Twitter: @gavinpayneuk

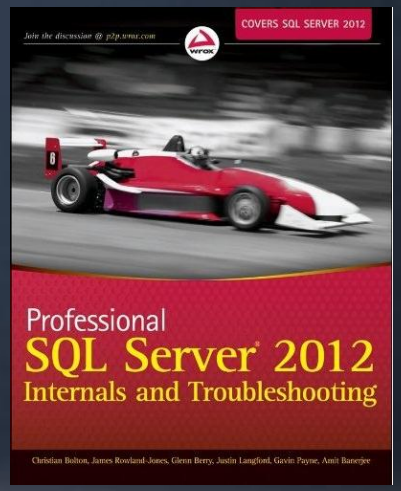
gavin@coeo.com

SQL SERVER 2012 NUMA INTERNALS

Gavin Payne – Senior Consultant

Microsoft Certified Master

http://sqlblogcasts.com/blogs/gavinpayneuk/



SESSION OBJECTIVES

- **Level 300 Internals Overview**
- Confirm what NUMA is
- How and why NUMA is used by SQL Server
- Troubleshooting considerations

ACKNOWLEDGEMENTS

- **Guy Bowerman** – SQLOS Program Manager
- **Thomas Kejser**
- **Jimmy May**
  @aspiringgeek

INTRODUCTION TO NUMA

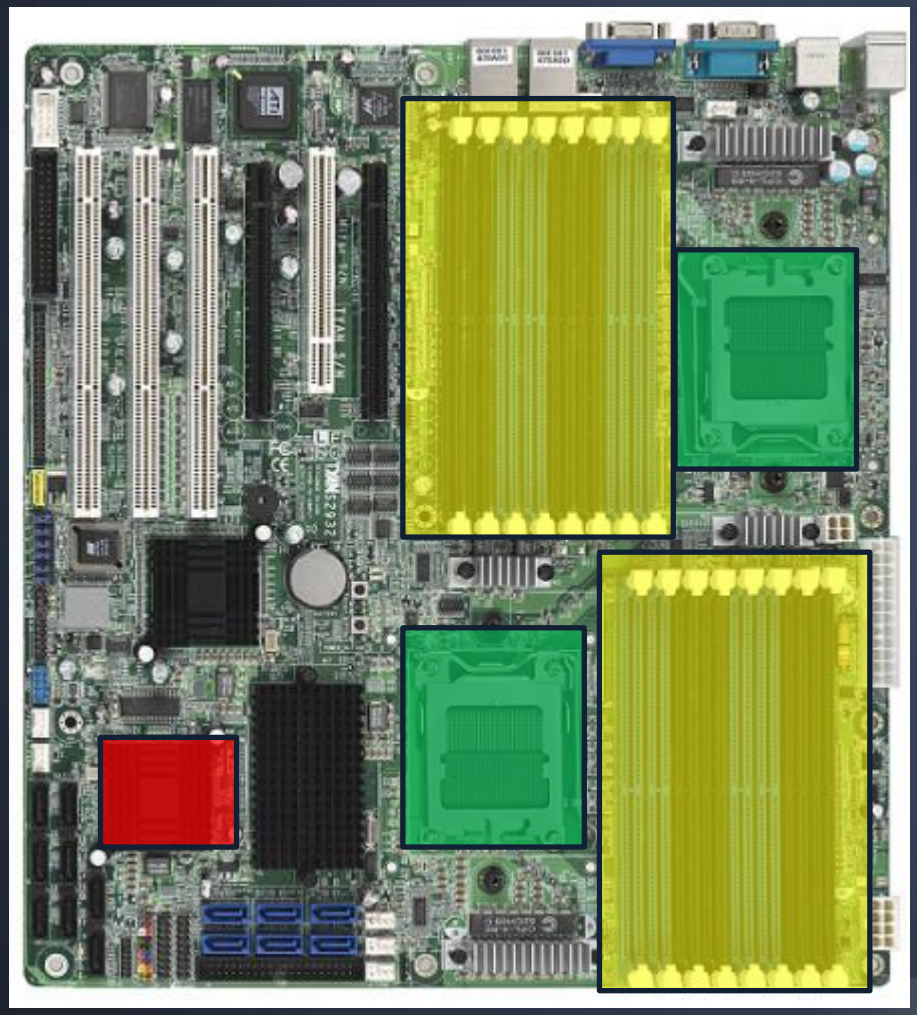
WHAT IS NUMA?

