
Start-DbaXESession -SqlInstance $Server -Session "PerformanceIssues"
```

# Query Store

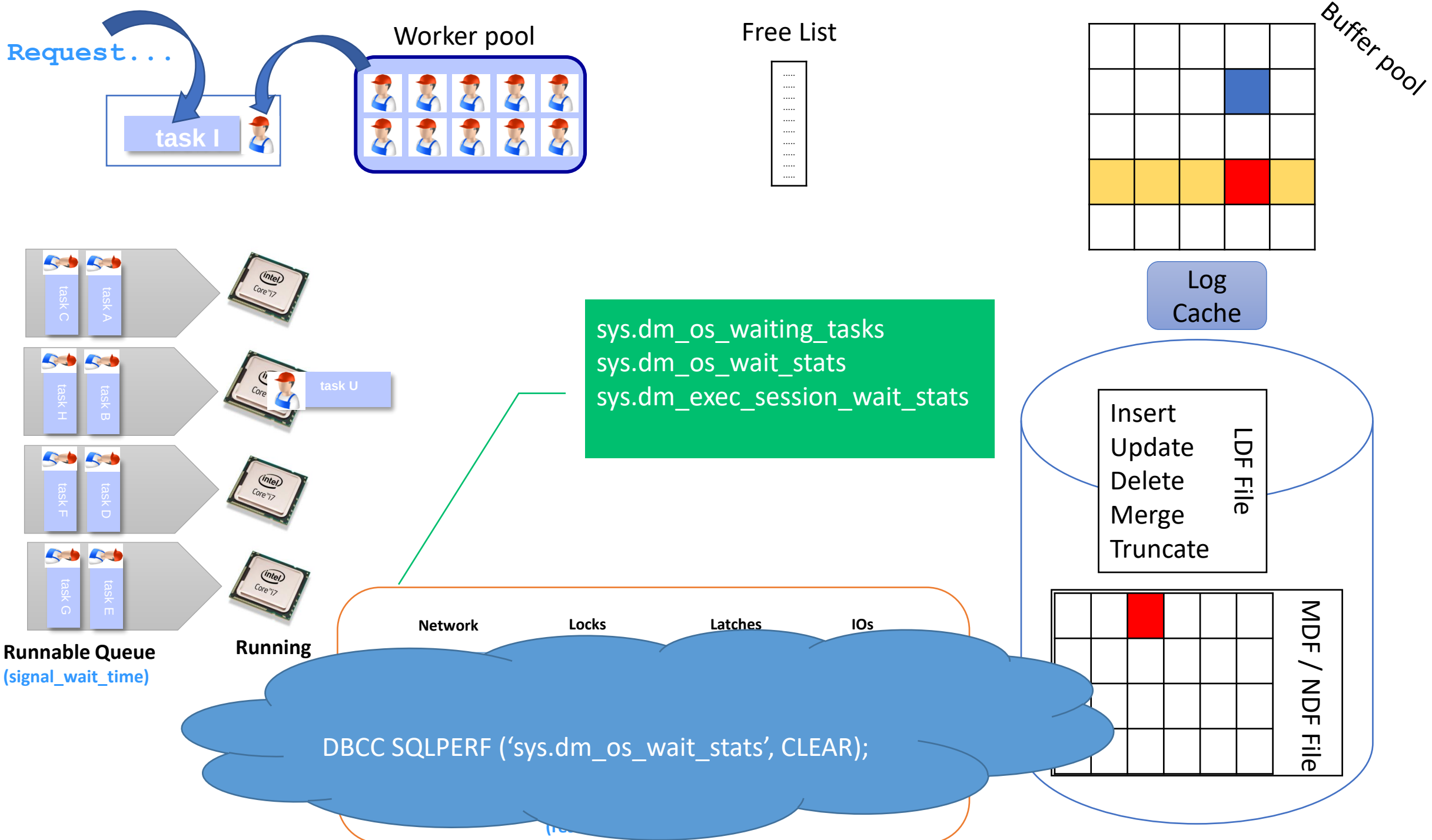
- SQL Server 2016+
- Enregistre et compare les plans d’exécution
- Spécifique à chaque base



# Wait and Queues

Technique pragmatique pour la détection de goulots d'étranglement

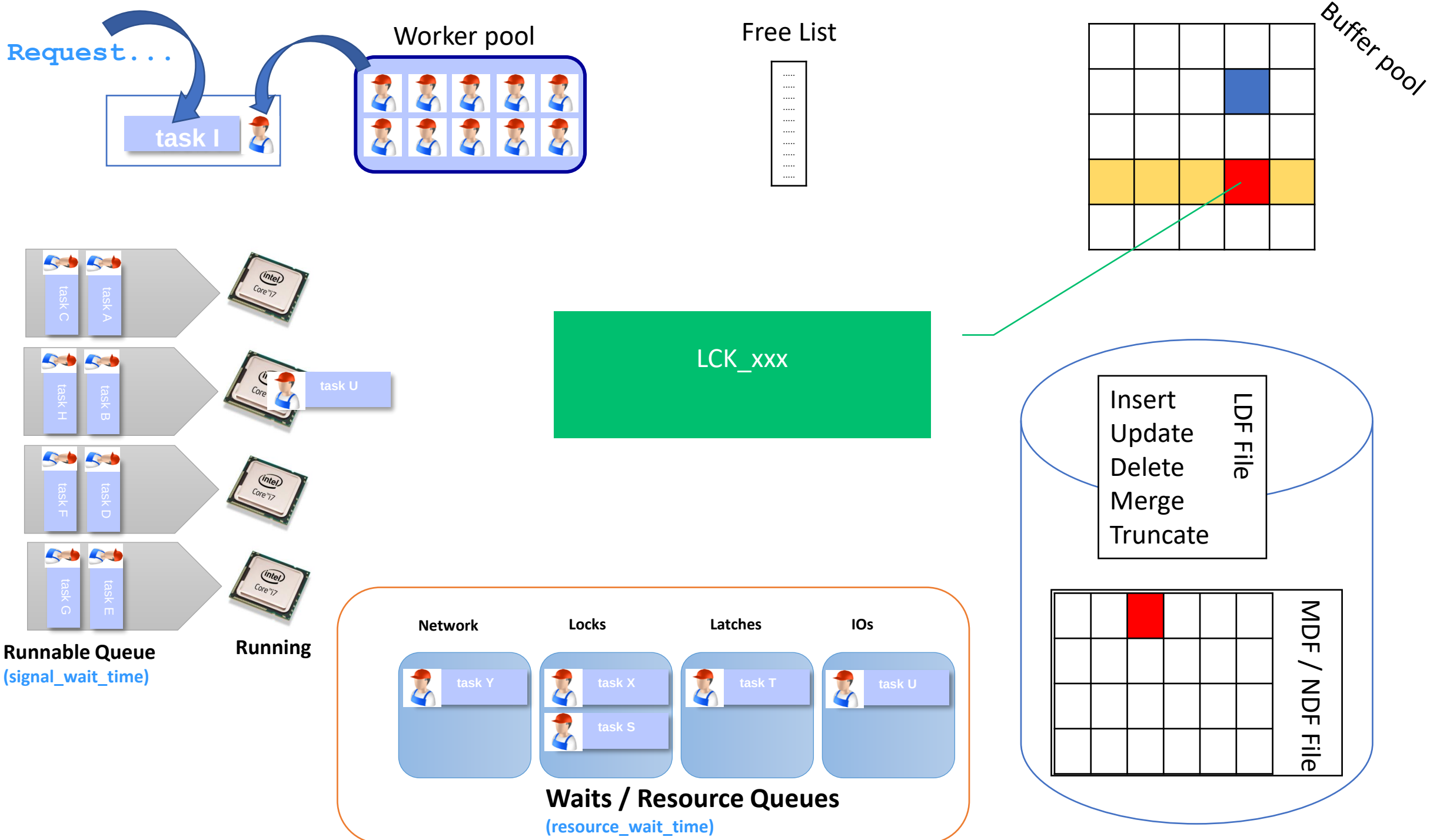

Request...
