

Module 1

Architecture, scheduling, and wait statistics

Copyright Tibor Karaszi Consulting and Cornerstone Group AB

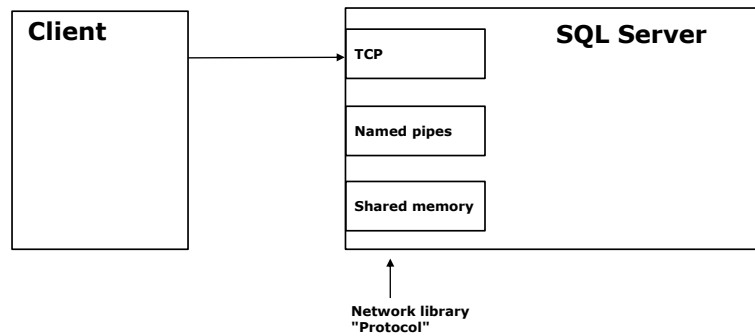
Module Overview

- SQL Server components and SQLOS
- SQL Server scheduling
- Wait statistics

- Connection protocols
- Database engine
- SQLOS
- Multiple CPUs—SMP and NUMA
- The query lifecycle
- Sources for (performance) information and monitoring

Connection protocols

- Network protocols, also known as Network Libraries (Netlib)
 - Shared memory
 - Named Pipes
 - TCP/IP
- Packets submitted through the Netlibs uses the Tabular Data Stream (TDS) protocol



Network Protocols and Network Libraries

<https://docs.microsoft.com/en-us/sql/sql-server/install/network-protocols-and-network-libraries>

Relational Engine

- Relational engine
 - Parses and optimizes the queries
 - Manages the caching and execution of execution plans

Storage Engine

- Storage engine
 - Manages buffer pages
 - I/O to the physical files
 - Transactions
 - Locking

SQLOS (PAL)

- SQLOS
 - Abstraction layer over common operating system functions, providing task and memory management
 - Also known as Platform Abstraction Layer

SQL Server Architecture Explained: Named Pipes, Optimizer, Buffer Manager
<https://www.guru99.com/sql-server-architecture.html>

SQLOS

- SQL Server requirements for low-level resource (memory, schedulers, synchronization objects, and so on) are complex
- Many services inside the engine need access to these low-level resources
- Components to provide this access are grouped together into a single functional unit called the SQLOS
- SQL Server components make calls to the SQLOS
- SQLOS provides many functions, including memory management, scheduling, I/O management, locking framework, transaction management, deadlock detection, and exception handling

SQL SERVER – What is SQL Server Operating System?

<https://blog.sqlauthority.com/2015/11/11/sql-server-what-is-sql-server-operating-system/>

The query lifecycle

- SQL Server Network Interface
 - Receive query from client
- Relational engine
 - Parse query and bind to database objects
 - Compile query plan
 - Execute query plan
 - Generate results
- Storage Engine
 - Used by relational engine to access data and metadata
- SQLOS
 - Provides low-level functions to storage and relational engines

Query Processing Architecture Guide

<https://docs.microsoft.com/en-us/sql/relational-databases/query-processing-architecture-guide>

Sources for (performance) information and monitoring

- Dynamic Management Views, DMVs
 - Will be used throughout the course
 - Examples
 - sys.dm_exec_sessions
 - sys.dm_os_schedulers
 - sys.dm_tran_locks
 - sys.dm_io_virtual_file_stats
 - sys.dm_db_index_physical_stats
- Extended Events
 - Discussed in a later module
- Performance Monitor
 - Discussed in a later module
- Activity Monitor

System Dynamic Management Views

<https://docs.microsoft.com/en-us/sql/relational-databases/system-dynamic-management-views/system-dynamic-management-views>