Non-Uniform **M**emory **A**rchitecture is a hardware design that improves server scalability by removing motherboard bottlenecks

*Physical memory becomes either **local** or **remote** to a specific processor*

*Accessing **remote** memory is slower than **local** memory*

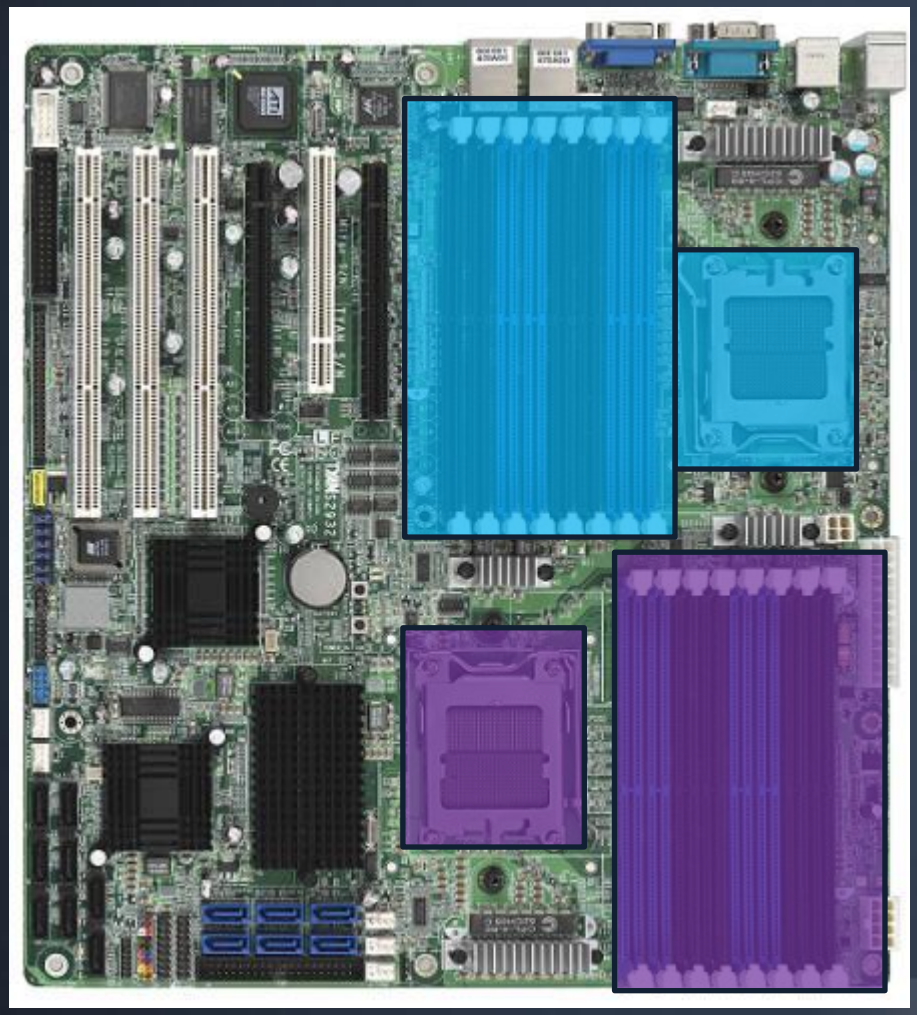
WHAT NUMA ISN'T



SMP

- Both CPUs have direct access to all memory
- Both share a Front Side Bus

WHAT IS NUMA?



NUMA

- Each node (or CPU) has its own memory bank
- Memory access management is now distributed

