SQL Server: Practical Troubleshooting

Who is this guy with heavy accent?

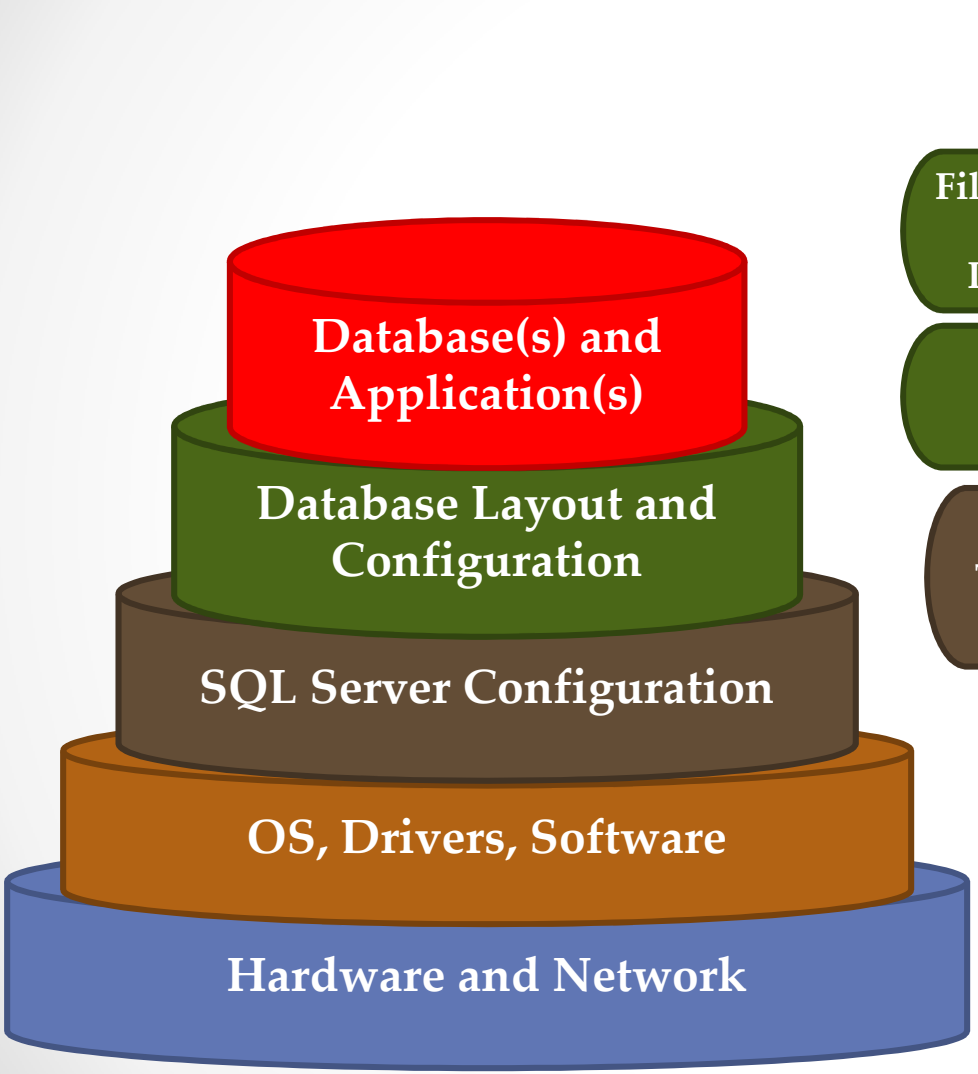
- 11+ years of experience working with Microsoft SQL Server
- Microsoft SQL Server MVP
- Microsoft Certified Master (SQL Server 2008)
- MCPD
  - Enterprise Application Developer

- Blog: http://aboutsqlserver.com
  - Session will be available for download
- Email: dmitri@aboutsqlserver.com

What is it all about?

- **We will talk about:**
  - SQL Server execution model
  - Wait Statistics 101- How different problems present themselves
- **Session goals:**
  - Share the experience
  - Demonstrate the set of techniques that helps to analyze OLTP systems
- **What is out of scope:**
  - We don't want to miss lunch, do we?
  - How to configure and maintain SQL Server instances
  - **Troubleshooting of Data Warehouse / Reporting blueprint systems**

Full Picture



# of VLFs
Auto-Shrink
Filegroups and RAIDs
Auto-Growth
Multiple DBs on the Server
Auto-Close
Instant File Initialization
Memory Configuration
TempDB
Multiple Instances
Running Software
Faulty Drivers
32 bit OS
Virtualization
Server Specs
IO Subsystem
Network Topology and Configuration

Full Picture (1)

- Hardware and Network
  - Does server have enough power to handle the system?
  - I/O subsystem
    - RAID levels
    - I/O throughput (use SQLIO/SQLIOSim for the testing)
    - Disk alignment and sector size (generally 64K sector is the best)
  - Network throughput – what is the slowest component in the topology?
- OS
  - Are drivers up to date and optimally configured?
  - In case of 32 bit OS – do you have memory settings configured correcly (AWE, /3GB /UserVA)?
  - Do you have Min/Max server memory and "Lock Pages in Memory" set?
  - What software is running on the server?
  - Is it virtual server? Are there balloon driver? Is host overcommitted? What is the current host load?

Full Picture (2)

- SQL Server configuration
  - Do you have multiple instances running on the same server?
  - Do you have multiple databases running on the same server?
    - Is it mixed workload (OLTP/DW)?
    - Different audit/security requirements?
  - TempDB
    - Is it on the fastest disk array?
    - How many files does it have?
    - Is space pre-allocated?
  - What is SQL Server memory configuration?
  - Is Instant File Initialization enabled?
- Database
  - Do you have Auto-shrink and Auto-close disabled?
  - Do you pre-allocate enough space for log file? How many VLF log file has?
  - What log file auto-growth parameters do you have?
  - How many filegroups / files database has?
  - Database files placement and RAID levels

Create Baseline

- Operation standpoint
  - Most part of performance metrics are meaningless by themselves
    - *"I have 25 full scans per second. Is everything OK with my system"?*
    - *"My disk latency is 20ms. Should I be worried?"*
  - Baseline helps to be proactive
- Helps to demonstrate achievements to the management and/or customer ☺
  - *"We decreased CPU utilization"* vs. *"% of signal waits decreased from 50% to 15%".*