Glenn Berry's SQL Server Diagnostic Queries

<https://glennsqlperformance.com/resources/>

Activity Monitor

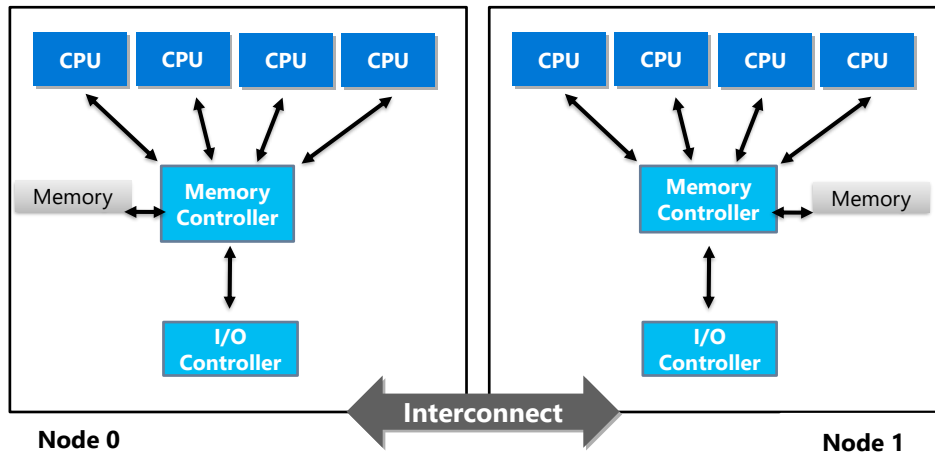
<https://docs.microsoft.com/en-us/sql/relational-databases/performance-monitor/activity-monitor>

Explaining Activity Monitor

<https://sqlblog.karaszi.com/explaining-activity-monitor/>

Lesson: Windows scheduling vs. SQL Server scheduling

- Preemptive vs. non-preemptive scheduling
- SQL Server on OS Scheduler and User Mode Scheduler
- SQL Server execution model
- User request lifecycle



Logical diagram of a two-node NUMA system

Demo Parallel

Parallelism in SQL Server Execution Plan

<https://www.mssqltips.com/sqlservertip/5404/parallelism-in-sql-server-execution-plan/>

CPU related configuration settings

- Processor affinity mask
 - Which CPU cores can we use (sets a schedule online or offline)
 - Most frequently set to 0, meaning all CPUs
- Cost threshold for parallelism
 - The breaking point for when SQL Server will possibly let the query go parallel.
 - Based on "hypothetical seconds" (seconds on a machine from ca 1998)
 - The default 5 is generally considered a way to low value
 - Increase to for instance 50 or 100
 - Total cost for query execution is higher when going parallel because of synchronization of the threads
 - For instance on a laptop from 2020, 5 corresponds to 0.082 seconds.
- Max degree of parallelism
 - Can be set at server, database and query level
 - Recommended config:
 - Same as number of cores up to 8.
 - If more than 1 numa node:
 - If up to 16 logical CPU per node then same as logical CPUs per node
 - If more then 16 logical CPUs per node then half the number of CPUs per node

Configure the max degree of parallelism Server Configuration Option

<https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/configure-the-max-degree-of-parallelism-server-configuration-option>

Understanding and Using Parallelism in SQL Server

<https://www.red-gate.com/simple-talk/databases/sql-server/learn/understanding-and-using-parallelism-in-sql-server/>

Configure the cost threshold for parallelism Server Configuration Option

<https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/configure-the-cost-threshold-for-parallelism-server-configuration-option>

Preemptive vs. non-preemptive scheduling

- Each CPU core can run one task at a time
- Preemptive scheduling
 - Is driven by the view of prioritized computation
 - Means that a low-priority process is pre-empted out of the processor by a high-priority process
- Non-preemptive scheduling
 - Means that the priority of a process doesn't matter
 - Is where a process is not preempted, and executes until it explicitly yields the processor
- Windows uses preemptive scheduling
- SQL Server uses non-preemptive scheduling

Understanding SQL Server Schedulers, Workers and Tasks

<https://www.mssqltips.com/sqlservertip/4403/understanding-sql-server-schedulers-workers-and-tasks/>

