



Microsoft SQL Server®

Memory, TempDB, CPU, I/O & Cloud Performance

Surendra Panpaliya

Day 3

01

Module 5:
TempDB &
Memory Tuning

02

Module 6:
Parallelism, CPU
& I/O Tuning

03

Module 7:
Query Store, HA
& Azure SQL
Performance

Module 6: Parallelism, CPU & I/O Tuning

PART 1: WHAT IS PARALLELISM?

Parallelism

Parallelism means:

SQL Server uses **multiple CPU cores** to run **one query faster**

Real-Life Example



- Cooking for 1 person → 1 cook
- Cooking for 100 people → multiple cooks

But...

! Too many cooks in small kitchen = chaos

PART 2: MAXDOP & COST THRESHOLD

1 **MAXDOP** **(Maximum** **Degree of** **Parallelism)**



What is MAXDOP?



MAXDOP controls:



Maximum number of CPU cores one query can use

Simple Rule (CSC Standard)

Scenario	MAXDOP
OLTP systems	4 or 8
Reporting systems	Higher allowed
Mixed workload	Controlled

Real-Life Example



One car
allowed on
narrow bridge

Too many
cars → traffic
jam

LAB 1 – Check Current MAXDOP

```
SELECT value_in_use  
FROM sys.configurations  
WHERE name = 'max degree of parallelism';
```

LAB 1 – Check Current MAXDOP

Change MAXDOP (Example)

```
EXEC sys.sp_configure 'max degree of parallelism', 4;  
RECONFIGURE;
```

2

Cost Threshold for Parallelism

What is Cost Threshold?

- It decides:
- “Is this query expensive enough to go parallel?”
- Default = **5** (too low for modern systems)

Real-Life Example



Small repair → 1 worker



Big construction → many workers

CSC Recommendation

Workload	Cost Threshold
OLTP	30–50
Mixed	25–40

LAB 2 – Change Cost Threshold



```
EXEC sys.sp_configure 'cost threshold for parallelism', 40;
```



```
RECONFIGURE;
```

PART 3: CXPACKET vs CXCONSUMER

CXPACKET

Meaning

- Parallel threads are **waiting for each other**

Often caused by:

- Too much parallelism
- Skewed data

CXCONSUMER

Meaning

Coordinator thread waiting

→ This is **normal** and usually harmless

Simple Rule

Wait	Meaning
CXPACKET	Investigate
CXCONSUMER	Usually ignore

PART 4: PARALLELISM MISUSE IN OLTP

Common OLTP Problem



Small queries going parallel



Many users



CPU exhaustion

10 customers

10 cooks

But only 1 stove

Real-Life Example



Fix Strategy

✓ Increase cost threshold

✓ Reduce MAXDOP

✓ Avoid hints initially

PART 5: HIGH CPU DIAGNOSIS

What Causes High CPU?

Cause	Example
Bad execution plans	Table scans
Excessive parallelism	CXPACKET
Parameter sniffing	Wrong plans
Compilation storms	Ad-hoc SQL

LAB 3 – Identify CPU-Heavy Queries

Tool: Query Store

```
SELECT
```

```
    q.query_id,  
    rs.avg_cpu_time,  
    rs.count_executions
```

```
FROM sys.query_store_runtime_stats rs
```

```
JOIN sys.query_store_query q
```

```
ON rs.query_id = q.query_id
```

```
ORDER BY rs.avg_cpu_time DESC;
```

CSC Insight

- Fix top **2–3 queries**, CPU drops massively

PART 6: I/O & DISK LATENCY

What is I/O Latency?

Latency = **how long disk takes to respond**

Real-Life Example

Ordered item:

- Amazon Prime → fast
- Normal delivery → slow

LAB 4 – File- Level I/O Latency

Tool:

sys.dm_io_virtual_file_stats

SELECT

```
DB_NAME(vfs.database_id) AS db_name,  
mf.name,  
vfs.num_of_reads,  
vfs.io_stall_read_ms,  
(vfs.io_stall_read_ms / NULLIF(vfs.num_of_reads,0))  
AS avg_read_latency_ms  
FROM sys.dm_io_virtual_file_stats(NULL, NULL) vfs  
JOIN sys.master_files mf  
ON vfs.database_id = mf.database_id  
AND vfs.file_id = mf.file_id;
```

How CSC Reads This

Latency

< 10 ms

Meaning

Excellent

10–20 ms

Acceptable

> 20 ms

Problem

PART 7: STORAGE TIERING BEST PRACTICES

What is Storage Tiering?

Different files on different storage speeds.

CSC Best Practice

FILE TYPE

STORAGE

Data files

Fast SSD

Log files

Very fast SSD

TempDB

Fastest disk

Real-Life Example



Lift for heavy goods



Stairs for people

PART 8: OPTIMIZE QUERIES CAUSING CPU SPIKES

Common Fixes

- ✓ Add missing indexes
- ✓ Rewrite queries
- ✓ Fix parameter sniffing
- ✓ Update statistics
- ✓ Use Query Store plan forcing

LAB 5 – CPU Spike Fix

1. Identify CPU-heavy query (Query Store)
2. Check execution plan
3. Add covering index
4. Rerun and compare CPU

PART 9: CSC INCIDENT – END-TO-END WALKTHROUGH

Problem

- CPU always high
- CXPACKET waits
- Slow application

Investigation

- ✓ Checked Query Store
- ✓ Found small queries going parallel
- ✓ Checked MAXDOP & cost threshold

Fix

- ✓ Increased cost threshold to 40
- ✓ Reduced MAXDOP to 4
- ✓ Optimized top 3 queries

Result

- ✓ CPU stabilized
- ✓ Application faster
- ✓ No hardware change

Summary

Term	Simple Meaning
Parallelism	Using many CPUs
MAXDOP	Max CPUs per query
Cost Threshold	Parallel decision
CXPACKET	Parallel imbalance

Summary

Term	Simple Meaning
CXCONSUMER	Normal wait
CPU Spike	Too much work
I/O Latency	Disk slowness
Tiering	Right disk for right file



**Thank you for
your support and
patience**

Surendra Panpaliya
Founder and CEO
GKTCS Innovations
<https://www.gktcs.com>