SQL Server Execution Model

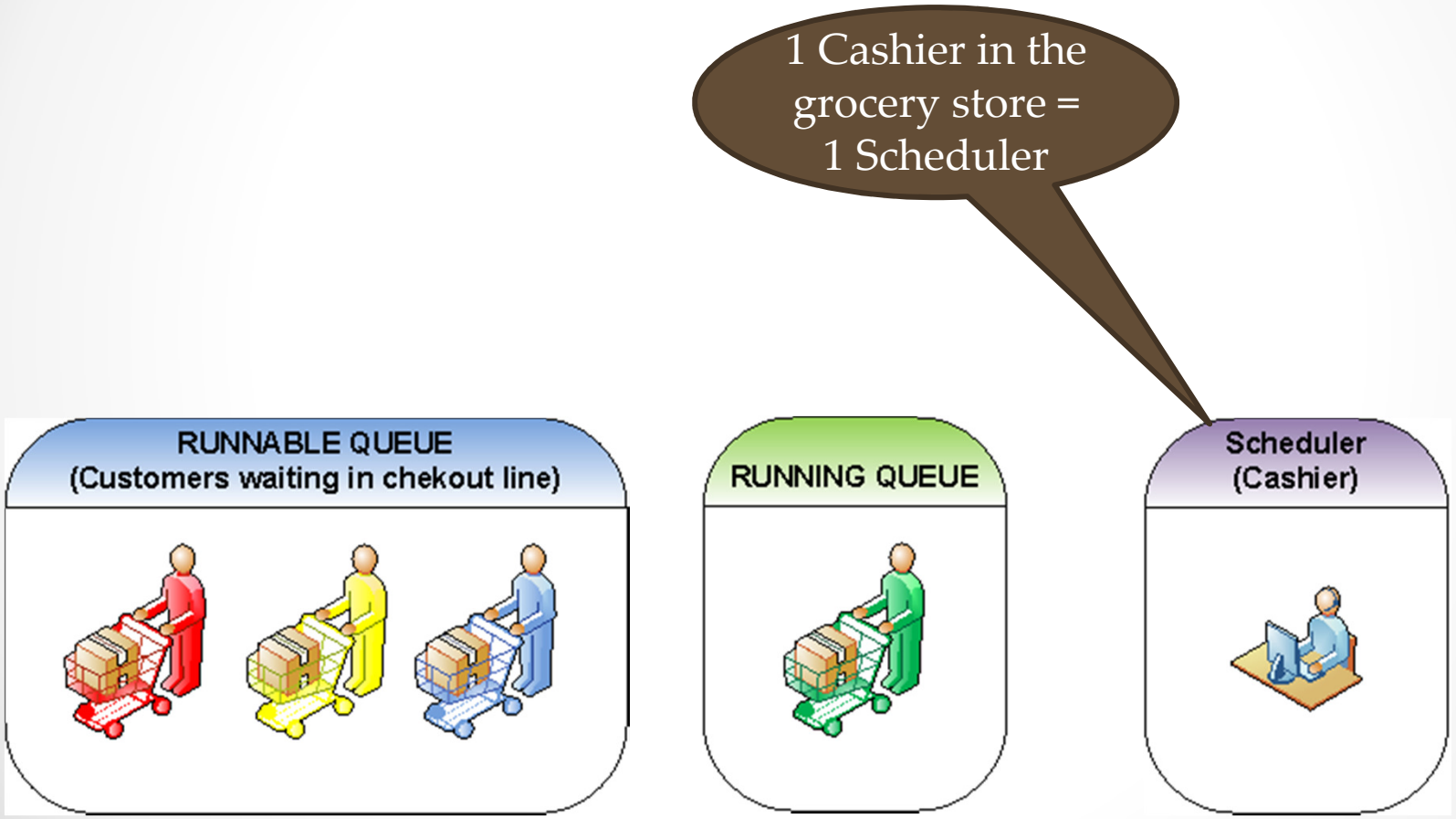
SQLOS

- Layer between SQL Server and Windows
- Responsible for
  - Scheduling
  - I/O operations
  - Memory and Resource Management

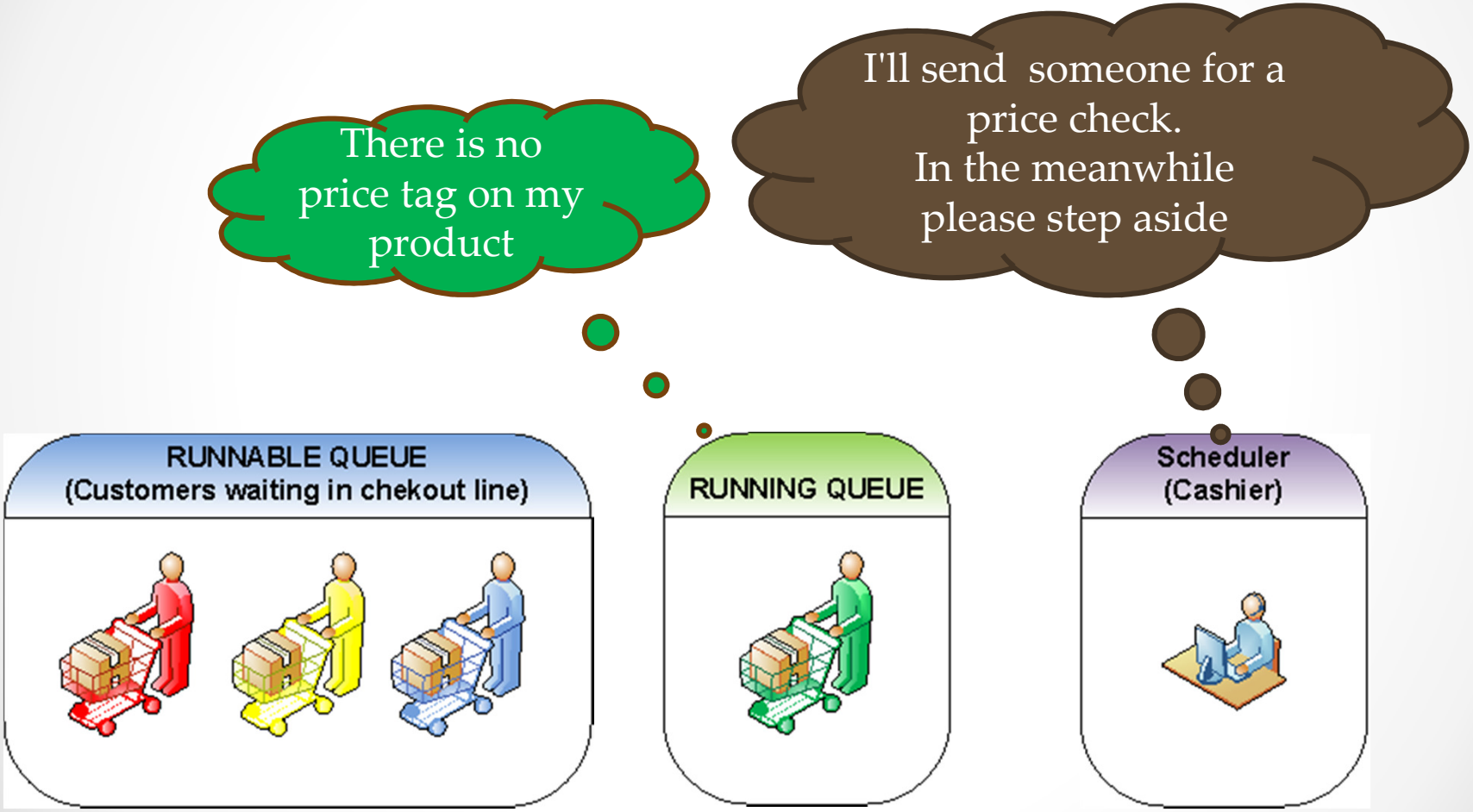
SQL Server Execution Model

- SQLOS assigns 1 scheduler per **logical** CPU
- Worker Threads created and evenly divided across schedulers
- Batch assigns to 1 or multiple workers and stays until completed
- Worker states:
  - Running – currently executing on CPU
  - Suspended – waiting for resource
  - Runnable – waiting for it's turn to be executed