CPU Scheduling Basics – Windows and SQL Server

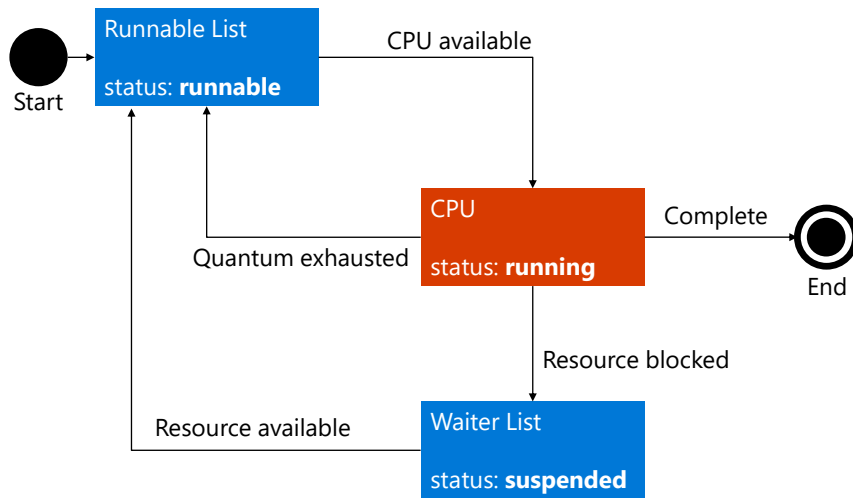
<https://www.nocentino.com/posts/2016-05-26-cpu-scheduling-basics-windows-and-sql-server/>

SQL SERVER : SQL Server Scheduler

<https://www.sqlservercentral.com/blogs/sql-server-sql-server-scheduler>

- Different names in different versions of SQL Server:
 - UMS in SQL Server 7.0 and 2000
 - SOS in SQL Server 2005 and later
- Enables non-preemptive scheduling
- Maintains five lists to facilitate scheduling:
 - Worker list
 - Runnable list
 - Waiter list
 - I/O list
 - Timer list

SQL Server execution model



SQL SERVER – Life of a SQL Query – Query States

<https://blog.sqlauthority.com/2014/04/17/sql-server-life-of-a-sql-query-query-states/>

SQL server DB status runnable, sleeping, suspended, running, pending and background meanings

<http://sqlserversavari.blogspot.com/2019/10/sql-server-db-status-runnable-sleeping.html>

User Request Life Cycle

- Connection established: **sys.dm_exec_connections**
- Session ID assigned: **sys.dm_exec_sessions**
- Request created: **sys.dm_exec_requests**
- Task (s) created: **sys.dm_os_tasks**
- Task assigned to worker: **sys.dm_os_workers**
- Worker runs on operating system thread: **sys.dm_os_threads**
- Scheduler manages task activity: **sys.dm_os_schedulers**

Demo Monitoring scheduling

Thread and Task Architecture Guide

<https://docs.microsoft.com/en-us/sql/relational-databases/thread-and-task-architecture-guide>

Lesson: Waits and queues

- Overview of waits and queues
- Viewing wait statistics
- LCK_M_* wait types
- PAGELATCH_* wait types
- PAGEIOLATCH_* wait types
- CXPACKET and CXCONSUMER wait types
- WRITELOG wait type
- Other common wait types

Overview of waits and queues

- Waits
 - A request or task is said to “wait” if a required resource is not available
 - SQL Server tracks the resources being waited for as wait type
 - Resource waits—**Suspended** tasks wait for data pages
 - Signal waits—**Runnable** tasks wait for CPU time
- Queues
 - System resources and utilization, which are measured using Performance Monitor performance counters, DMVs, and other tools

SQL Server 2005 Waits and Queues – Microsoft

https://download.microsoft.com/download/4/7/a/47a548b9-249e-484c-abd7-29f31282b04d/Performance_Tuning_Waits_Queues.doc

In Review: SQL Server 2005 Waits and Queues

<https://www.brentozar.com/archive/2015/10/in-review-sql-server-2005-waits-and-queues-2/>

Viewing wait statistics

- When a thread waits on a resource, SQL Server tracks wait information
 - Aggregated wait statistics for all wait types since startup of instance
 - `sys.dm_os_wait_stats`
 - Current wait statistics
 - `sys.dm_os_waiting_tasks`
 - Aggregated wait statistics for active sessions
 - `sys.dm_exec_session_wait_stats`
 - Introduced in 2016
 - The actual execution plan

Demo Wait stats views

Boost SQL Server Performance with Wait Statistics

<https://www.sqlshack.com/boost-sql-server-performance-with-wait-statistics/>

SQL Server Wait Statistics (or please tell me where it hurts...)

