

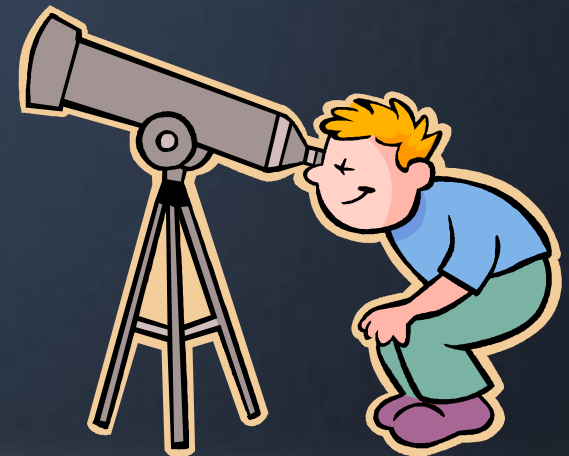


The SQL Server Experts
Discover. Optimise. Mentor.

THROUGH THE VIRTUAL LOOKING GLASS: MONITORING SQL SERVER IN THE VIRTUAL WORLD

Gavin Payne
Senior Consultant

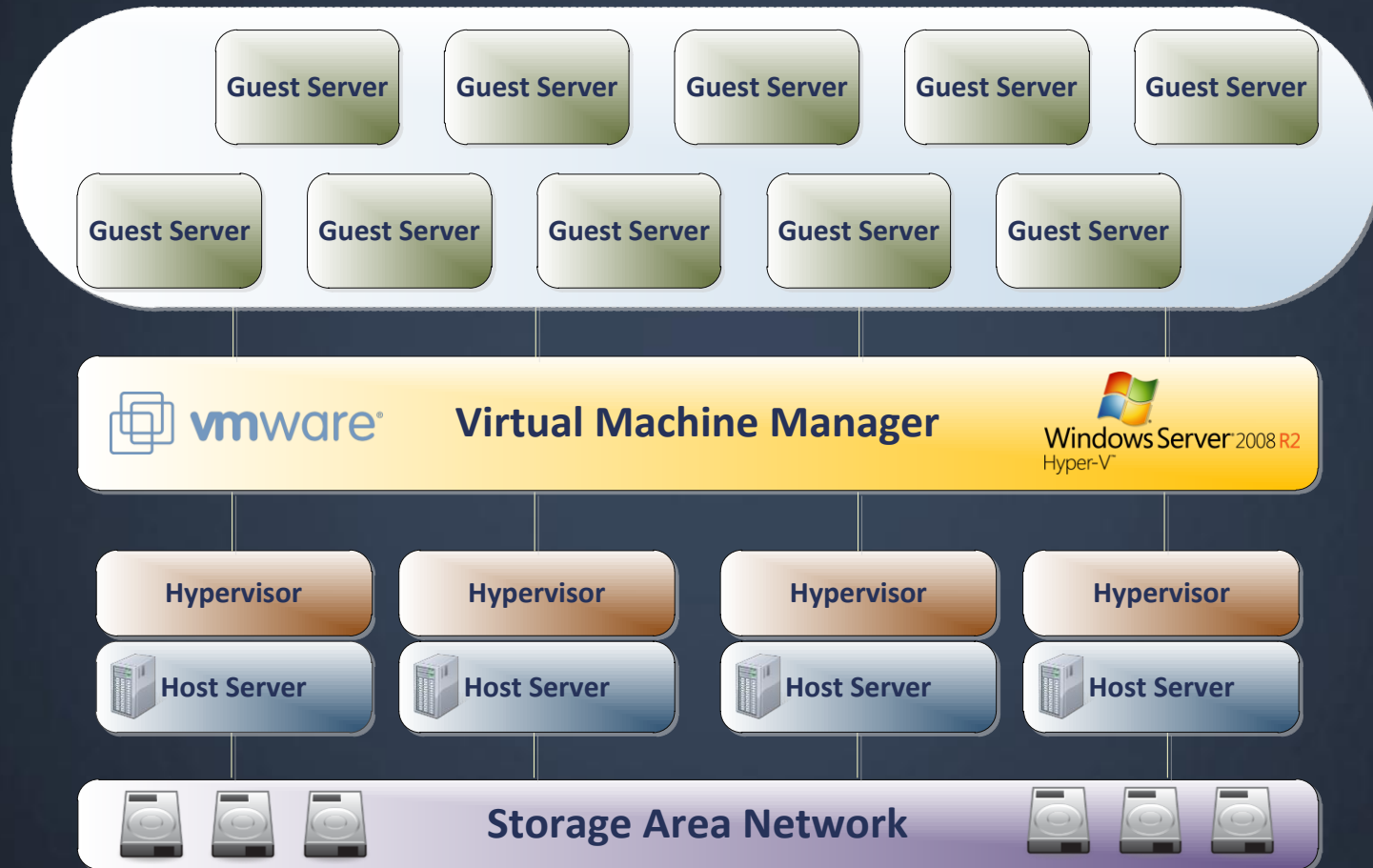
gavin@coeo.com



AGENDA

- Virtualisation refresher
- ***Why we have to change the way we monitor***
- Live monitoring of CPU, Memory and Storage
- Monitoring best practices for the virtual world

VIRTUALISATION TERMINOLOGY



SQL SERVER AND VIRTUALISATION

Detection

- SQL Server will report starting on a virtual server

02/09/2011 19:30:01	Server	Authentication mode is MIXED.
02/09/2011 19:29:59	Server	System Manufacturer: 'Microsoft Corporation', System Model: 'Virtual Machine'.
02/09/2011 19:29:59	Server	Server process ID is 3240.
02/09/2011 19:29:59	Server	All rights reserved.

- However, no new error messages or features

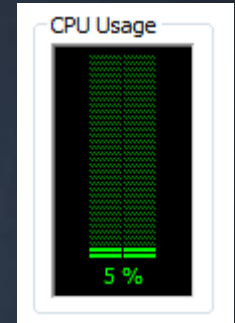
Hot-add Memory