task I
Worker pool
Free List
Buffer pool
Log Cache
task C
task A
task H
task B
task U
task F
task D
task G
task E
Runnable Queue
(signal_wait_time)
Running
sys.dm_os_waiting_tasks
sys.dm_os_wait_stats
sys.dm_exec_session_wait_stats
Network
Locks
Latches
IOs
DBCC SQLPERF ('sys.dm_os_wait_stats', CLEAR);
Insert
Update
Delete
Merge
Truncate
LDF File
MDF / NDF File

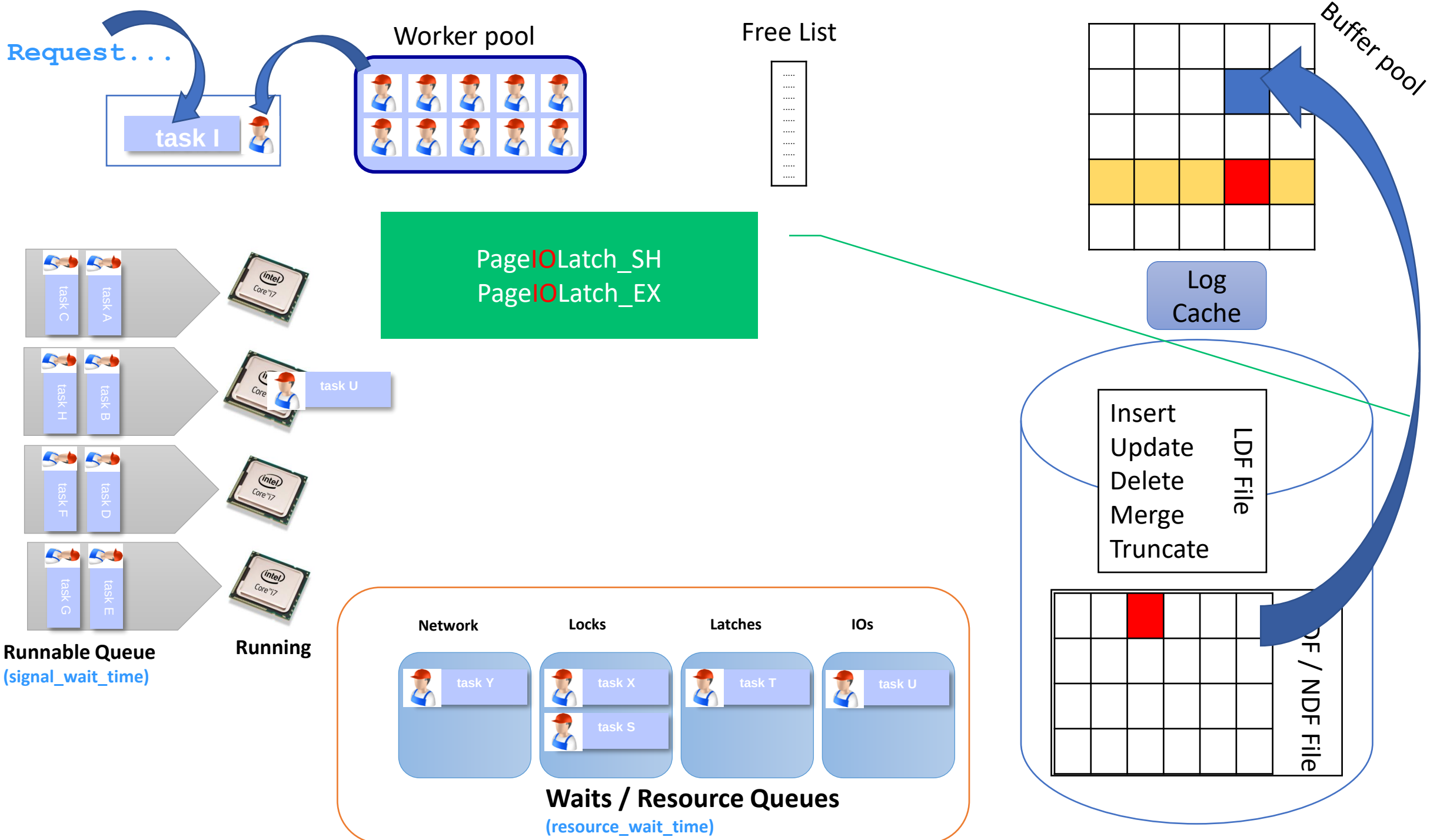
# Attentes les plus fréquentes

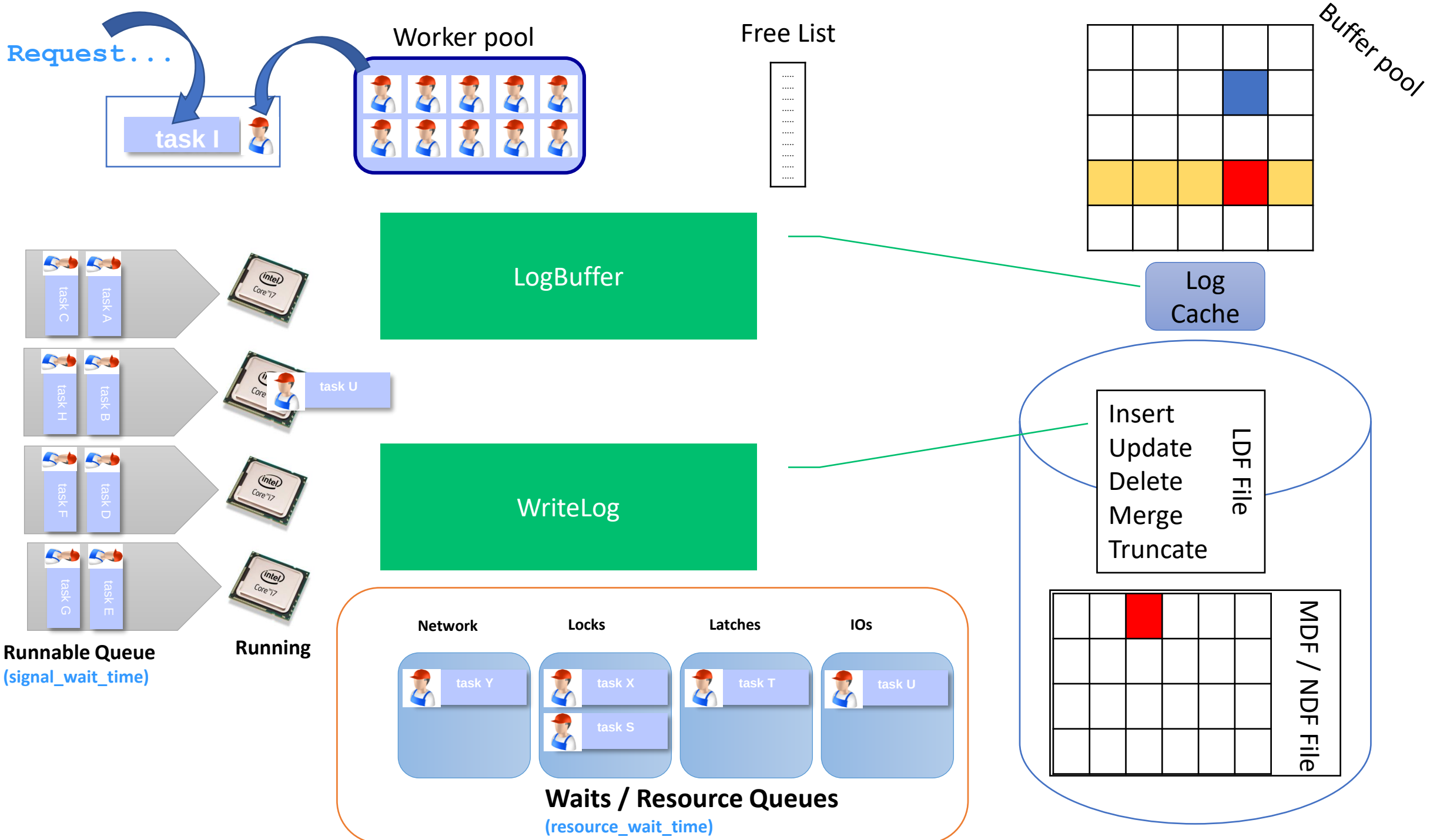

| WaitType | Wait_Sec | Resource_Sec | Signal_Sec | Wait Count | Wait Percentage |
|---|---|---|---|---|---|
| WRITELOG | 10780538.49 | 10429545.21 | 350993.29 | 645164078 | 75.36 |
| PREEMPTIVE_XE_DISPATCHER | 1759175.48 | 1759175.48 | 0.00 | 3 | 12.30 |
| SOS_SCHEDULER_YIELD | 400567.84 | 301.54 | 400266.30 | 236500272 | 2.80 |
| LCK_M_S | 296141.00 | 295495.85 | 645.15 | 710421 | 2.07 |
| ASYNC_NETWORK_IO | 248598.86 | 248457.49 | 141.37 | 215088 | 1.74 |
| CXPACKET | 215274.82 | 197232.30 | 18042.52 | 12754864 | 1.50 |