LOCAL VS. REMOTE MEMORY

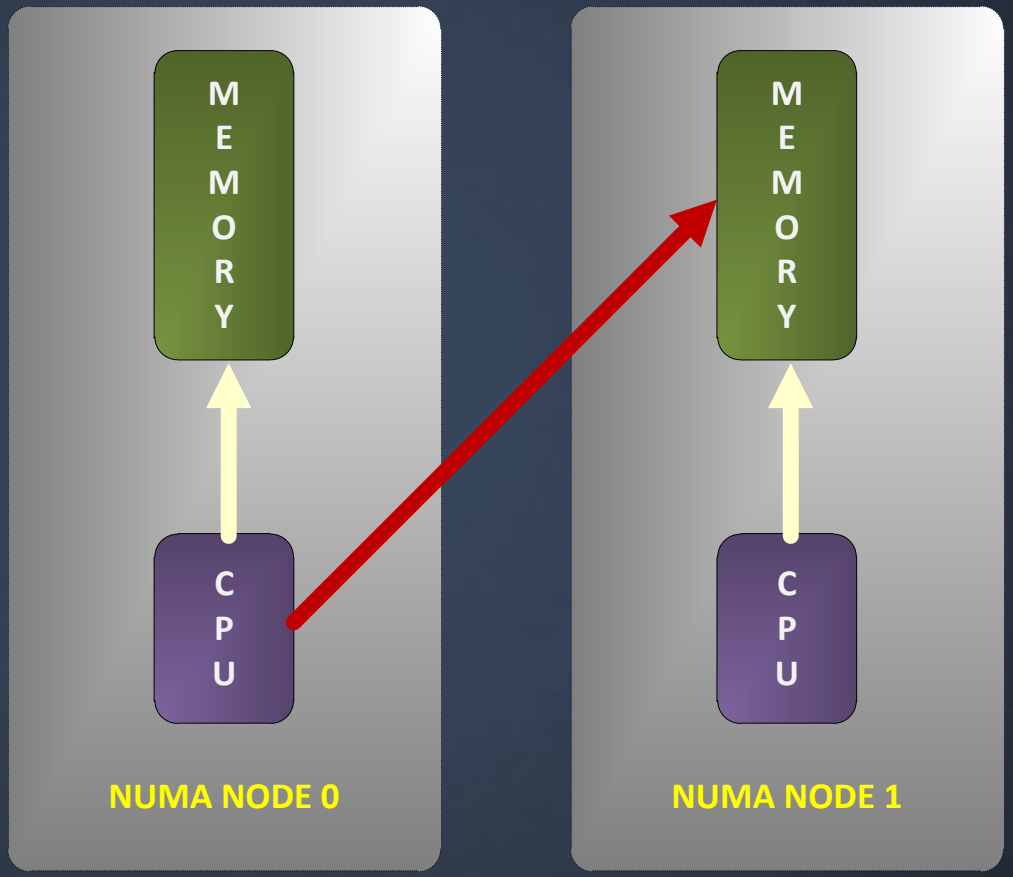
Local Access

Minimum latency

(100-200 CPU cycles)

Remote Access

Increased latency



MEASURING REMOTE ACCESS LATENCY

Sysinternals tool - coreinfo.exe

SQLCat Lab Server

HP DL980G7 with 8 x 8 core CPUs

```
Logical Processor to NUMA Node Map:

********------------------------------------------------------  NUMA Node 0
--------********----------------------------------------------  NUMA Node 1
----------------********------------------------------------  NUMA Node 2
------------------------********--------------------------------  NUMA Node 3
--------------------------------********------------------------  NUMA Node 4
----------------------------------------********----------------  NUMA Node 5
------------------------------------------------********--------  NUMA Node 6
----------------------------------------------------------********  NUMA Node 7
```

Coeo | The SQL Server Experts
Discover. Optimise. Mentor.