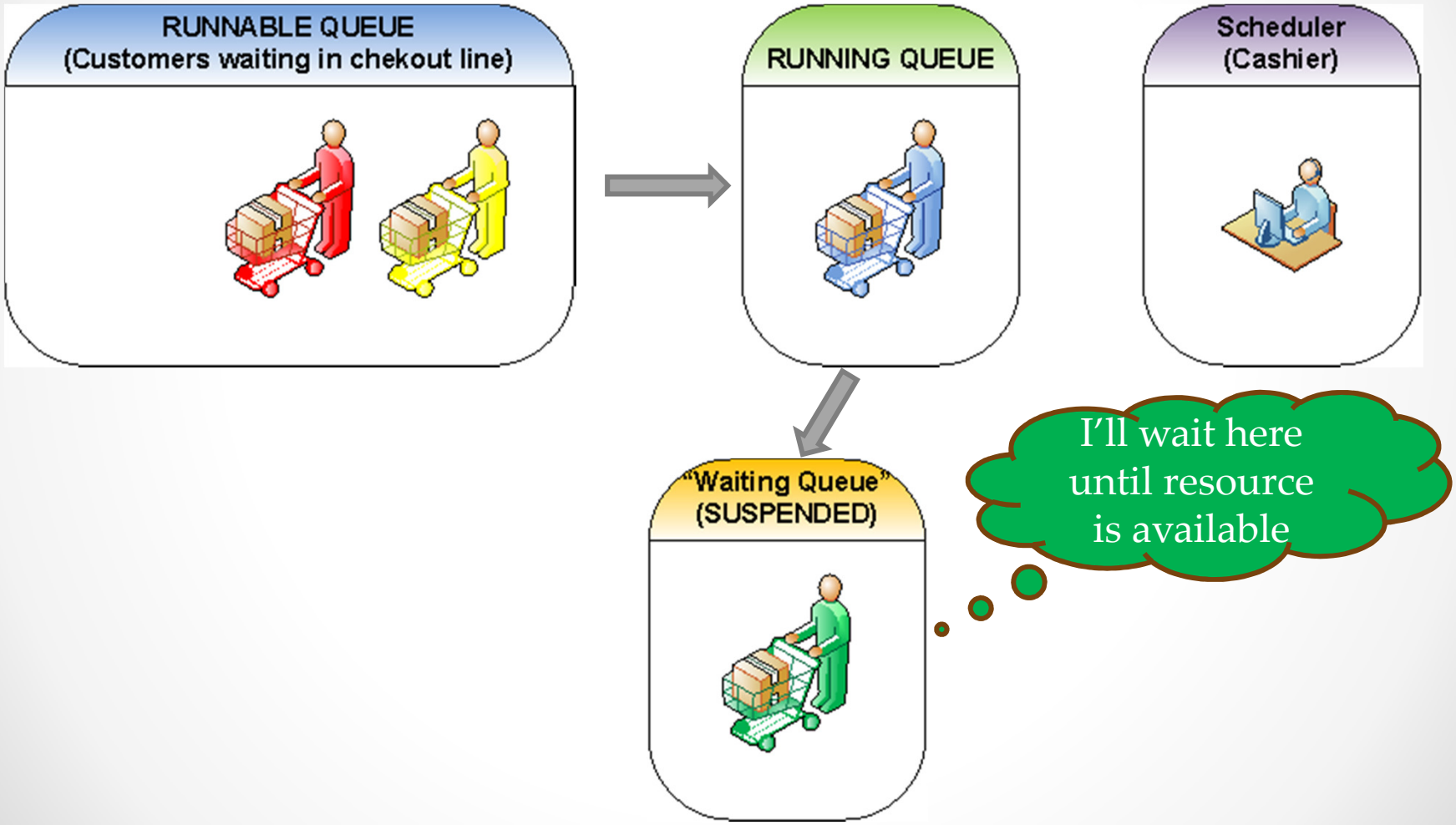
Execution Model – 1 Scheduler



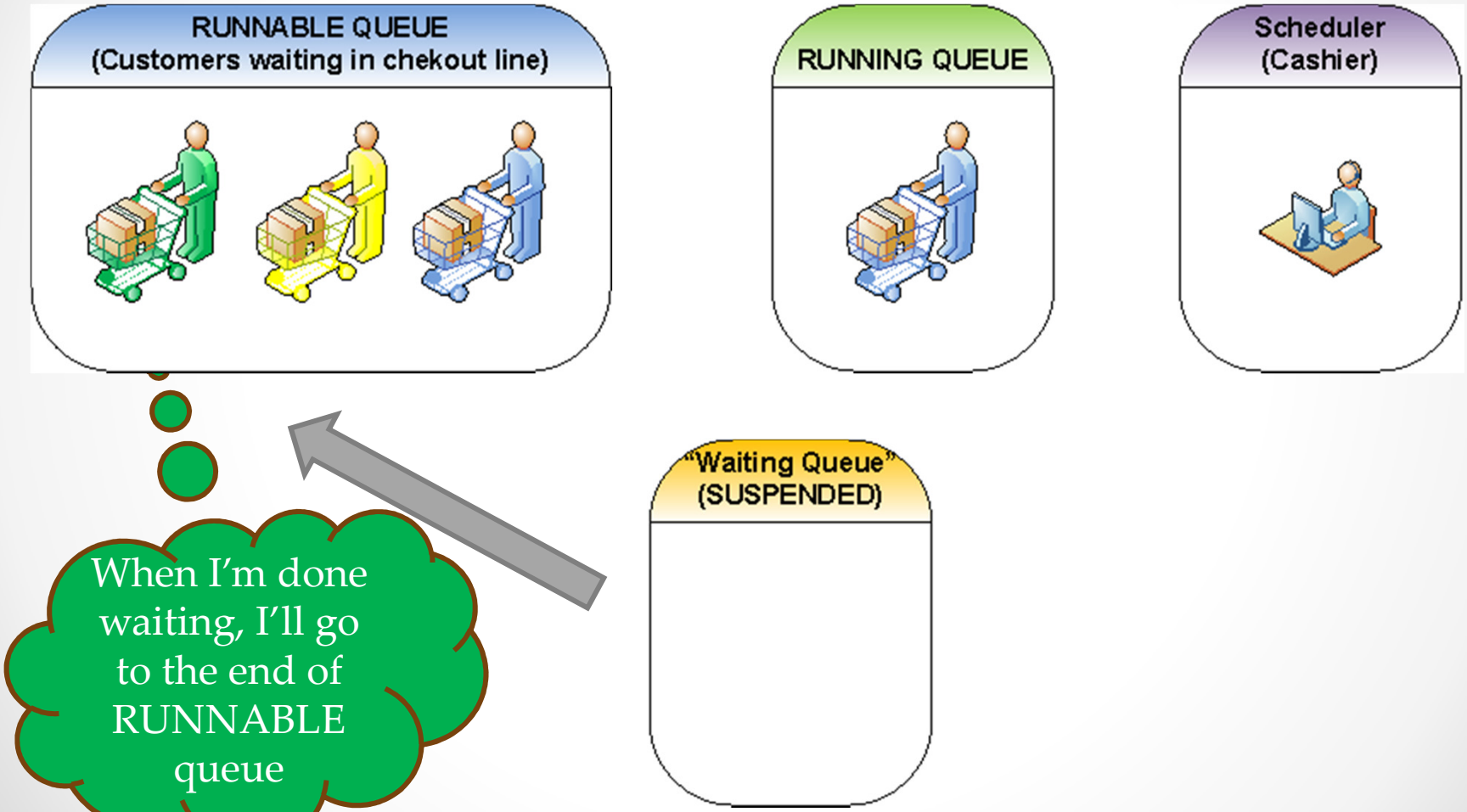
Execution Model – 1 Scheduler