| PAGELATCH_EX | 91874.66 | 8171.45 | 83703.21 | 244349126 | 0.64 |
| PAGEIOLATCH_SH | 70499.86 | 70160.77 | 339.09 | 2218419 | 0.49 |
| PAGEIOLATCH_EX | 59815.51 | 59679.20 | 136.31 | 452519 | 0.42 |
| PAGELATCH_SH | 56729.54 | 55900.11 | 829.43 | 4624950 | 0.40 |

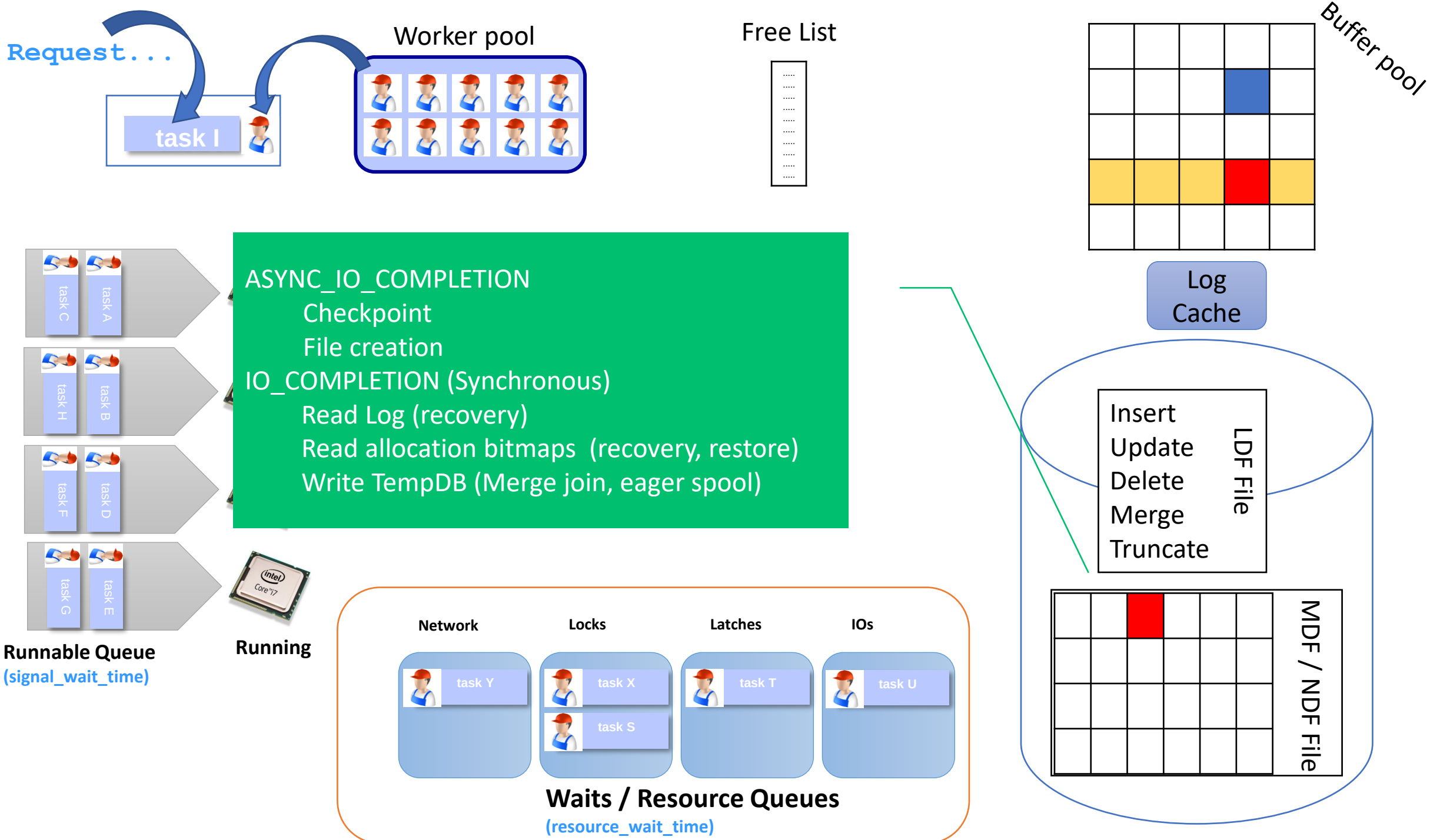
| WaitType | Wait_Sec | Resource_Sec | Signal_Sec | Wait Count | Wait Percentage |
|---|---|---|---|---|---|
| OLEDB | 42284.37 | 42284.37 | 0.00 | 2243385801 | 32.16 |
| ASYNC_IO_COMPLETION | 14996.42 | 14996.02 | 0.41 | 1730 | 11.41 |
| BACKUPBUFFER | 14563.56 | 14010.49 | 553.07 | 11753930 | 11.08 |
| BACKUPIO | 13811.04 | 13768.30 | 42.74 | 4242444 | 10.50 |
| PAGEIOLATCH_SH | 11954.69 | 11864.22 | 90.46 | 19529888 | 9.09 |
| ASYNC_NETWORK_IO | 8717.07 | 7564.87 | 1152.20 | 26038151 | 6.63 |
| WRITELOG | 7599.80 | 7306.75 | 293.04 | 7155355 | 5.78 |
| PAGEIOLATCH_EX | 3380.40 | 3343.49 | 36.91 | 6031488 | 2.57 |
| SOS_SCHEDULER_YIELD | 2983.83 | 53.80 | 2930.03 | 42931635 | 2.27 |