- Existing feature, now used to support Dynamic Memory

GOOD VS. BAD CONTENTION

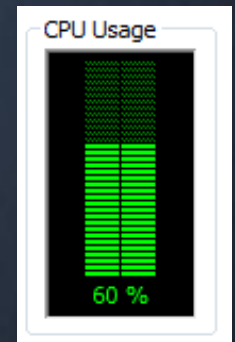
What is good contention?

- 5% CPU usage, 95% free capacity
- Could consolidate 20:1
- Realistically between 10:1 and 15:1



And bad contention?

- The hypervisor will have to control resources
- Immediate access to high performing physical resources can no longer be assured





WHY MONITORING HAS TO CHANGE

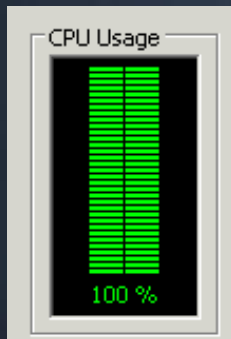


The SQL Server Experts
Discover. Optimise. Mentor.

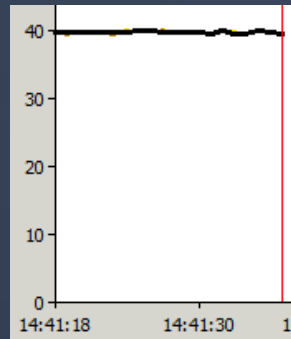
WHY HAS MONITORING CHANGED?

Windows doesn't know the truth anymore

- What does 100% CPU or 8GB available represent?



Task Manager
Guest OS



% Guest Runtime
Host OS

Virtual machine limit (percentage):	<input type="text" value="40"/>
Percent of total system resources:	<input type="text" value="20"/>

Hyper-V
CPU Resource Limit

WHY HAS MONITORING CHANGED?