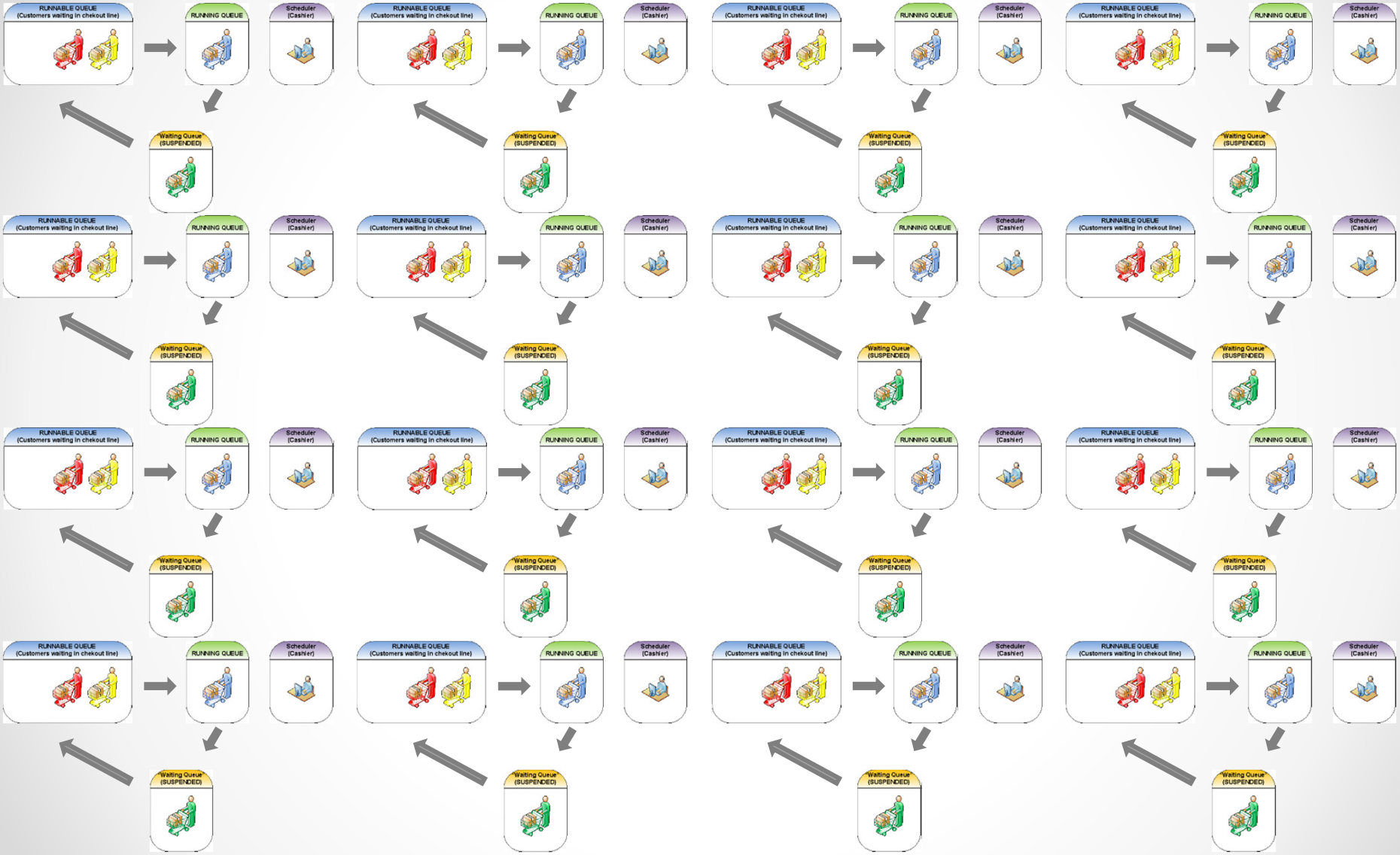
Execution Model – 1 Scheduler



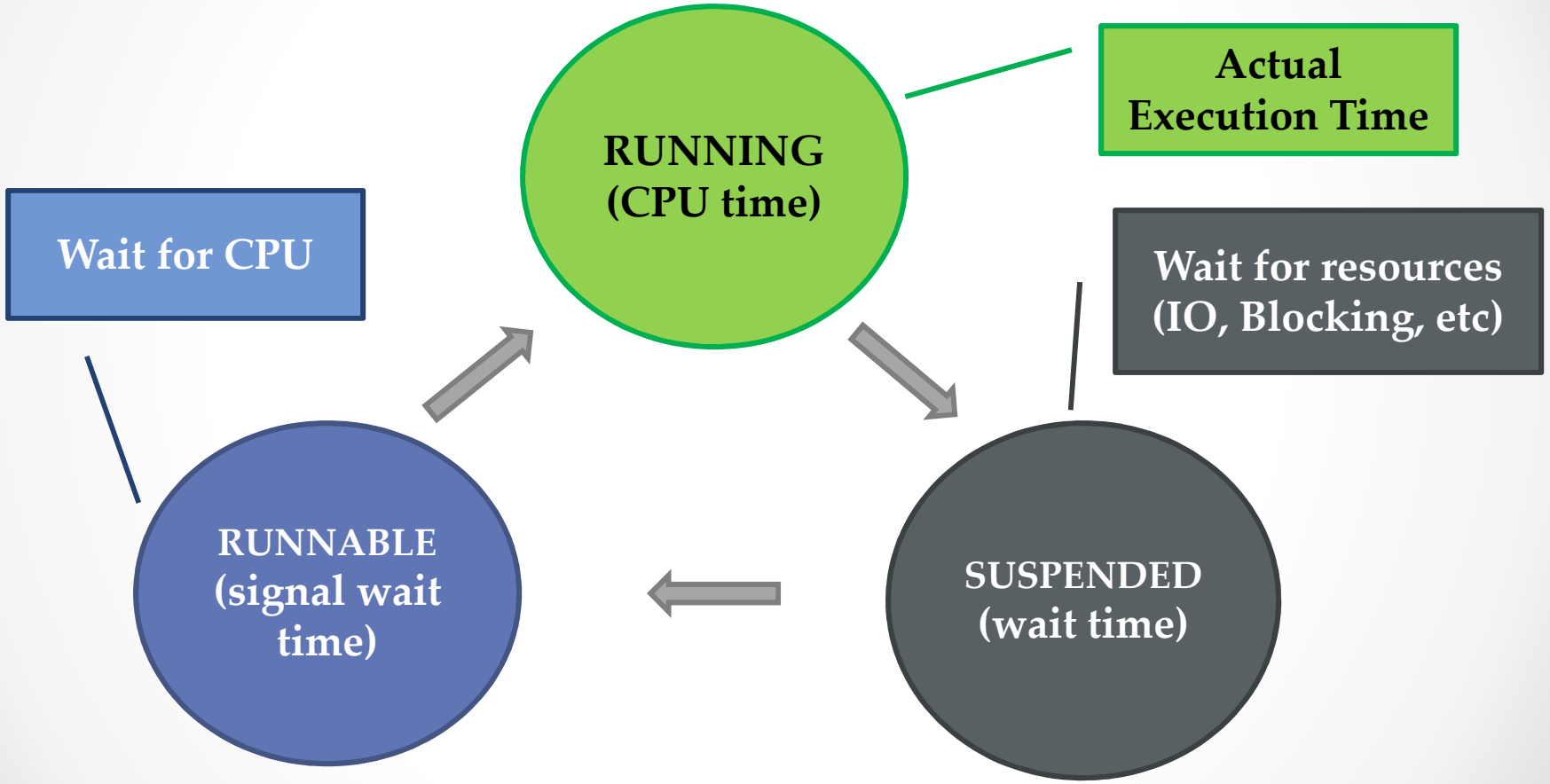
Execution Model – 1 Scheduler



More than 1 CPU?