MEASURING REMOTE ACCESS LATENCY

Sysinternals tool - coreinfo.exe

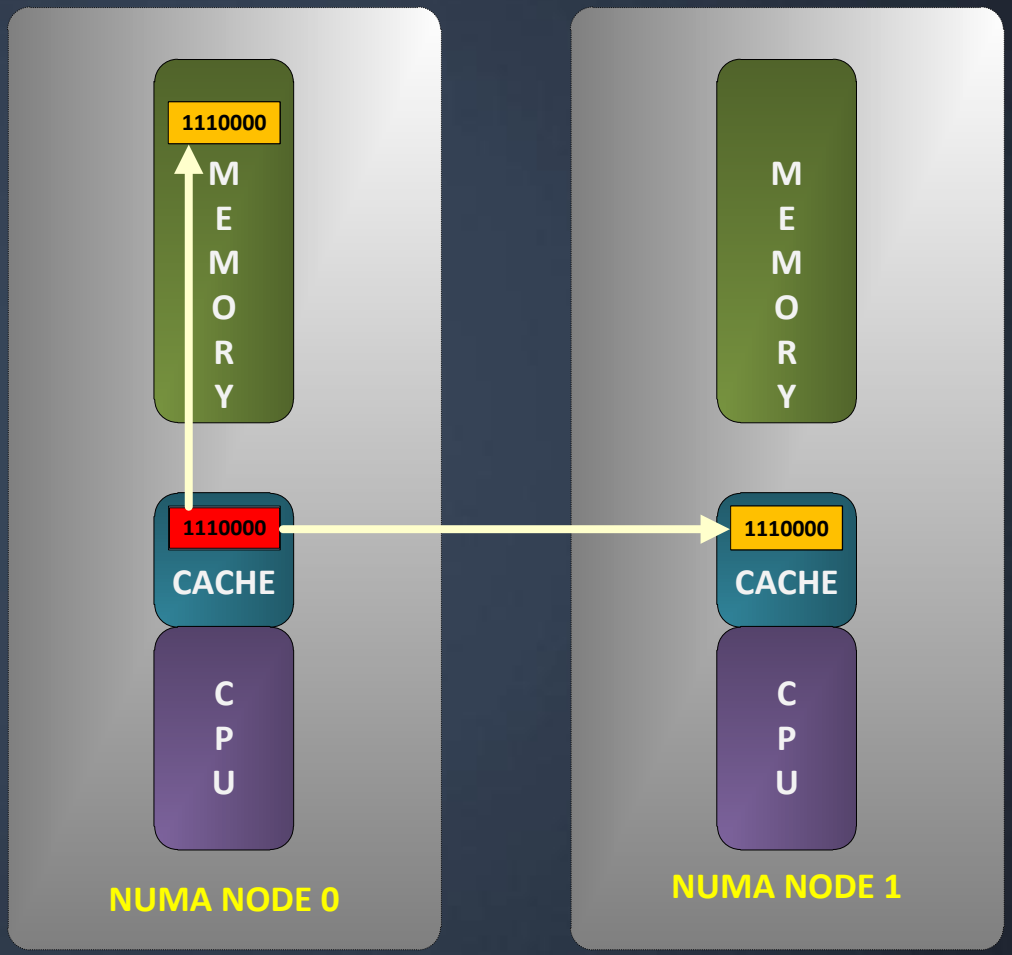
Example of Approximate Cross-NUMA Node Access Cost:

NODE	0	1	2	3	4	5	6	7
0	1.0	1.5	2.3	2.7	2.7	2.7	2.7	2.7
1	1.5	1.0	2.7	2.7	3.3	3.3	3.1	1.3
2	2.3	2.3	1.0	2.2	3.9	3.9	3.8	2.7
3	2.7	2.7	1.5	1.0	3.3	3.3	3.1	2.7
4	3.6	3.6	2.7	2.7	1.0	2.5	3.3	2.4
5	3.3	3.3	3.3	3.3	1.5	1.0	2.6	2.3
6	4.0	3.9	4.0	3.9	2.3	2.3	1.0	1.5
7	3.4	3.4	3.3	3.3	2.7	2.7	1.5	1.0

CACHE COHERENT NUMA

Changes made in one node's cache are instantly reflected anywhere else the data is in cache