| IO_COMPLETION | 2434.85 | 2378.08 | 56.77 | 9309193 | 1.85 |
| LCK_M_S | 1405.47 | 1405.10 | 0.38 | 4145 | 1.07 |
| CXPACKET | 1267.15 | 1061.68 | 205.47 | 776636 | 0.96 |

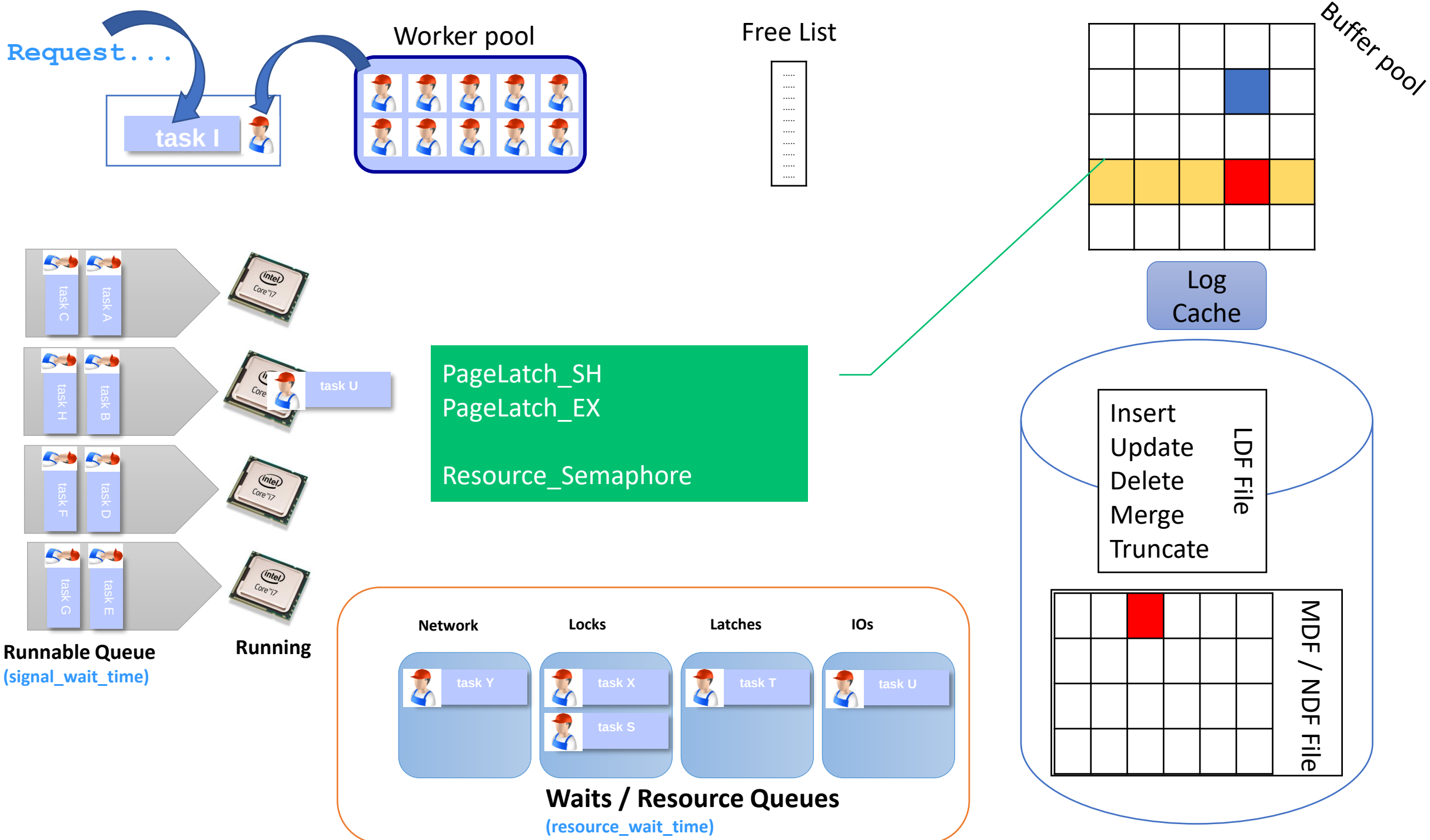
| session_id | wait_type | waiting_tasks_count | wait_time_ms | max_wait_time_ms | signal_wait_time_ms |
|---|---|---|---|---|---|
| 304 | TRACEWRITE | 3192 | 74838 | 1206 | 270 |
| 60 | WRITELOG | 43005 | 29223 | 393 | 19987 |
| 67 | LCK_M_S | 37 | 13283 | 4103 | 3 |
| 228 | SOS_SCHEDULER_YIELD | 569 | 1096 | 9 | 1095 |
| 236 | SOS_SCHEDULER_YIELD | 139 | 871 | 17 | 871 |
| 356 | SOS_SCHEDULER_YIELD | 146 | 528 | 9 | 528 |
| 444 | ASYNC_NETWORK_IO | 13 | 376 | 135 | 8 |
| 175 | SOS_SCHEDULER_YIELD | 175 | 368 | 13 | 368 |
| 60 | MEMORY_ALLOCATION_EXT | 281356 | 348 | 0 | 0 |
| 239 | BACKUPTHREAD | 20 | 326 | 297 | 1 |
| 430 | SOS_SCHEDULER_YIELD | 169 | 277 | 15 | 277 |
| 189 | MSQL_XP | 99 | 190 | 48 | 0 |
| 427 | SOS_SCHEDULER_YIELD | 164 | 183 | 8 | 183 |
| 228 | CXPACKET | 92 | 182 | 6 | 31 |