Simplified Query Life Cycle



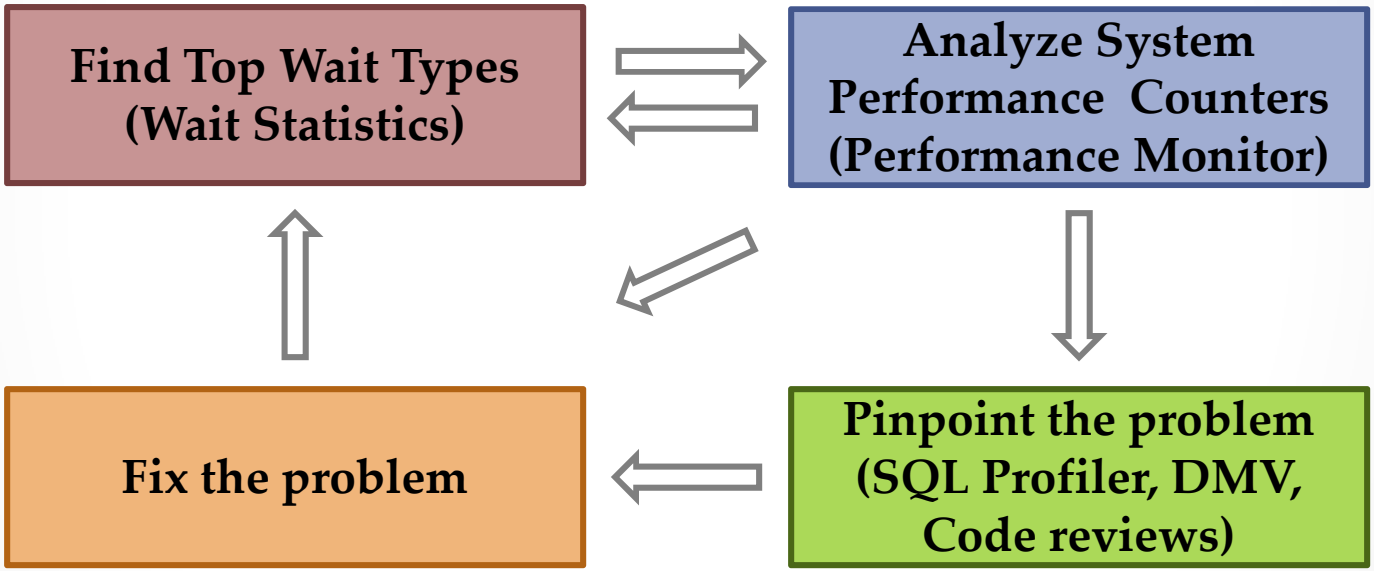
Wait Statistics 101

- Wait Statistics – what server is waiting for

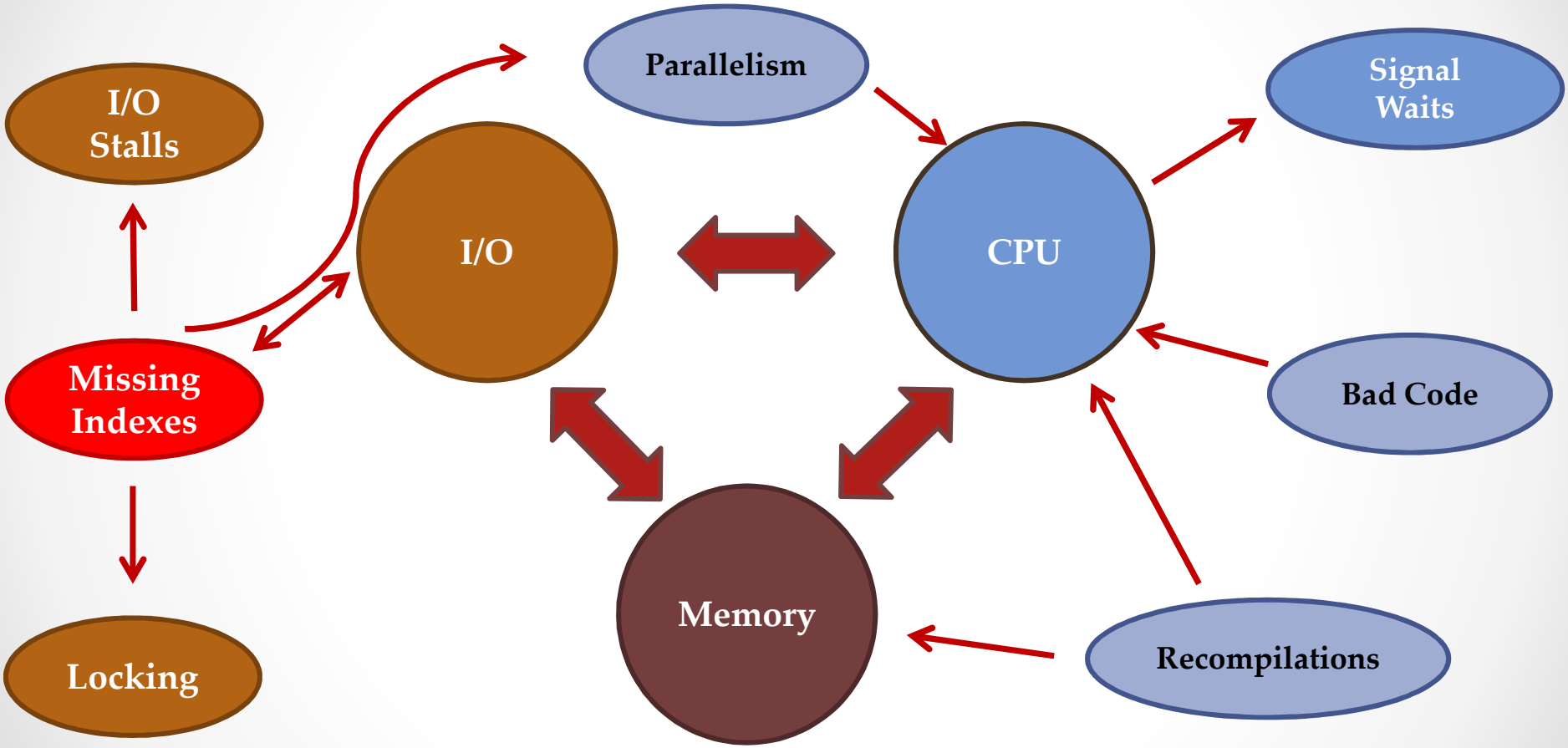
```
SELECT
     wait_type, wait_time_ms,
     convert(decimal(7,4), 100.0 * wait_time_ms / SUM(wait_time_ms) OVER()) AS [Percent]
from
      sys.dm_os_wait_stats
where
     wait_type NOT IN ('CLR_SEMAPHORE','LAZYWRITER_
     ,'SLEEP_SYSTEMTASK','SQLTRACE_BUFFER_FLUSH','W
     ,'REQUEST_FOR_DEADLOCK_SEARCH','XE_TIMER_EVENT
     'CLR_MANUAL_EVENT','CLR_AUTO_EVENT','DISPATCHE
     ,'XE_DISPATCHER_WAIT', 'XE_DISPATCHER_JOIN', '
     ,'OLEDB', 'MSQL_DQ' )
     and wait_type not like 'Broker%'
order by
     [Percent] Desc
```

	wait_type	wait_time_ms	Percent
1	BACKUPBUFFER	11311975189	20.6899
2	BACKUPIO	11153312594	20.3997
3	PAGEIOLATCH_EX	4620890485	8.4517
4	WRITELOG	3983896698	7.2866
5	PAGEIOLATCH_SH	3719440813	6.8029
6	CXPACKET	3630197534	6.6397
7	OLEDB	3464571854	6.3368
8	MSQL_XP	2591424522	4.7398
9	ASYNC_IO_COMPLETION	2028625085	3.7104
10	SOS_SCHEDULER_YIELD	1870324254	3.4209
11	RESOURCE_SEMAPHORE	1314585339	2.4044
12	MSQL_DQ	1072516276	1.9617
13	LCK_M_U	816581103	1.4935
14	ASYNC_NETWORK_IO	598159386	1.0940
15	PAGELATCH_SH	351490272	0.6429