Can itself introduce an overhead



SQL SERVER AND NUMA

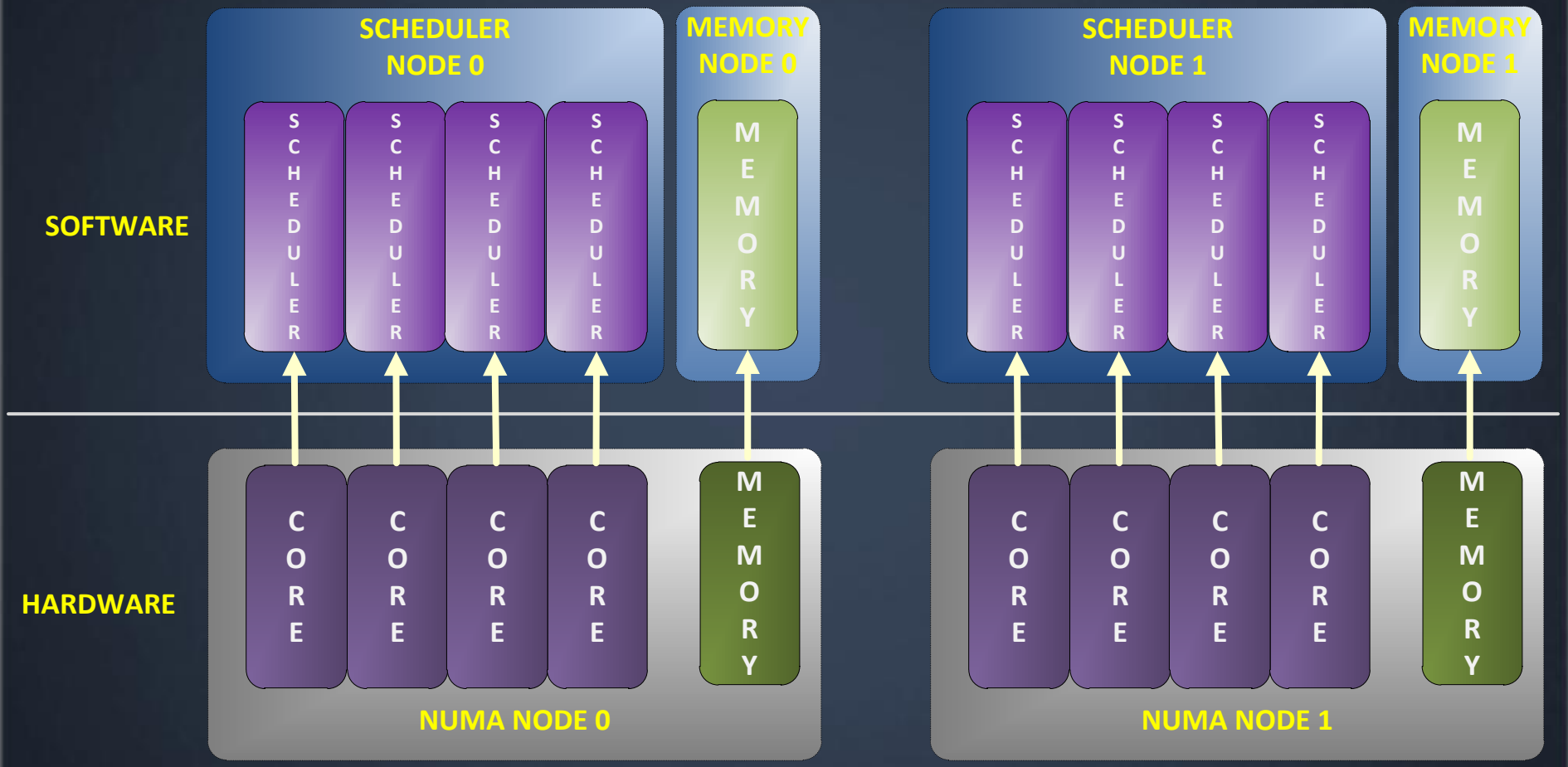
Why does it matter to SQL Server?

- Limited optimisations in SQL Server 2000 SP4
- Central to SQLOS's memory support in 2005
- What about SQL Server 2008 and 2012?

SQL SERVER INTERNALS

coeo
The SQL Server Experts
Discover. Optimise. Mentor.

INTERNAL SQL SERVER NODES



What about HyperThreading?

Coeo
The SQL Server Experts
Discover. Optimise. Mentor.

SQLOS SCHEDULER NODES

sys.dm_os_schedulers

scheduler_id	cpu_id	parent_node_id	status
0	0	0	VISIBLE ONLINE
1	1	0	VISIBLE ONLINE
2	2	1	VISIBLE ONLINE
3	3	1	VISIBLE ONLINE
1048576	0	64	VISIBLE ONLINE (DAC)

sys.dm_os_nodes

node_id	dec-to-bin mask
0	0000000011111111
1	1111111100000000

Node configuration: node 1: CPU mask: 0x000000000000ff00:0 Active CPU mask: 0x000000000000ff00:0. This message provides a description of the NUMA configuration for this computer.