More sources of confusion

System

Processor: Intel(R) Core(TM) i3-2100 CPU @ 3.10GHz 3.09 GHz
Installed memory (RAM): 2.93 GB
System type: 64-bit Operating System

- Some hardware calls bypass the hypervisor

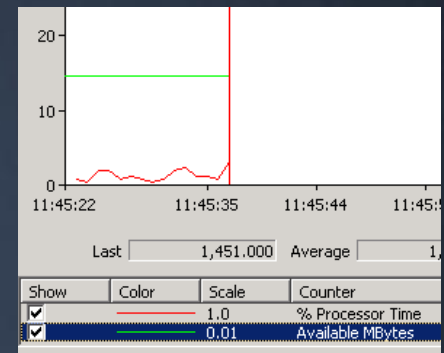
wait_type	waiting_tasks_count	wait_time_ms	max_wait_time_ms	signal_wait_time_ms
PAGEIOLATCH_SH	643	13138	339	15
SOS_SCHEDULER_YIELD	13526	249	36	206

- SQL Server wait stats show symptoms not causes

INFORMATION VS. MISINFORMATION

Information is still visible

- Task Manager
- Performance Monitor and PAL
- SQL Server wait stats and Perfstats



Misinformation is equally visible

- ***Dynamic resource allocation and contention are often invisible to these tools***

CPU MONITORING



The SQL Server Experts
Discover. Optimise. Mentor.

CPU MONITORING

Where to monitor CPU usage

- Wait stats from within SQL Server?

Yes

wait_type	waiting_tasks_count	wait_time_ms
DIRTY_PAGE_POLL	3563	393686
LOGMGR_QUEUE	2850	393657
HADR_FILESTREAM_IOMGR_IOCOMPLETION	739	383577
SOS_SCHEDULER_YIELD	573	1612
SLEEP_TASK	494	192665
SP_SERVER_DIAGNOSTICS_SLEEP	391	387389

- Perfmon within the Guest Server?