Never-ending troubleshooting

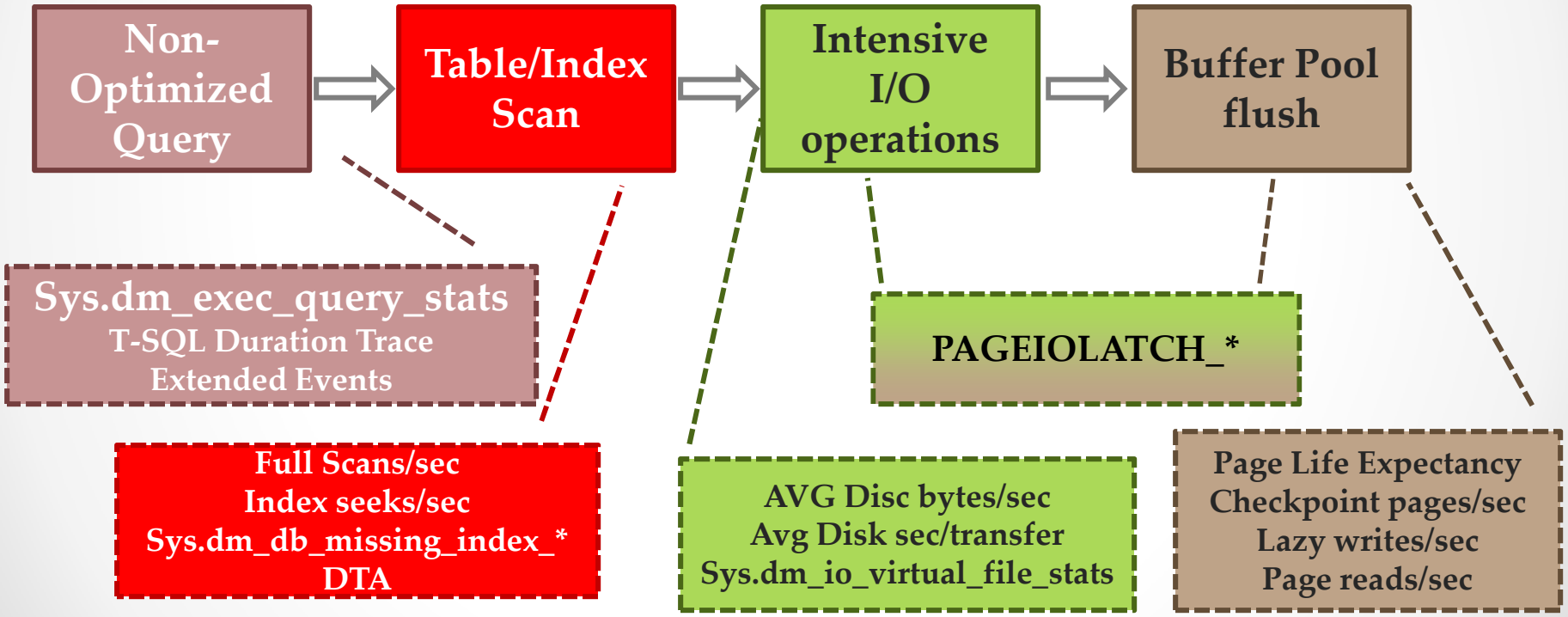


Everything is related



Memory and I/O bottlenecks

- In 95% of the cases caused by non-optimized queries



I/O and Memory issues troubleshooting

Type	Name	Description
Wait Types:	**PAGEIOLATCH_***	Disk to memory transfer
	IO_COMPLETION	I/O operations. Usually non data pages
	ASYNC_IO_COMPLETION	Asynchronous I/O
	WRITELOG, LOGMRG	Log I/O operations
Performance Objects:	~~Buffer cache hit ratio~~	How often page found in the cache. Do not use
	~~(Avg) Disk Queue Length~~	The length of the disk queue.
	Page life expectancy	How long page stays in the cache. Watch the trends. As the starting point – should be > (DB_CACHE_SIZE / 4GB) * 300 sec.
	Checkpoint pages/sec Lazy writers/sec	How often pages saved to disk Memory pressure: High values + low page life expectancy
	Page reads/sec	Number of physical page reads that are issued per second
	Avg Disk Bytes/* Avg Disk sec / Transfer	Disk performance counters

I/O and Memory issues troubleshooting

Type	Name	Description
Wait Types:	RESOURCE_SEMAPHORE	Memory grants wait and statistics Waits should be minimal for OLTP Expected for Data Warehouse type systems
Performance Objects:	Memory Grant Pending	
	Memory Grant Outstanding	
DMV:	**Sys.dm_exec_query_stats**	Query execution statistics
	Sys.dm_io_virtual_file_stats	I/O statistics for database files. Io_stall – total time that users waited for I/O
	sys.dm_os_memory_clerks DNCC MEMORYSTATUS	What is using memory