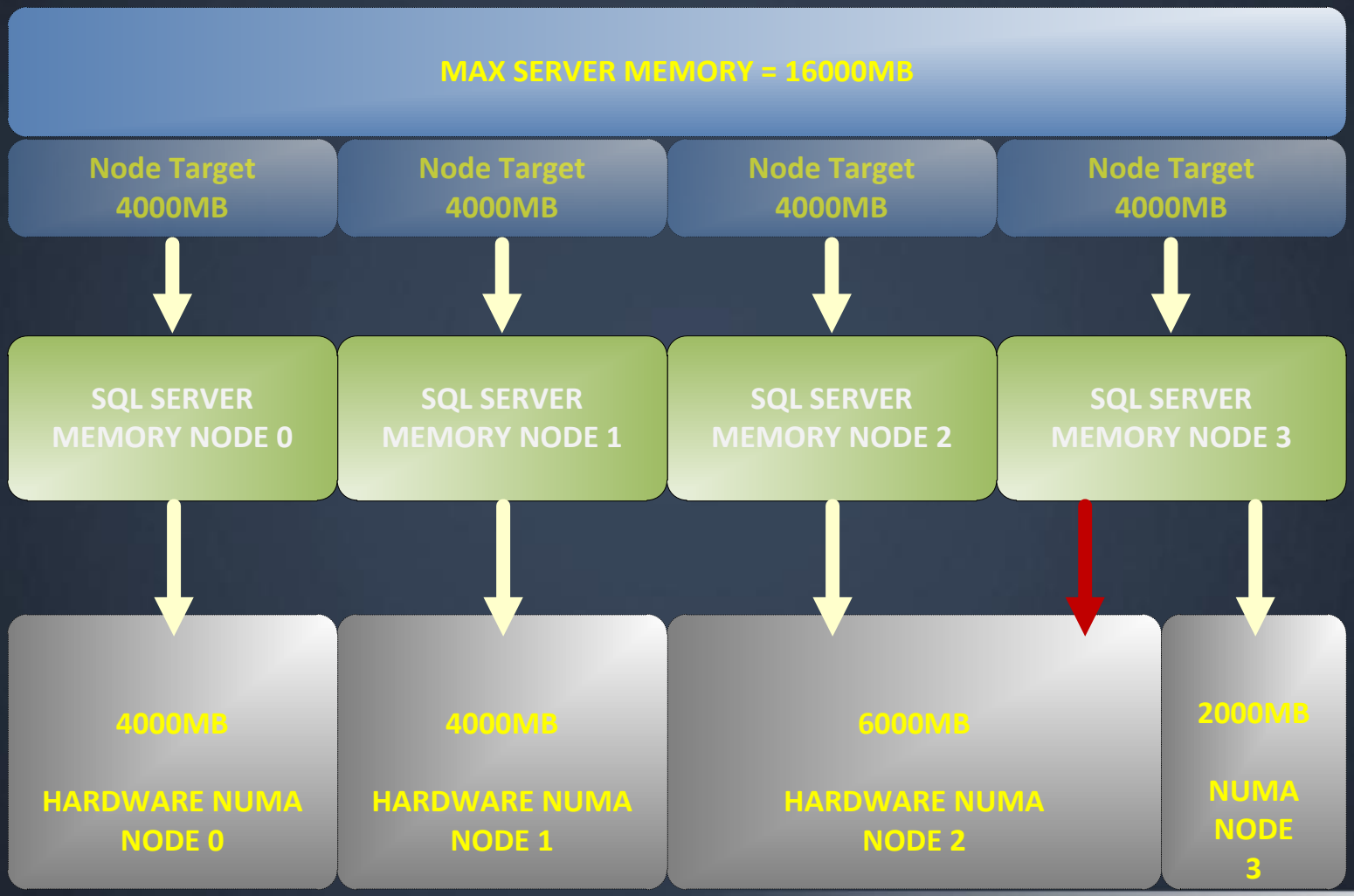
Node configuration: node 0: CPU mask: 0x00000000000000ff:0 Active CPU mask: 0x00000000000000ff:0. This message provides a description of the NUMA configuration for this computer.

Coeo The SQL Server Experts Discover. Optimise. Mentor.