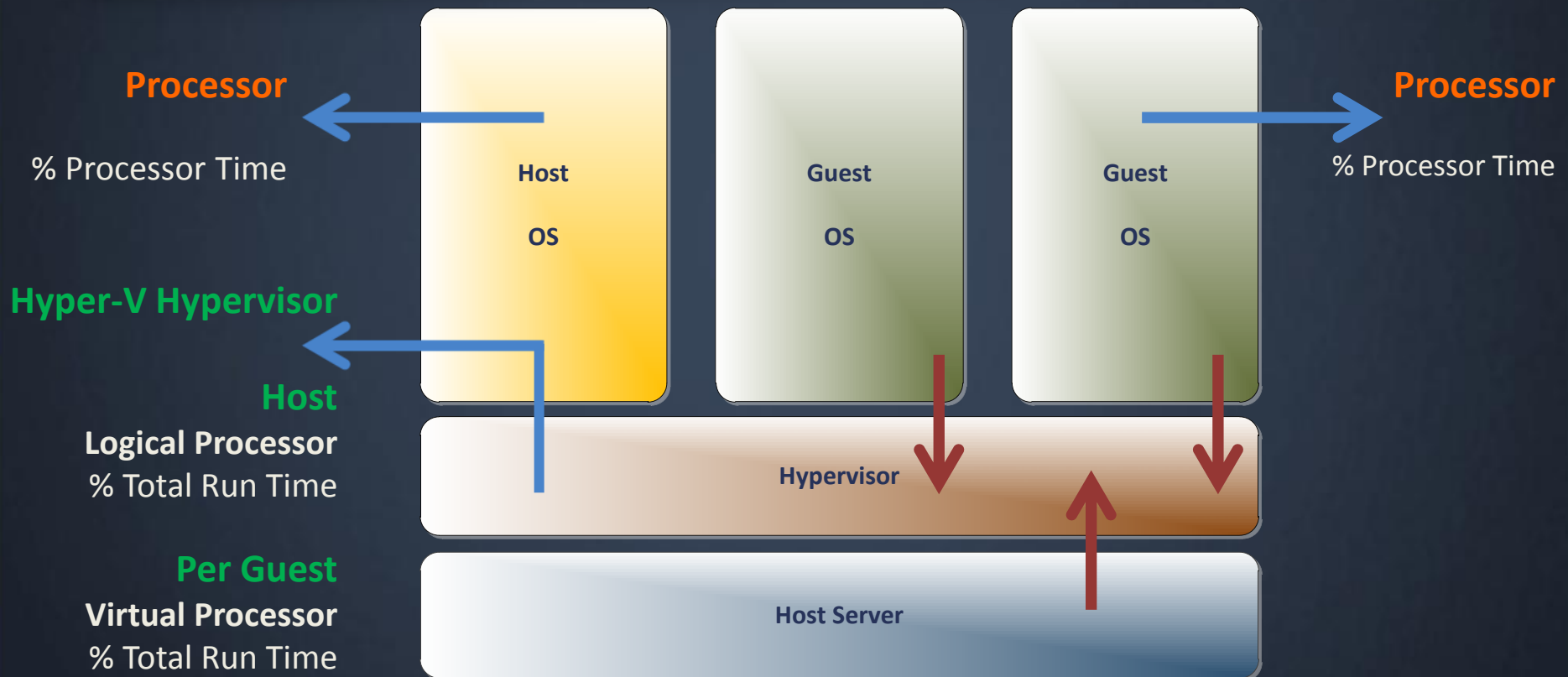
Maybe

- Perfmon within the Host Server?

Yes*

CPU MONITORING

Hyper-V CPU Monitoring Model



A background image showing a close-up of a hand typing on a white computer keyboard. The image is slightly blurred and has a dark, semi-transparent overlay.

DEMO

CPU monitoring



The SQL Server Experts
Discover. Optimise. Mentor.



MEMORY MONITORING



The SQL Server Experts
Discover. Optimise. Mentor.

MEMORY MONITORING

Why has monitoring memory has changed?

- Demand based memory allocation is now common
- *Each VM can have as much as it needs **while it can***



- “Physical memory” can be added dynamically
- Reclamation methods are transparent to the OS and apps
- *As a result Task Manager and Perfmon are **misleading***

MEMORY MONITORING

Hyper-V Dynamic Memory methods

- **Enlightened Memory Addition** adds memory to Windows
- **Ballooning Driver** reclaims memory from Windows by allocating itself non-pageable memory

SQL Server and Dynamic Memory



- On startup **Hot-Add Memory** support is detected
- **Buffer Pool VAS** is set to **16x** Startup Memory
- **Buffer Pool Committed Size** is dynamically limited to available physical memory

MEMORY MONITORING

Monitoring SQL Server with Dynamic Memory

- What can **grow**, *and shrink*?
- **2008**: Buffer Pool **Denali**: Total memory
- Use `sys.dm_os_sys_info`
and `sys.dm_os_sys_memory`
- Compare BPool Target as
a % of Physical Memory

A background image showing a close-up of a hand typing on a white computer keyboard. The image is slightly blurred and has a dark, semi-transparent overlay.

DEMO

Memory monitoring



The SQL Server Experts
Discover. Optimise. Mentor.