Sys.dm_exec_query_stats

```
SELECT TOP 250
    SUBSTRING(qt.TEXT, (qs.statement_start_offset/2)+1,
        ((
            CASE qs.statement_end_offset
                WHEN -1 THEN DATALENGTH(qt.TEXT)
```

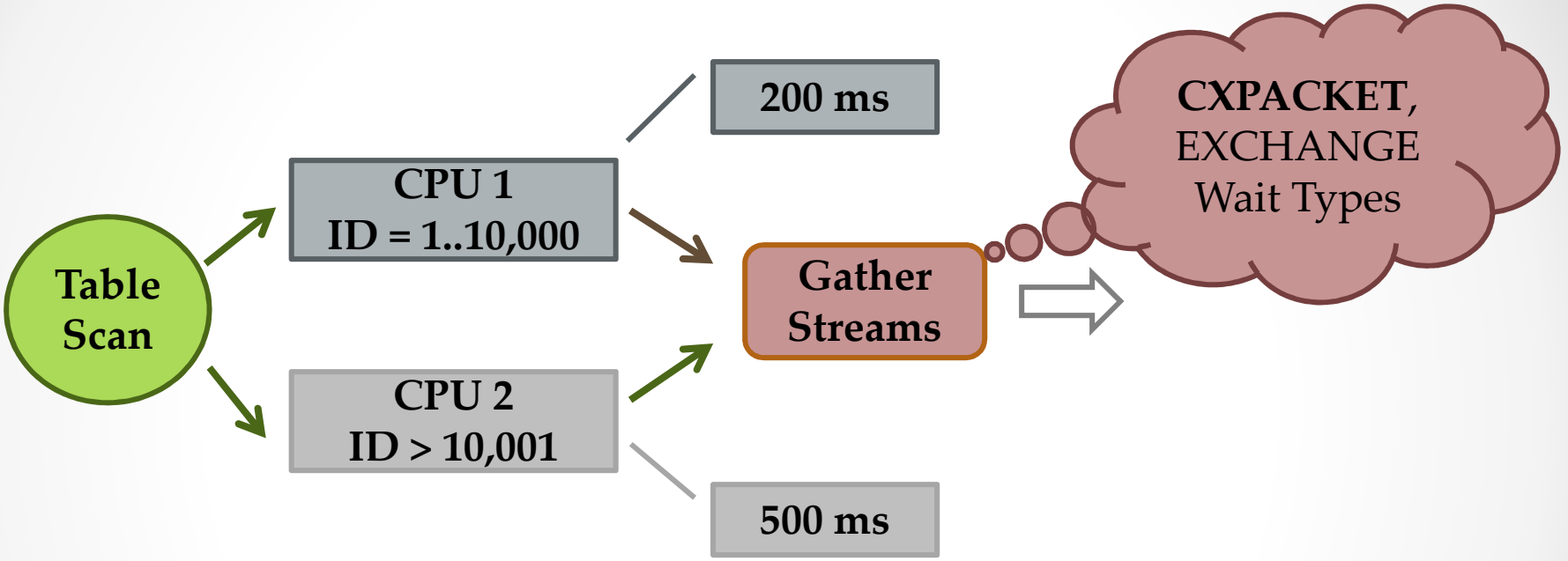
	SQL	Exec ...	Avg IO	query_plan	Total Reads	Total Writes	Total CPU
1	select Subj, cast(R...	1	6816382	<ShowPlanXM...	6816296	86	24297389
2	select UID, DOCTY...	26455	4143503	<ShowPlanXM...	109616393555	0	154369131409
3	DELETE TOP (@d...	1	4096631	<ShowPlanXM...	4096468	163	26538518
4	insert into #tmpRep...	62	3690210	NULL	228750206	42859	3351099613
5	update #tmpReportI...	62	3139967	NULL	194677952	7	2406888686
6	insert into #tmpRep...	58	2516483	NULL	145905711	50341	1761652781
7	select D.*, O.CATE...	16	1848720	<ShowPlanXM...	29579527	0	64629691
8	update #tmpReport...	13	1520333	<ShowPlanXM...	19764334	5	194722131
9	select D.*, I.Catego...	36	1511917	<ShowPlanXM...	54429042	0	114735561
10	update #tmpReport...	26	1459946	<ShowPlanXM...	37958482	138	447010567
11	update #tmpReport...	12	1426777	<ShowPlanXM...	17121325	4	164099386
12	insert into #tmpRep...	53	1079374	NULL	57198359	8467	865721533

```
ORDER BY
    [Avg IO] desc
option (recompile)
```

Troubleshooting IO Issues

Demo

Parallelism issues



- Parallelism is not required for *tuned* OLTP Systems
- Parallelism always exists in Data Warehouse Systems
- MaxDOP must be <= # of CPUs per hardware NUMA node
- Consider to increase "Cost Threshold for Parallelism" rather than change MAXDOP in OLTP