SQL SERVER NUMA MEMORY INTERNALS

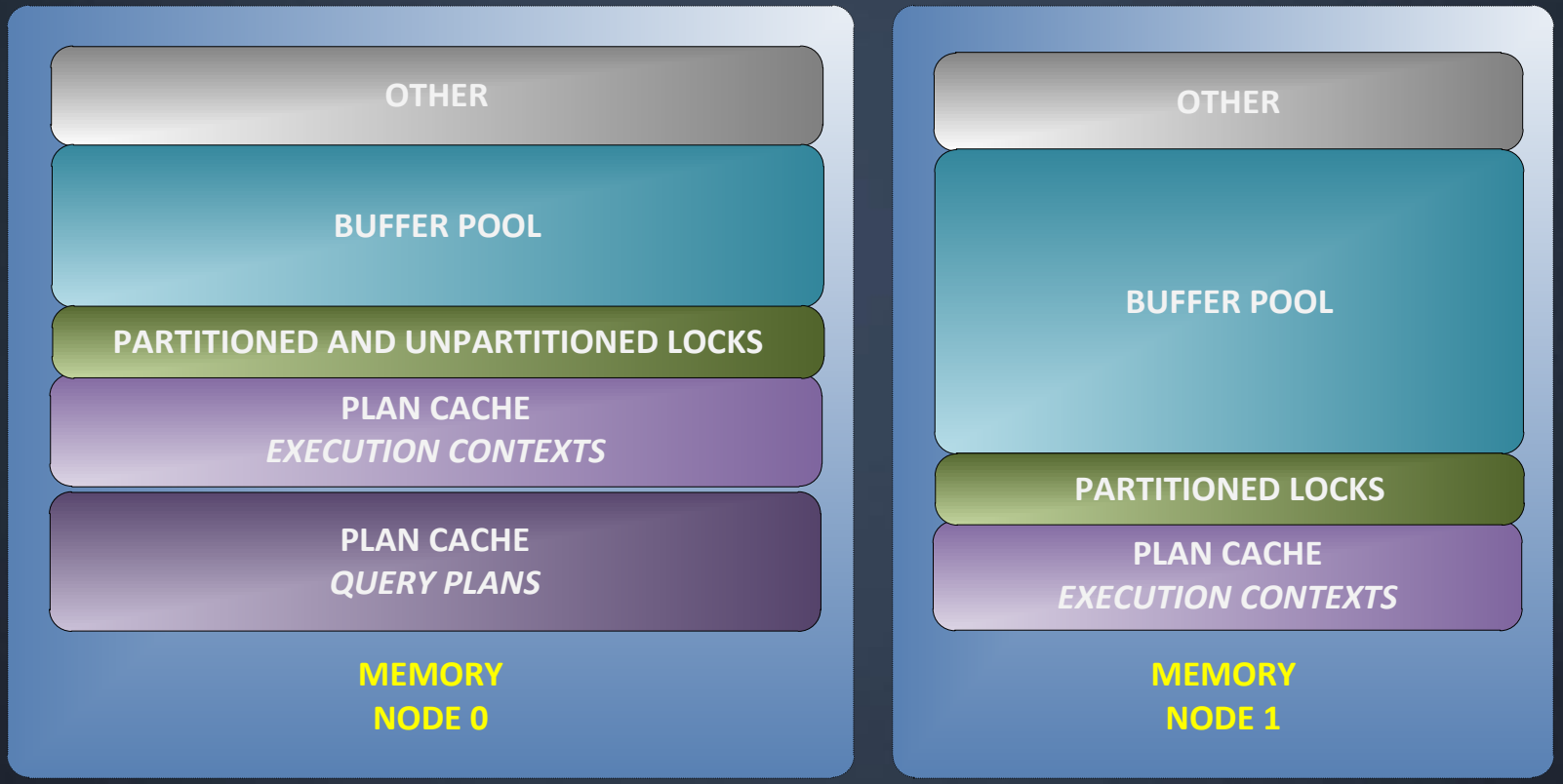
- SQL Server memory nodes map directly to hardware NUMA nodes
- SQL Server's objective is to reduce the need for remote memory access
- A task's memory objects get created within the same NUMA node wherever possible

MAX SERVER MEMORY



SQL SERVER NUMA MEMORY INTERNALS

- What gets stored in a memory node?



MONITORING SQLOS MEMORY NODES

sys.dm_os_memory_nodes

memory_node_id	virtual_address_space_committed_kb
0	958336
1	86852
64 (DAC)	20

Perfmon

Color	Scale	Counter	Instance	Parent	Object	Computer
	0.00001	Database pages	000	---	MSSQL$INST1:Buffer Node	\\COEONUMALAB
	0.00001	Database pages	001	---	MSSQL$INST1:Buffer Node	\\COEONUMALAB
	0.1	Page life expectancy	000	---	MSSQL$INST1:Buffer Node	\\COEONUMALAB
	0.1	Page life expectancy	001	---	MSSQL$INST1:Buffer Node	\\COEONUMALAB
	0.000001	Target Node Memory (KB)	000	---	MSSQL$INST1:Memory Node	\\COEONUMALAB
	0.000001	Target Node Memory (KB)	001	---	MSSQL$INST1:Memory Node	\\COEONUMALAB
	0.000001	Total Node Memory (KB)	000	---	MSSQL$INST1:Memory Node	\\COEONUMALAB
	0.000001	Total Node Memory (KB)	001	---	MSSQL$INST1:Memory Node	\\COEONUMALAB

