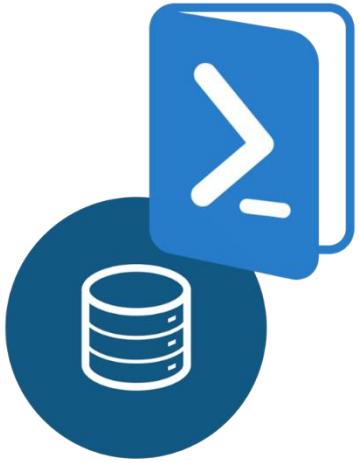
SQL SERVER BENCHMARKING
THE POWERSHELL SPEEDOMETER

Mike Fal - www.mikefal.net

www.mikefal.net

Microsoft
CERTIFIED
Solutions Expert
Data Platform



@Mike_Fal

Get-Agenda

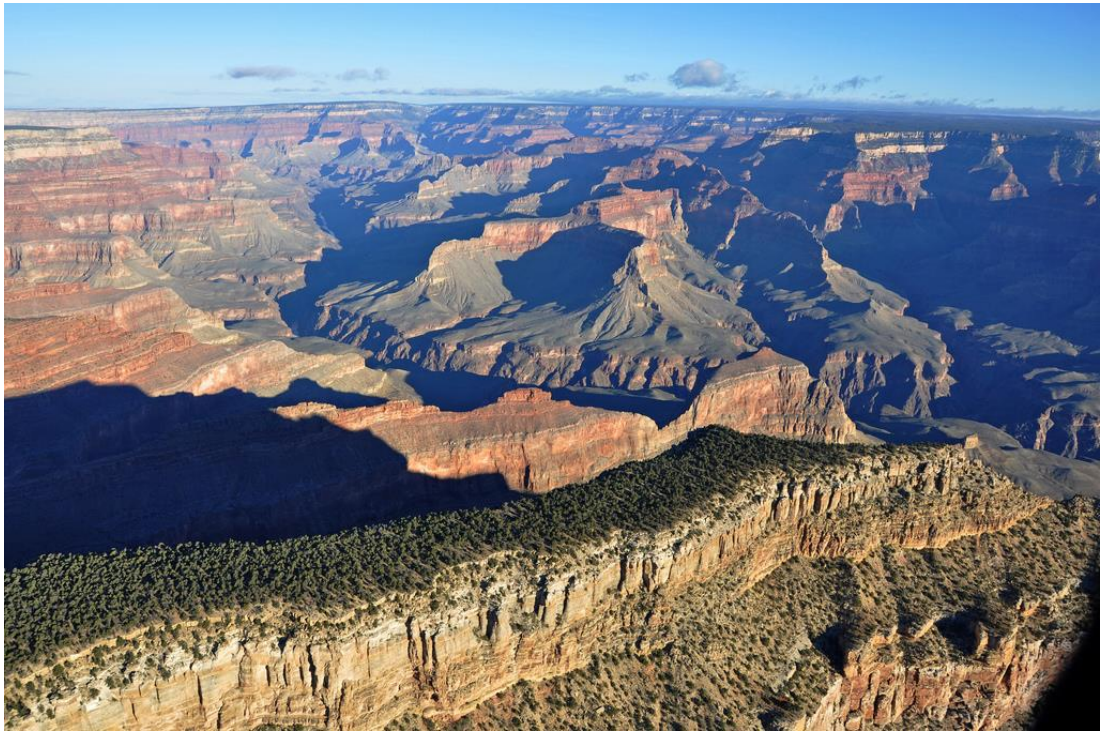
What is benchmarking?

Benchmarking Tools

Benchmarking Metrics

Demo – Capturing a Benchmark

Ground Rules



Don’t focus on the code, focus on the concepts.

Ask questions!

Benchmarking?

Simple Real Life Benchmarking

6 miles in 36 minutes - 6:00 per mile (10 MPH)

6 miles in 30 minutes - 5:00 per mile (12 MPH)

Tools and Metrics

Tools:

- Something to measure distance (GPS, tape measure, etc.)
- Something to measure time (Stopwatch, timer)

Metrics:

- Time to complete the distance
- Speed/pace

Traditional Tools - Perfmon

File Action View Window Help

- Performance
  - Monitoring Tools
    - Performance Monitor
  - Data Collector Sets
  - Reports

100 90 80 70 60 50 40 30 20 10 0

10:43:57 AM 10:44:10 AM 10:44:20 AM 10:44:30 AM 10:44:40 AM 10:44:50 AM 10:45:00 AM 10:45:10 AM 10:45:20 AM 10:43:5... 10:43:56 AM