| 225 | SOS_SCHEDULER_YIELD | 185 | 170 | 4 | 170 |
| 228 | ASYNC_NETWORK_IO | 14 | 166 | 68 | 5 |
| 172 | SOS_SCHEDULER_YIELD | 177 | 161 | 11 | 161 |
| 228 | PAGELATCH_EX | 140 | 134 | 4 | 133 |
| 356 | PAGEIOLATCH_SH | 58 | 134 | 8 | 0 |
| 222 | CXPACKET | 8 | 115 | 57 | 0 |
| 87 | SLEEP_TASK | 12 | 115 | 15 | 0 |
| 228 | SLEEP_TASK | 48 | 96 | 8 | 96 |
| 344 | SOS_SCHEDULER_YIELD | 26 | 72 | 8 | 72 |
| 228 | PAGEIOLATCH_SH | 67 | 67 | 16 | 0 |
| 207 | SOS_SCHEDULER_YIELD | 190 | 67 | 1 | 67 |
| 218 | SOS_SCHEDULER_YIELD | 169 | 66 | 5 | 66 |
| 239 | PREEMPTIVE_OS_CREATEFILE | 11 | 65 | 11 | 0 |
| 222 | ASYNC_NETWORK_IO | 2 | 62 | 62 | 0 |
| 85 | SOS_SCHEDULER_YIELD | 198 | 62 | 4 | 62 |
| 380 | PAGEIOLATCH_SH | 21 | 58 | 8 | 0 |
| 130 | SOS_SCHEDULER_YIELD | 14 | 55 | 7 | 55 |
| 380 | SOS_SCHEDULER_YIELD | 120 | 49 | 4 | 49 |
| 246 | SOS_SCHEDULER_YIELD | 17 | 41 | 4 | 41 |
| 239 | SOS_SCHEDULER_YIELD | 191 | 41 | 4 | 40 |
| 238 | SOS_SCHEDULER_YIELD | 64 | 37 | 4 | 37 |