<https://www.sqlskills.com/blogs/paul/wait-statistics-or-please-tell-me-where-it-hurts/>

`sys.dm_os_wait_stats` (Transact-SQL)

<https://docs.microsoft.com/en-us/sql/relational-databases/system-dynamic-management-views/sys-dm-os-wait-stats-transact-sql>

`sys.dm_os_waiting_tasks` (Transact-SQL)

<https://docs.microsoft.com/en-us/sql/relational-databases/system-dynamic-management-views/sys-dm-os-waiting-tasks-transact-sql>

`sys.dm_exec_session_wait_stats` (Transact-SQL)

<https://docs.microsoft.com/en-us/sql/relational-databases/system-dynamic-management-views/sys-dm-exec-session-wait-stats-transact-sql>

Troubleshooting SQL Server Wait Stats

<https://www.sentryone.com/white-papers/troubleshooting-sql-server-wait-stats>

Using Wait Stats to Find Why SQL Server is Slow

<https://www.brentozar.com/sql/wait-stats/>

Get Detailed Wait Stats with SQL Server Execution Plan

<https://www.mssqltips.com/sqlservertip/5081/get-detailed-wait-stats-with-sql-server-execution-plan/>

LCK_M_* wait types

- A thread waits on an incompatible lock
- Possible causes include:
 - A large update or table scan causing lock escalation
 - Unnecessary shared locks on data being accessed
- Possible solutions include:
 - Locking may or may not be a root cause:
 - Use **sys.dm_os_waiting_tasks** to find out the wait type that the lead blocker is waiting on
 - Find out which queries are waiting too long for locks using the blocked process report
 - Consider using a different isolation level
 - Review indexing strategy and optimize queries

SQL Server Wait Types

<https://www.sqlskills.com/help/waits/#/>

PAGELATCH_* wait types

- Occurs when a task waits to latch a page in the buffer pool
- Possible causes include:
 - Contention for “hot” pages
 - Contention for file allocation pages
- Possible solutions include:
 - Use a new indexing strategy or partitioning
 - Use fewer short-lived objects, or add data files

SQL Server Wait Types

<https://www.sqlskills.com/help/waits/#p>

PAGEIOLATCH_* wait types

- A task waits for a data page to be retrieved from disk to memory
- Possible causes include:
 - I/O subsystem is slow
 - Poor query performance
- Possible solutions include:
 - Replace I/O subsystem
 - Review and update indexes and statistics
 - Identify queries with parallel scans and implicit conversions in the query plan
 - Reduce page splits

SQL Server Wait Types

<https://www.sqlskills.com/help/waits/#p>

CXPACKET and CXCONSUMER wait types

- A task participating in a parallelized request completes before other tasks in the request
 - CXPACKET – producer is slow producing rows
 - CXCONSUMER – consumer is waiting for rows. Can generally be ignored.
- Possible causes include:
 - Bad query plan selection
 - Uneven distribution of work among tasks
 - Some parallel tasks slowed by other wait types
- Possible solutions include:
 - Update statistics
 - Resolve underlying problems
 - Reduce MAXDOP
 - Raise cost threshold for parallelism

SQL Server Wait Types

<https://www.sqlskills.com/help/waits/cxpacket/>

<https://www.sqlskills.com/help/waits/cxconsumer/>

WRITELOG wait type

- A task is waiting for a transaction log buffer to be flushed to disk
- Possible causes include:
 - Poor log disk I/O
 - Many small transactions
 - Unnecessary indexes
 - Frequent page splits
- Possible solutions include:
 - Move the log to faster storage
 - Use fewer, larger transactions
 - Review indexing

SQL Server Wait Types

<https://www.sqlskills.com/help/waits/writelog/>

Other common wait types

- **SOS_SCHEDULER_YIELD**
 - Waiting for CPU
- **THREADPOOL**
 - Waiting for worker thread
- **ASYNC_NETWORK_IO**
 - Waiting for client to consume data
- **RESOURCE_SEMAPHORE**
 - Waiting for a memory grant
- **CMEMTHREAD**
 - Add or remove execution plan to cache
 - Possible indication of plan cache bloat

SQL Server Wait Types

https://www.sqlskills.com/help/waits/sos_scheduler_yield/

<https://www.sqlskills.com/help/waits/threadpool/>

https://www.sqlskills.com/help/waits/async_network_io/

https://www.sqlskills.com/help/waits/resource_semaphore/

<https://www.sqlskills.com/help/waits/cmesthread/>

Lab 1: Architecture, scheduling, and wait statistics

- Ex 1: Document the hardware configuration
- Ex 2: Determine the major wait statistics

Estimated Time: 30 minutes