Last 0.000 Average 7.093 Minimum 0.000 Maximum 99.971 Duration 1:40

Show	Color	Scale	Counter	Instance	Parent	Object	Computer
☑		1.0	% Processor Time	_Total	---	Processor Information	\\SHION
☑		1000.0	Avg. Disk sec/Read	_Total	---	PhysicalDisk	\\SHION
☑		1000.0	Avg. Disk sec/Write	_Total	---	PhysicalDisk	\\SHION
☑		1.0	Disk Transfers/sec	_Total	---	PhysicalDisk	\\SHION
☑		1.0	Buffer cache hit ratio	---	---	SQLServer:Buffer Manager	\\SHION
☑		0.1	Page life expectancy	---	---	SQLServer:Buffer Manager	\\SHION

Traditional Tools - T-SQL

SQLQuery1.sql - SHION.tempdb (SHION\Mike (55))* - Microsoft SQL Server Management Studio

File Edit View Query Project Debug Tools Window Help

New Query | tempdb | Execute | Debug

Object Explorer

Connect

- SHION (SQL Server 12.0.2000 - SHION\
  - Databases
  - Security
  - Server Objects
  - Replication
  - AlwaysOn High Availability
  - Management
  - Integration Services Catalogs
  - SQL Server Agent

SQLQuery1.sql - SH...(SHION\Mike (55))*

```
select * from sys.dm_os_performance_counters
```