| 60 | SOS_SCHEDULER_YIELD | 57 | 33 | 4 | 33 |
| 228 | PAGELATCH_SH | 42 | 32 | 4 | 32 |
| 236 | IO_COMPLETION | 11 | 28 | 4 | 12 |
| 67 | PAGEIOLATCH_SH | 39 | 26 | 4 | 1 |
| 67 | WRITELOG | 56 | 25 | 4 | 11 |


Request...
task I
Worker pool
Free List
Buffer pool
task C
task A
task H
task B
task F
task D
task G
task E
task U
Runnable Queue
(signal_wait_time)
Running
LCK_xxx
Insert
Update
Delete
Merge
Truncate
LDF File
MDF / NDF File
Network
Locks
Latches
IOs
task Y
task X
task S
task T
task U
Waits / Resource Queues
(resource_wait_time)

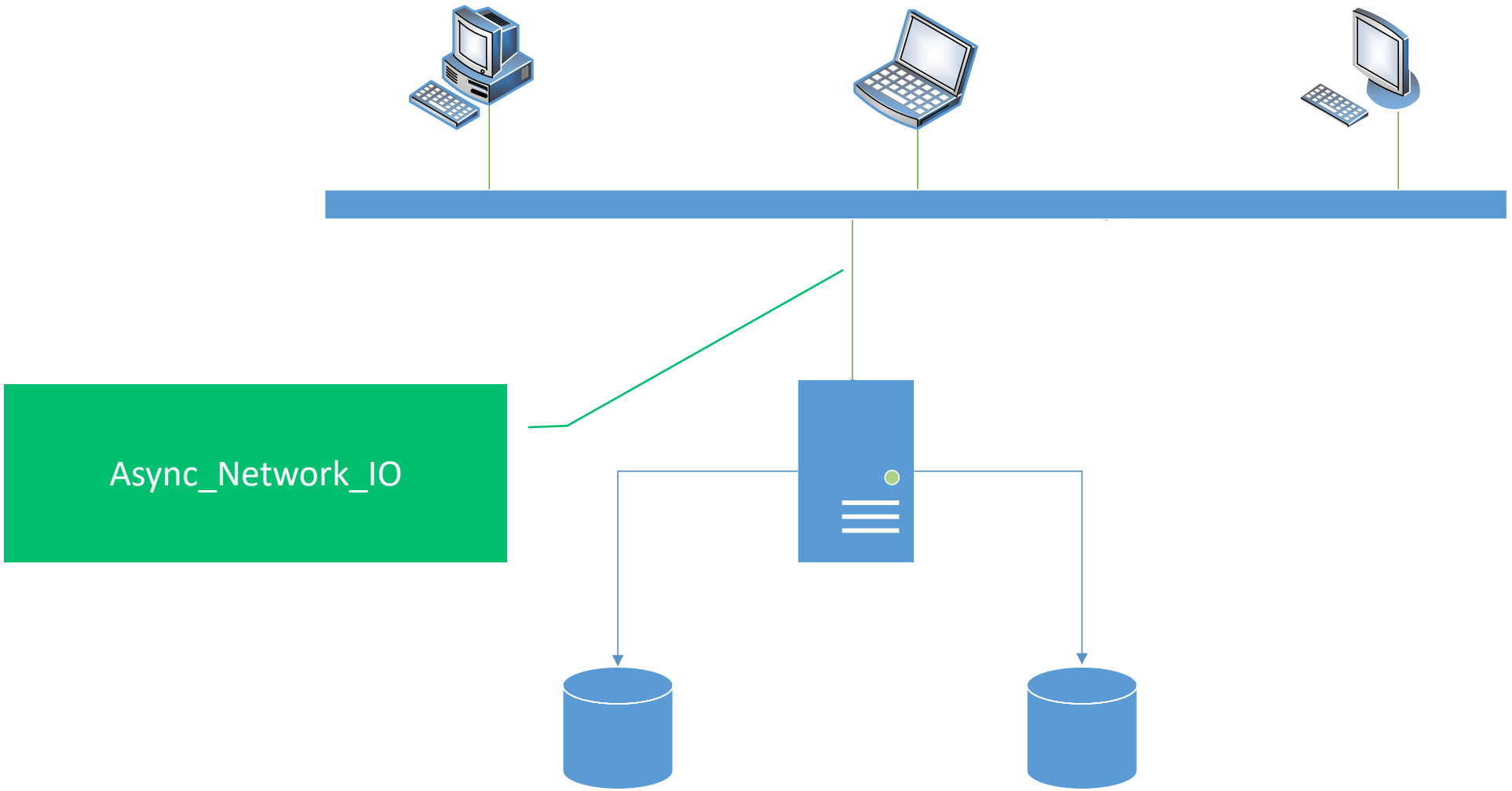

Request...
task I
Worker pool
Free List
Buffer pool
PageIOLatch_SH
PageIOLatch_EX
Log Cache
task C
task A
task H
task B
task U
task F
task D
task G
task E
Runnable Queue
(signal_wait_time)
Running
Insert
Update
Delete
Merge
Truncate
LDF File
DF / NDF File
Network
Locks
Latches
IOs
task Y
task X
task S
task T
task U
Waits / Resource Queues
(resource_wait_time)

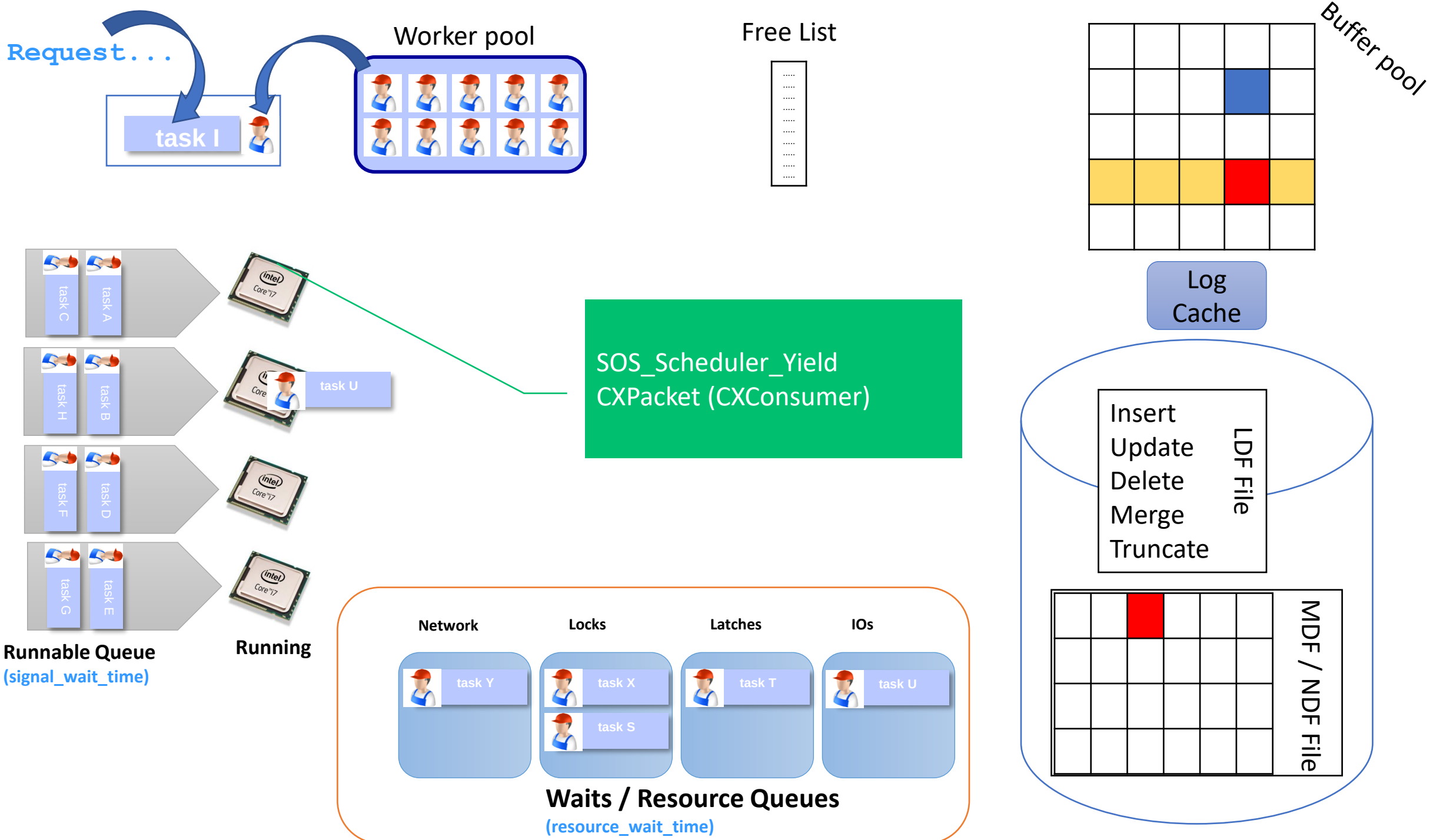

Request...