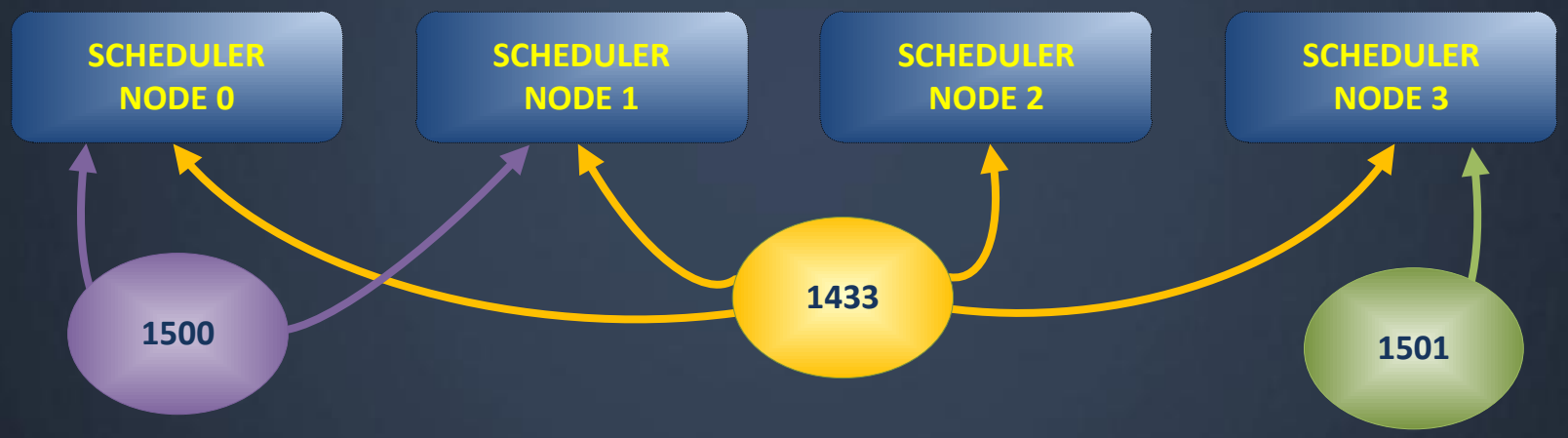
INTERNAL PROCESSES WITH NUMA

Consider each NUMA node as a mini-instance

- IO Completion Port
- Schedule Monitor
- Resource Monitor
- Lazywriter controlled by Checkpoint
- **But only ever a single Log Manager**

USER CONNECTIONS

- Connections are bound to a specific node on creation
- The node is assigned on a round robin basis



- TCP/IP ports can be used to direct specific workloads to specific NUMA nodes

QUERY EXECUTION

Where do buffer pool pages come from?

Session A on Node 1

Session B on Node 2

```
select * from tblTest
```

MEMORY NODE 1

```
insert into tblTest
```

MEMORY NODE 1

MEMORY NODE 1

```
select * from tblTest
```

MEMORY NODE 2

```
insert into tblTest
```

MEMORY NODE 1

```
select * from tblTest
```

MEMORY NODE 2

Coeo The SQL Server Experts Discover. Optimise. Mentor.

PARALLEL QUERIES

- SQLOS's objective is to run all of a parallel query's threads in the same node
- Reducing remote memory lookups less important

- Controlling task or data page nodes are not considered
- Typically, a different node to both of these

- Parallel queries do not respect any TCP/IP port related NUMA node affinities

SOFT NUMA

Logical partitioning within a NUMA node

- Unique to SQL Server
- Why?

Partition workloads without NUMA hardware

- More TCP/IP port to scheduler mappings
- No additional lazywriters
- No influence over memory locality

NUMA CONSIDERATIONS

- Consider each node as a mini-instance
- How will it change your monitoring? (PLE etc)
- Install DIMMs evenly across NUMA nodes!

SUMMARY

- NUMA is fundamental to SQL Server's internal memory management
- You can't change how it works but you can know how it works
- Consider adding node_id to your DMV monitoring