100 %

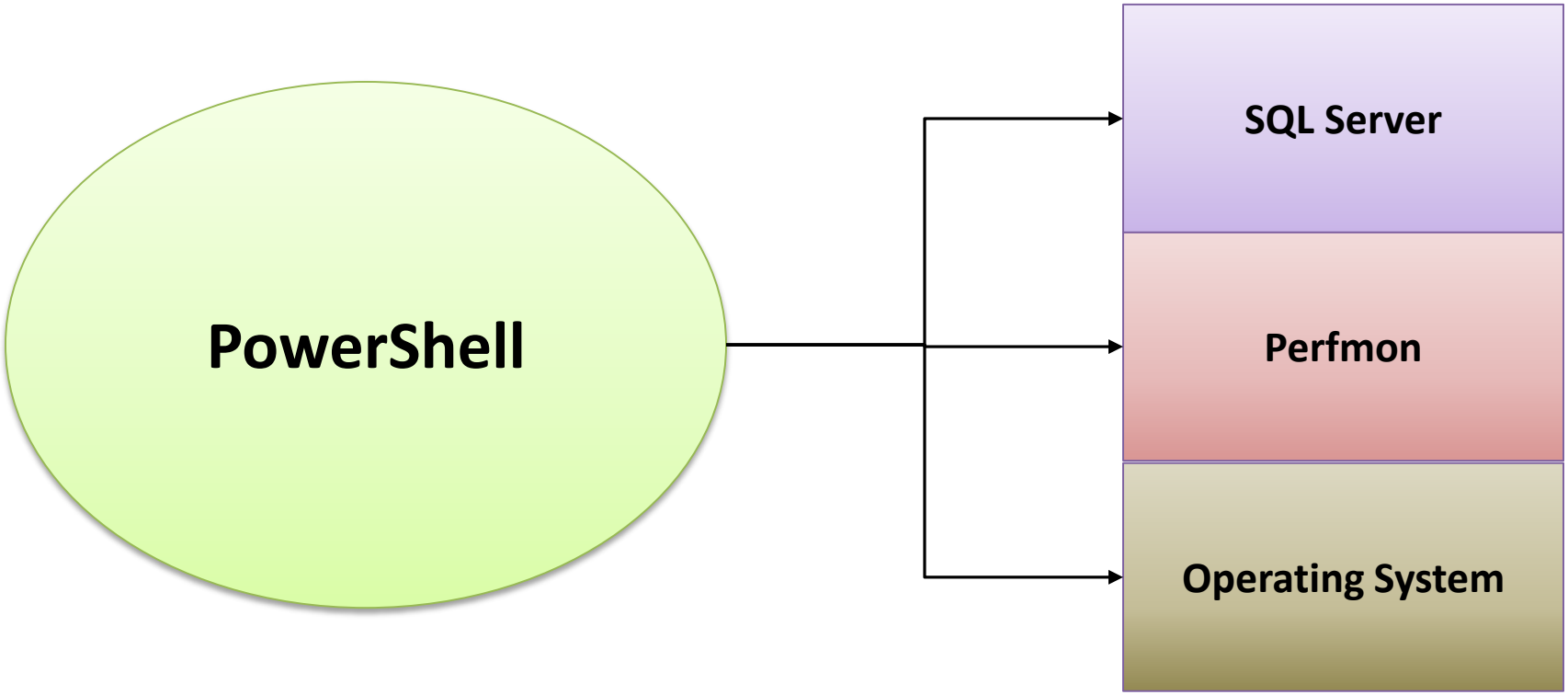
Results | Messages

	object_name	counter_name	instance_name	cntr_value	cntr_type
1	SQLServer:Memory Broker Clerks	Memory broker clerk size	Buffer Pool	8034	65792
2	SQLServer:Memory Broker Clerks	Simulation benefit	Buffer Pool	0	65792
3	SQLServer:Memory Broker Clerks	Simulation size	Buffer Pool	0	65792
4	SQLServer:Memory Broker Clerks	Internal benefit	Buffer Pool	0	65792
5	SQLServer:Memory Broker Clerks	Periodic evictions (pages)	Buffer Pool	0	65792
6	SQLServer:Memory Broker Clerks	Pressure evictions (pages/sec)	Buffer Pool	0	272696576
7	SQLServer:Memory Broker Clerks	Memory broker clerk size	Column store object pool	4	65792
8	SQLServer:Memory Broker Clerks	Simulation benefit	Column store object pool	0	65792
9	SQLServer:Memory Broker Clerks	Simulation size	Column store object pool	0	65792
10	SQLServer:Memory Broker Clerks	Internal benefit	Column store object pool	0	65792
11	SQLServer:Memory Broker Clerks	Periodic evictions (pages)	Column store object pool	0	65792
12	SQLServer:Memory Broker Clerks	Pressure evictions (pages/sec)	Column store object pool	0	272696576
13	SQLServer:Buffer Manager	Buffer cache hit ratio		1223	537003264
14	SQLServer:Buffer Manager	Buffer cache hit ratio base		1250	1073939712
15	SQLServer:Buffer Manager	Page lookups/sec		525127	272696576
16	SQLServer:Buffer Manager	Free list stalls/sec		0	272696576
17	SQLServer:Buffer Manager	Database pages		8034	65792
18	SQLServer:Buffer Manager	Target pages		16187392	65792
19	SQLServer:Buffer Manager	Integral Controller Slope		10	65792

Query executed successfully. | SHION (12.0 RTM) | SHION\Mike (55) | tempdb | 00:00:00 | 1610 rows

Ready | Ln 1 | Col 45 | Ch 45 | INS

Why PowerShell?



Powershell Components

Get-Counter cmdlet

Execute Perfmon collections

Server Management Objects (SMO)