task I
Worker pool
Free List
Buffer pool
LogBuffer
Log Cache
WriteLog
task C
task A
task H
task B
task U
task F
task D
task G
task E
Runnable Queue
(signal_wait_time)
Running
Insert
Update
Delete
Merge
Truncate
LDF File
MDF / NDF File
Network
Locks
Latches
IOs
task Y
task X
task S
task T
task U
Waits / Resource Queues
(resource_wait_time)


Request...
task I
Worker pool
Free List
Buffer pool
task C
task A
task H
task B
task F
task D
task G
task E
ASYNC_IO_COMPLETION
Checkpoint
File creation
IO_COMPLETION (Synchronous)
Read Log (recovery)
Read allocation bitmaps (recovery, restore)
Write TempDB (Merge join, eager spool)
Log Cache
Insert
Update
Delete
Merge
Truncate
LDF File
MDF / NDF File
Runnable Queue
(signal_wait_time)
Running
Network
Locks
Latches
IOs
task Y
task X
task S
task T
task U
Waits / Resource Queues
(resource_wait_time)


Request...
task I
Worker pool
Free List
Buffer pool
Log Cache
task C
task A
task H
task B
task U
task F
task D
task G
task E
Runnable Queue
(signal_wait_time)
Running
PageLatch_SH
PageLatch_EX
Resource_Semaphore
Insert
Update
Delete
Merge
Truncate
LDF File
MDF / NDF File
Network
Locks
Latches
IOs
task Y
task X
task S
task T
task U
Waits / Resource Queues
(resource_wait_time)

# Async_network_io



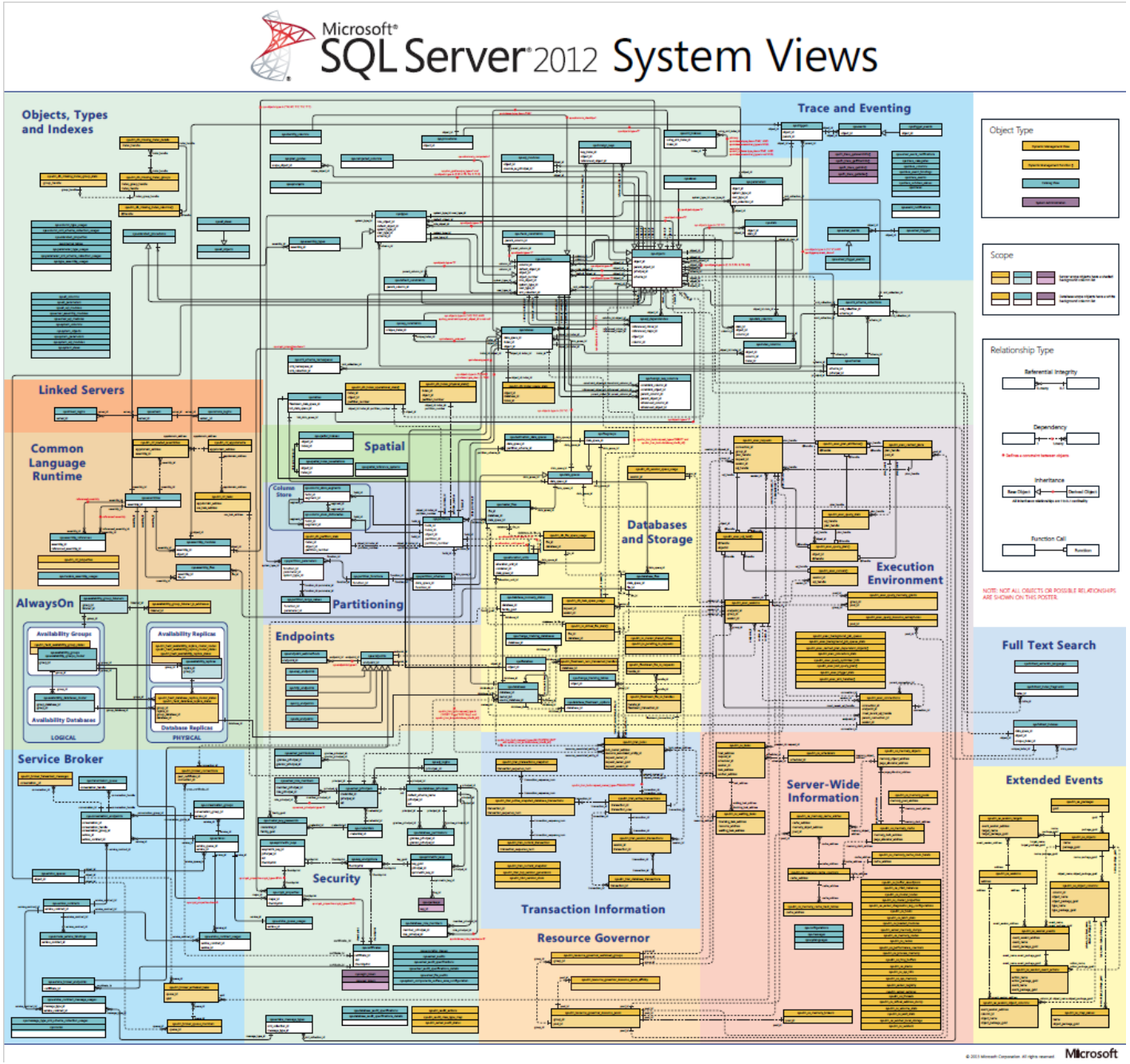

Request...