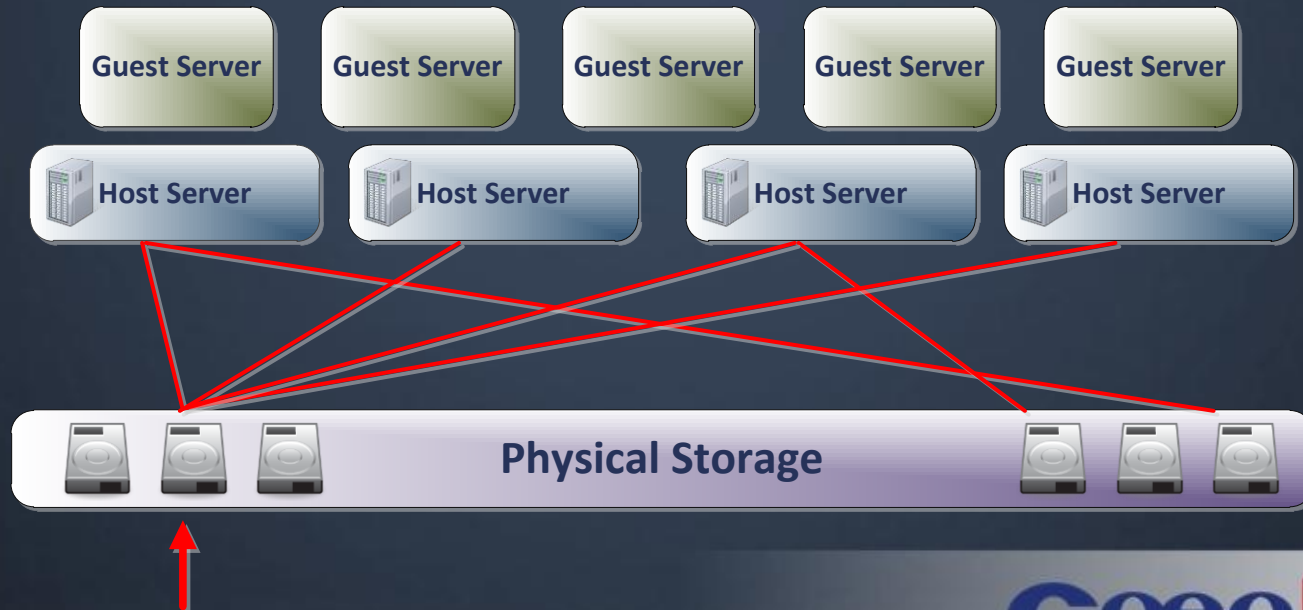


STORAGE MONITORING

STORAGE MONITORING

Very similar to physical storage monitoring

- IO throttling by the hypervisor is only just available
- However, hypervisors add a new layer of contention



STORAGE MONITORING

Warning signs from within SQL Server

- Traditional **PAGEIOLATCH_xx** wait stat is a warning but there maybe other causes
- High disk latencies from `sys.dm_io_virtual_file_stats`



Latency targets, even when virtualised:

Data files: < 20ms

Log files: < 10ms

STORAGE MONITORING

Monitoring within the guest and host servers

- Windows **logical** drive counters within the **guest**
- Windows **physical** drive counters within the **host**

Monitoring at the storage layer

- % of the **total** HBA and storage controller **capacity** in use



A background image showing a close-up of a hand typing on a white computer keyboard. The image is slightly blurred and has a dark, semi-transparent overlay.

DEMO

Storage monitoring



The SQL Server Experts
Discover. Optimise. Mentor.

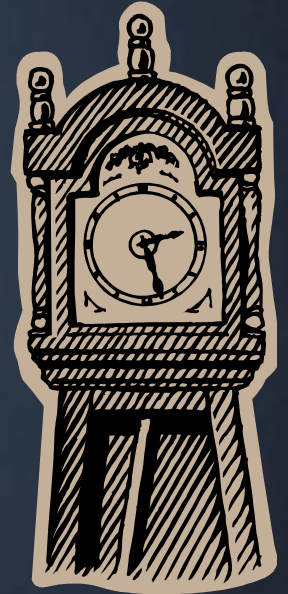
BEST PRACTICES



BEST PRACTICES

Look at the complete picture

- Compliment real time performance data
- Use known SQL Server metrics:
 - Backup times – calculate MB/s
 - ETL task runtimes – CPU availability



BEST PRACTICES

What to monitor and where

- **SQL Server** – Wait stats and Buffer Pool size
- **Guest OS** – % logical CPU, “Physical MB”, Available MB
- **Host OS** – % physical CPU, Physical IO
- **Storage Layer** – Consolidated HBA and Controller %'s



SUMMARY

- ☒ Wait Stats are good warning signs
- ☒ Monitor CPU at the Guest and Host
- ☒ Understand dynamic memory models
- ☒ Use job run times as benchmarks