Gather and parse SQL Server information

Custom Powershell Objects

Provide clean reporting of our info

Areas to Benchmark

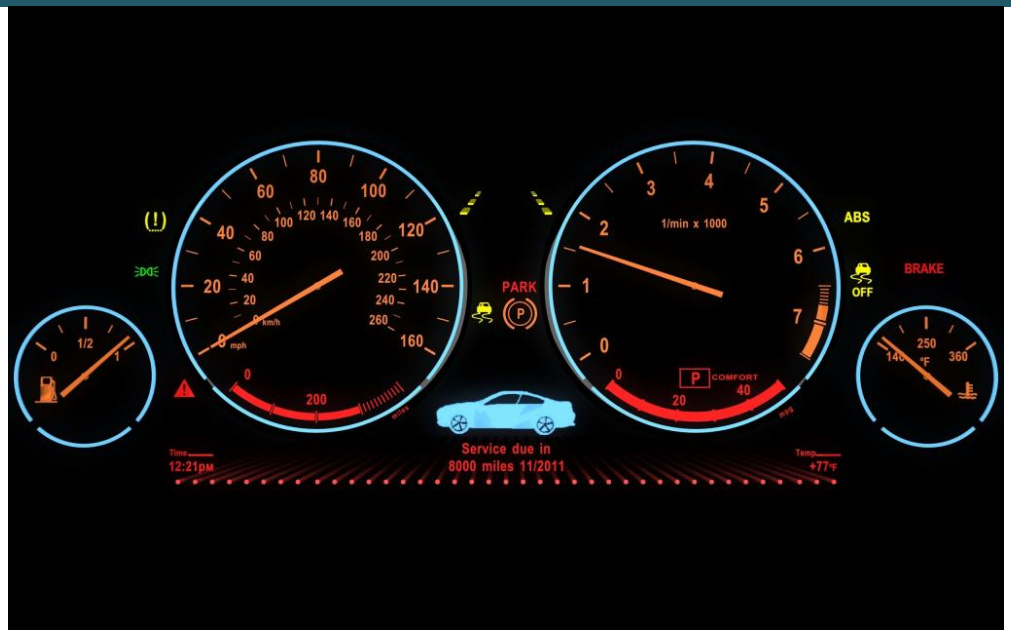
CPU Utilization

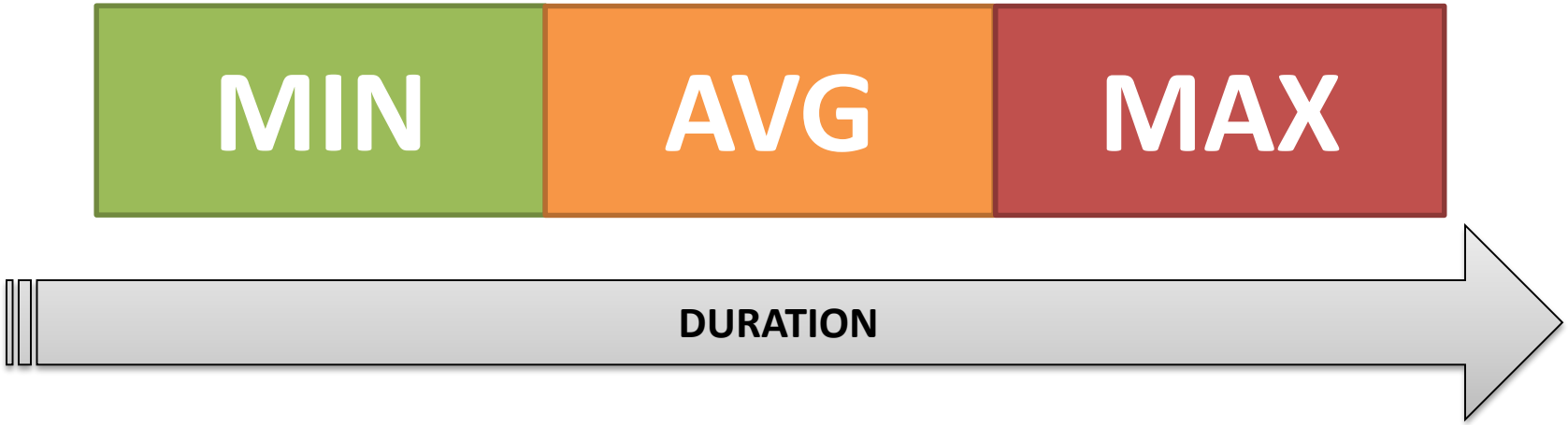
Memory Utilization

Disk Utilization

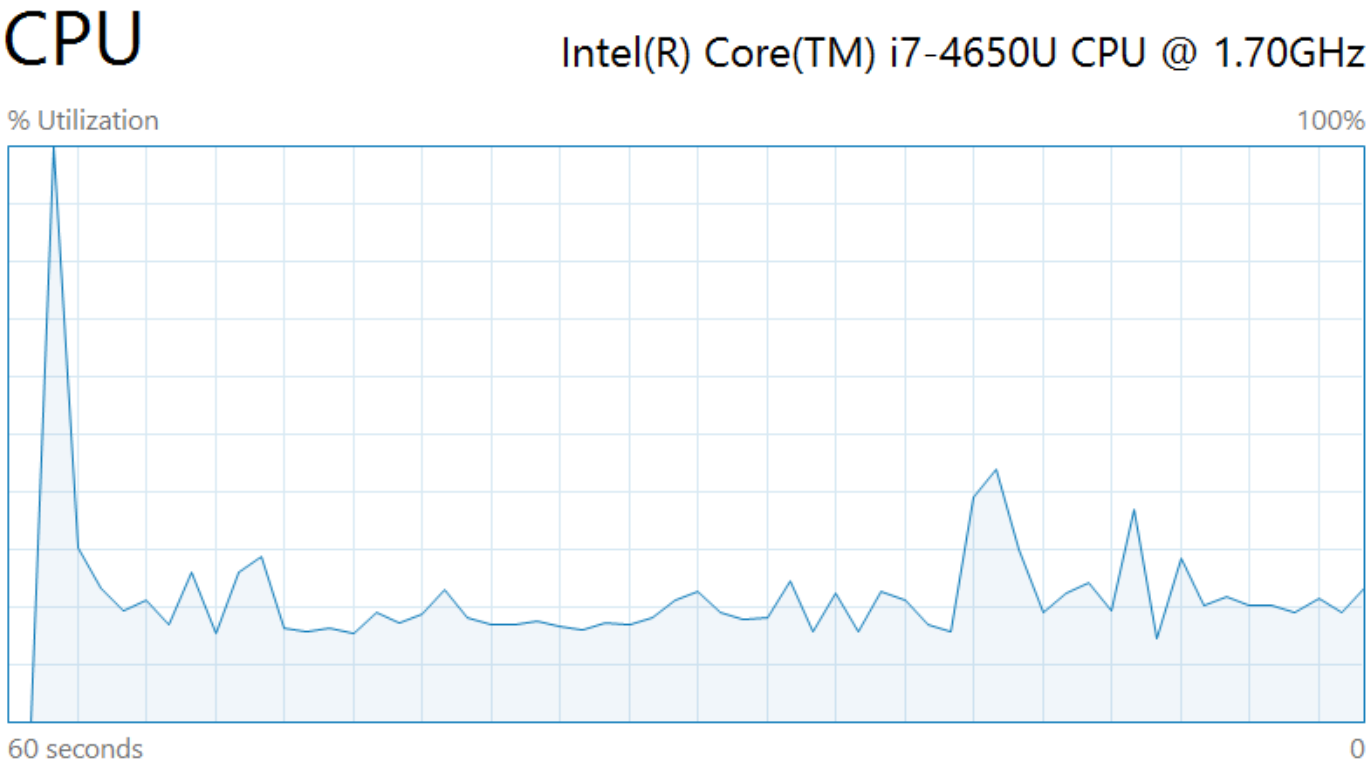
SQL Statistics

Building our SQL Server Speedometer



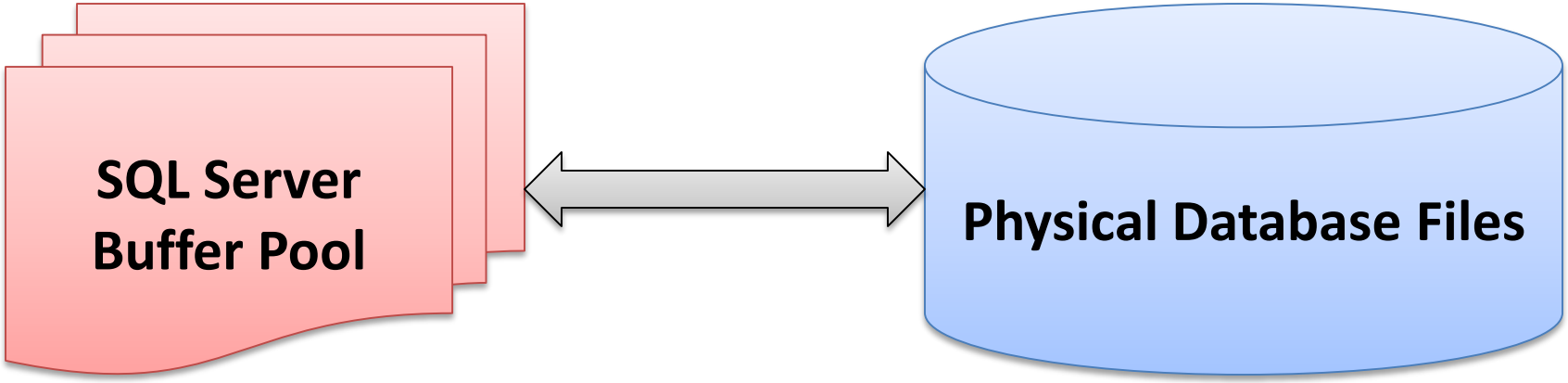


CPU Metrics




```
Get-Counter
-Counter @('\Processor(_Total)\% Processor Time')
```

Memory Metrics/Disk Metrics



Page Life Expectancy	**Average seconds/Read**
Buffer Cache Hit Ratio	**Average seconds/Write**
Total SQL Memory	**Disk Transfers/second(IOPs)**
Available Server Memory	

SQL Metrics

SQL Server Perfmon Metrics

- Batch Requests/second (transactions per second)
- User Connections

SQL Server Wait Statistics

- **Waits and Queues**
- **Top 10 – See where our bottlenecks are**

Demo!

So Now What?

Resources

SQLBenchmarker:

https://github.com/MikeFal/PowerShell/tree/master/SQLBenchmarker

Perfmon Counters:

http://helsinki.sqlpass.org/Portals/139/Files/SQL_post_29x21_2010_PerfmonFinal.pdf

Caveat: Always verify thresholds, this poster is old.

Questions

mike@mikefal.net

www.mikefal.net

@Mike_Fal