task I
Worker pool
Free List
Buffer pool
Log Cache
task C
task A
task H
task B
task U
task F
task D
task G
task E
Runnable Queue
(signal_wait_time)
Running
SOS_Scheduler_Yield
CXPacket (CXConsumer)
Insert
Update
Delete
Merge
Truncate
LDF File
MDF / NDF File
Network
Locks
Latches
IOs
task Y
task X
task S
task T
task U
Waits / Resource Queues
(resource_wait_time)

# “Technique” wait and queues

- Approche pragmatique
  - Itération sur les problèmes les plus importants
- Bibliographie (entre autres)
  - SQL Server 2005 Waits and Queues (Tom Davidson)
  - Performance Tuning Using Wait Statistics (Jonathan Kehayias, Erin Stellato)
  - Performance Tuning With Wait Statistics (Joe Sack)
- Description des attentes
  - https://www.sqlskills.com/help/waits/

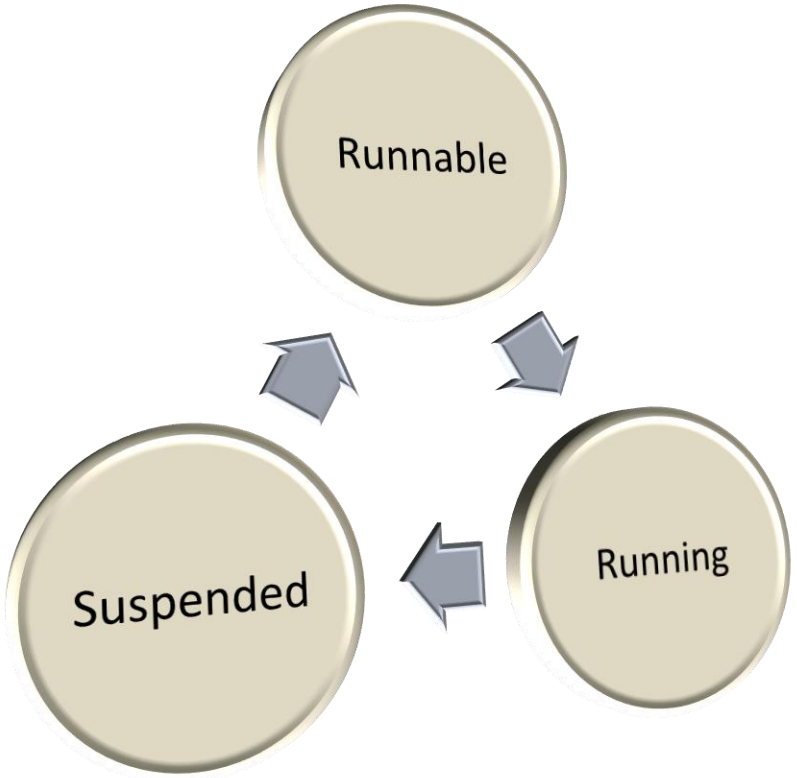
# DMVs et DMFs

Un œil sur l'activité



# Rappel - Etats d’une requête

- Running
  - Requête soumise
  - Compilation <-  -> tri des données
- Suspended
  - Requête non active
  - Attend une ressource
- Runnable
  - SPID dans la file d’attente
  - Attente d’un quantum de temps
- Pending
  - Pas de worker thread disponible
- Background
- Sleeping
  - Plus de travail a effectuer



# Activité en temps réel


- sys.dm_exec_sessions
- sys.dm_exec_connexions
- sys.dm_exec_requests
- sys.dm_tran_locks
- sp_whoisactive (@AdamMachanic)

# Activité historique


- sys.dm_exec_query_stats
- sys.dm_exec_procedure_stats
- sys.dm_exec_trigger_stats
- sys.dm_exec_function_stats

sys.dm_exec_sql_text
sys.dm_exec_query_plan

# Les index …


- sys.dm_db_missing_index_*
  - Ne pas prendre au pied de la lettre !
  - Mais de bonnes indications …
- sys.dm_db_index_physical_stats
  - Fragmentation, nombre de pages
  - Taille des enregistrements
- sys.dm_db_index_usage_stats
  - Lectures vs écritures
  - Index non utilisés
- sys.dm_db_index_operational_stats
  - Locks, latches
  - Forwarded fetch

| object_id | index_id | user_scans | user_seeks | user_lookups | user_read | user_updates |
|---|---|---|---|---|---|---|
| 1840373971 | 17 | 0 | 0 | 0 | 0 | 812,453,385 |
| 1840373971 | 18 | 0 | 0 | 0 | 0 | 62,707,474 |
| 1004894997 | 12 | 0 | 0 | 0 | 0 | 28,006,915 |
| 1840373971 | 110 | 0 | 0 | 0 | 0 | 21,069,789 |
| 1105035318 | 10 | 0 | 0 | 0 | 0 | 15,059,636 |
| 1105035318 | 7 | 0 | 0 | 0 | 0 | 15,059,636 |
| 1105035318 | 28 | 0 | 0 | 0 | 0 | 15,059,636 |
| 1105035318 | 8 | 0 | 0 | 0 | 0 | 14,653,046 |
| 1105035318 | 41 | 0 | 0 | 0 | 0 | 14,570,448 |
| 1105035318 | 42 | 0 | 0 | 0 | 0 | 14,570,402 |
| 1105035318 | 2 | 0 | 0 | 0 | 0 | 14,570,402 |

| object_id | index_id | row_lock_count | row_lock_wait_count | row_lock_wait_in_ms |
|---|---|---|---|---|
| 450412974 | 1 | 122,930,566 | 632,877 | 10,041,545 |
| 450412974 | 3 | 721,703 | 10,676 | 173,883 |
| 1013578649 | 1 | 14,495,641 | 216 | 103,346 |
| 32367530 | 1 | 16,108 | 728 | 84,668 |
| 812894313 | 3 | 28,610,039 | 220 | 18,689 |

| object_id | index_id | row_lock_count | row_lock_wait_count | row_lock_wait_in_ms |
|---|---|---|---|---|
| 904116199 | 1 | 153,460 | 12 | 175,063,393 |
| 1897213959 | 2 | 13,332,283 | 13 | 99,635,110 |
| 1513342193 | 3 | 9,533,727 | 2 | 33,459,970 |
| 1902407784 | 2 | 4,388,757,801 | 202 | 10,151,833 |
| 520857418 | 6 | 1,672 | 1 | 2,001,204 |
| 639809837 | 4 | 57,610,644 | 223 | 1,397,563 |

| object_id | index_id | leaf_insert_count | leaf_delete_count | leaf_update_count | leaf_ghost_count | forwarded_fetch_count |
|---|---|---|---|---|---|---|
| 645577338 | 0 | 226 174 | 231 897 | 4 091 359 | 0 | 405 325 |
| 677577452 | 0 | 1 771 | 0 | 0 | 0 | 54 963 |

# DMVs et DMFs

- Activité temps réel
- Historique lié à l'up time du service
  - historiser les données ?
- Enormément d'informations renvoyées
  - interprétation des chiffres !
- Scripts Glenn Berry (@GlennAlanBerry)
  - DMV Queries Archives - Glenn's SQL Server Performance (glennsqlperformance.com)

# Conclusion

Q & A

Problèmes les plus courants :

- IO disque (latence)
- Verrouillage excessif
- Statistiques hors d'âge
- Gestion du parallélisme
- Application client (RBAR, transactions longues …)

Importance d’une bonne configuration

- Host / VM (power option, type de CPU & disque)
- Windows (disques, antivirus, power option)
- SQL (rétention, purge des historiques)
- Plan de maintenance (Gestion des index / statistiques d’index)