Troubleshooting Parallelism

Demo

CPU Bottleneck

Type	Name	Description
Wait Types:	**SOS_SCHEDULER_YIELD**	Task is waiting for its quantum to be renewed
	CMEMTHREAD	Memory allocation from the same object. Possibly Ad-hoc sql
DMV:	**Sys.dm_os_wait_stats**	Signal_wait_time_ms > 25% of total waits
	sys.dm_os_memory_clerks	CACHESTORE_SQLCP: Memory for Ad-Hoc query plans
Performance Objects:	Batch Requests/sec	Total Batch Requests per second
	SQL Compilations/sec	Initial compilations + recompilations
	SQL Re-Compilations/sec	Recompilations

- Could mask:
  - Excessive Ad-Hoc SQL / Dynamic SQL / recompilations
  - Bad SQL Code
  - Non-optimized queries
- OLTP Systems:
  - Initial Compilations = Sql Compilations/sec – SQL Re-Compilations/sec
  - Plan Reuse = (Batch requests/sec – Initial Compilations) / Batch request/secs > 90%

Troubleshooting Recompilations

Demo

Scalar Functions

Demo

Async_Network_IO

- Server waits for client to consume data
- Could be:
  - Network issues
  - Client code issues
    - **READ ALL DATA BEFORE PROCESSING!**

Troubleshooting Recompilations

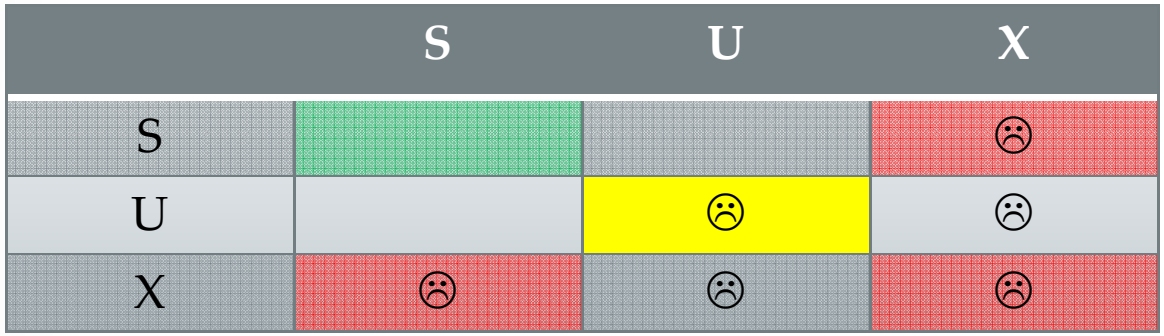
Demo

Locking, Blocking and Deadlocks

Type	Name	Description
Wait Types:	**LCK_M_***	Waiting for lock to be obtained
DMV:	**Sys.dm_tran_locks**	Currently active locks
Traces & **Extended Events**	Blocked Process Report	Tasks have been blocked for more than specified amount of time
	Deadlock graph	Deadlocks
Performance Objects:	Counters from <Instance>\Locks	Locks/Timeouts/Deadlocks statistics

Why Locking?

- Major Lock Types:
  - Shared (S) – acquired by readers
  - Exclusive (X) – acquired by writers
  - Update (U) – acquired by writers while locating rows for update
- Lock Compatibility Matrix:



	S	U	X
S			☹
U		☹	☹
X	☹	☹	☹

- SQL Server always obtains U/X locks regardless of isolation level (even read uncommitted)
- (X) Locks held till end of transactions
- **Beware of non-optimized queries**

Locking Issues

Demo

Lock Escalation

- SQL Server tries to escalate locks to the table/partitions level
  - Initial Threshold: ~5,000 locks on the object
  - If it fails, it tries again every ~1,250 locks
- Pattern: batch operation triggers lock escalation. All other sessions accessing the object are blocked
- Troubleshooting
  - High wait % of intent locks (LCK_M_I*)
  - SQL Profiler Locks: Escalation event
- Solution
  - Trace flag 1211 (instance level) – not recommended but sometimes required
  - SQL Server 2008+: *alter table .. set lock_escalation*
  - Optimistic transaction isolation levels
    - Row version model – writers don't block readers

Lock Escalation

Demo

Real Life Story

	wait_type	wait_time_...	Percent
1	CXPACKET	47237677	37.0492
2	LCK_M_IS	17641793	13.8367
3	PAGELATCH_UP	10757870	8.4375
4	LCK_M_SCH_S	10103857	7.9246
5	ASYNC_NETWORK_IO	9715441	7.6200
6	SOS_SCHEDULER_YIELD	8970275	7.0355
7	LCK_M_SCH_M	5748216	4.5084
8	OLEDB	3335574	2.6161
9	LCK_M_IX	3000305	2.3532
10	LATCH_EX	2621557	2.0561
11	ASYNC_IO_COMPLETION	1613775	1.2657
12	BACKUPIO	1443624	1.1323
13	IO_COMPLETION	1115441	0.8749
14	BACKUPBUFFER	902306	0.7077
15	WRITELOG	882498	0.6922

- Symptoms:
  - High % of Schema Lock Waits
  - High % of Parallelism Waits
  - Almost none Data I/O waits
- Step 1:
  - Focusing on the Schema Lock Waits
- Detected problem:
  - Constant rebuild of FTS index

Real Life Story

	wait_type	wait_time_...	Percent
1	CXPACKET	6059039	44.1425
2	ASYNC_IO_COMPLETION	1747127	12.7285
3	BACKUPIO	1483546	10.8082
4	BACKUPBUFFER	866660	6.3140
5	ASYNC_NETWORK_IO	573897	4.1811
6	SOS_SCHEDULER_YIELD	471540	3.4354
7	BACKUPTHREAD	436083	3.1770
8	LATCH_EX	417119	3.0389
9	IO_COMPLETION	331552	2.4155
10	LCK_M_S	299947	2.1852
11	WRITELOG	258726	1.8849
12	LCK_M_U	151601	1.1045
13	PAGEIOLATCH_EX	150622	1.0973

- Symptoms:
  - High % of Parallelism Waits
  - High % of Signal Waits
  - Almost none Data I/O waits
  - ~20% CPU Utilization
  - No Memory Pressure
- Detected problem:
  - Poorly optimized queries
  - Excessive use of multi-statement functions
  - Database is almost fully cached
    - No Physical data IO occurs

So.. If main bottleneck is

- I/O
  - Focus on I/O
- I/O and Memory
  - Focus on I/O
- Memory without I/O
  - Check Logical-only I/O
  - Check memory clerks
  - Google It ☺
- Parallelism in OLTP system
  - Most likely non-optimized queries
  - Increase "Cost Threshold for Parallelism" if needed rather than change MaxDOP
- Locking and blocking
  - Detect problematic queries
  - Beware of Lock Escalation
  - As the temporary solution – switch to READ COMMITTED SNAPSHOT
    - Be careful!
  - Focus on I/O. If I/O *looks* OK – check client code.

Q & A

- Thank you for the attending!
- Session will be available for download
  - http://aboutsqlserver.com/presentations

- Email: dmitri@aboutsqlserver.com