

Locality- Optimized Resource Alignment

Using local memory to increase processing
efficiency of HP NUMA servers



HP Integrity – The Most Trusted. Always.



©2009 Hewlett-Packard Development Company, L.P.
The information contained herein is subject to change without notice.



Introducing the speaker

- Tom Wylegala works in the UNIX Systems Lab, where he is the Partner Programs Manager for multifaceted programs such as LORA.
- Tom has worked at HP for 25 years in the areas of manufacturing, system architecture, hardware development, research, and software development.



Locality-Optimized Resource Alignment

- Multifaceted program to give an **immediate and significant increase in processing efficiency** by tailoring HP-UX 11i v3 to take advantage of the locality characteristics of HP NUMA servers with **no change to applications** and **no explicit tuning by the user**.
- **Improve ease of use** by hiding details of platform structure.
- **Integrate with power management strategy** when locality domains match power management domains.

Locality-Optimized
Resource Alignment

LORA

Agenda

Background on NUMA platforms

Efficiency advantage of local memory

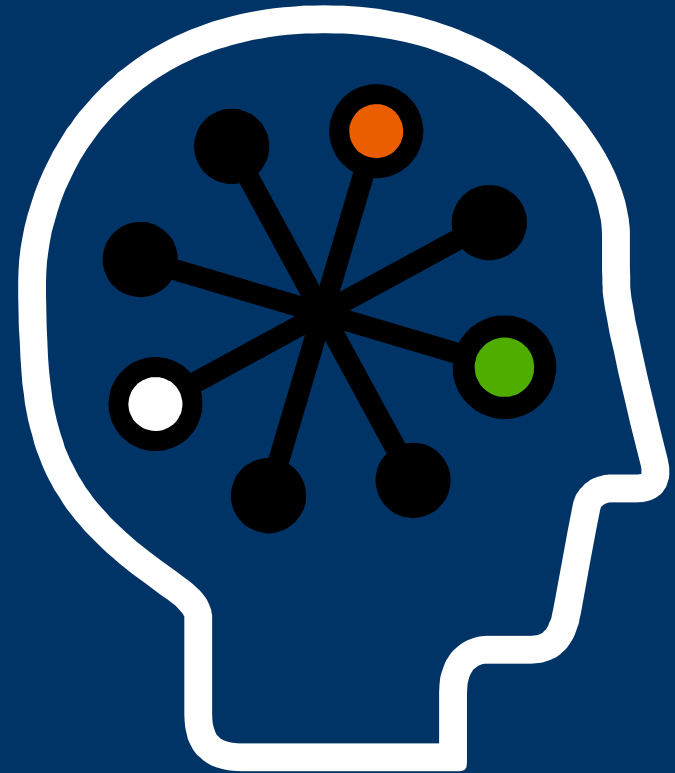
Facets of the LORA program

LORA with vPars

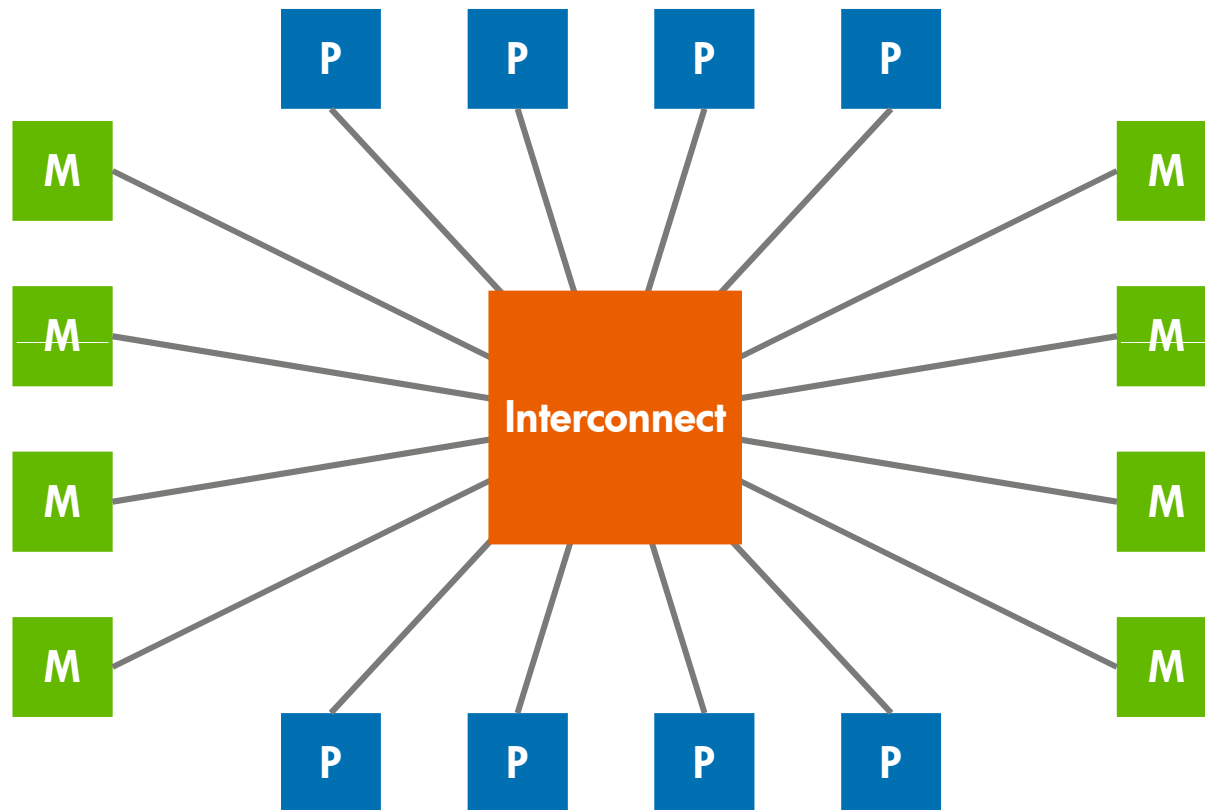
LORA with Integrity Virtual Machines

LORA in non-virtualized environments

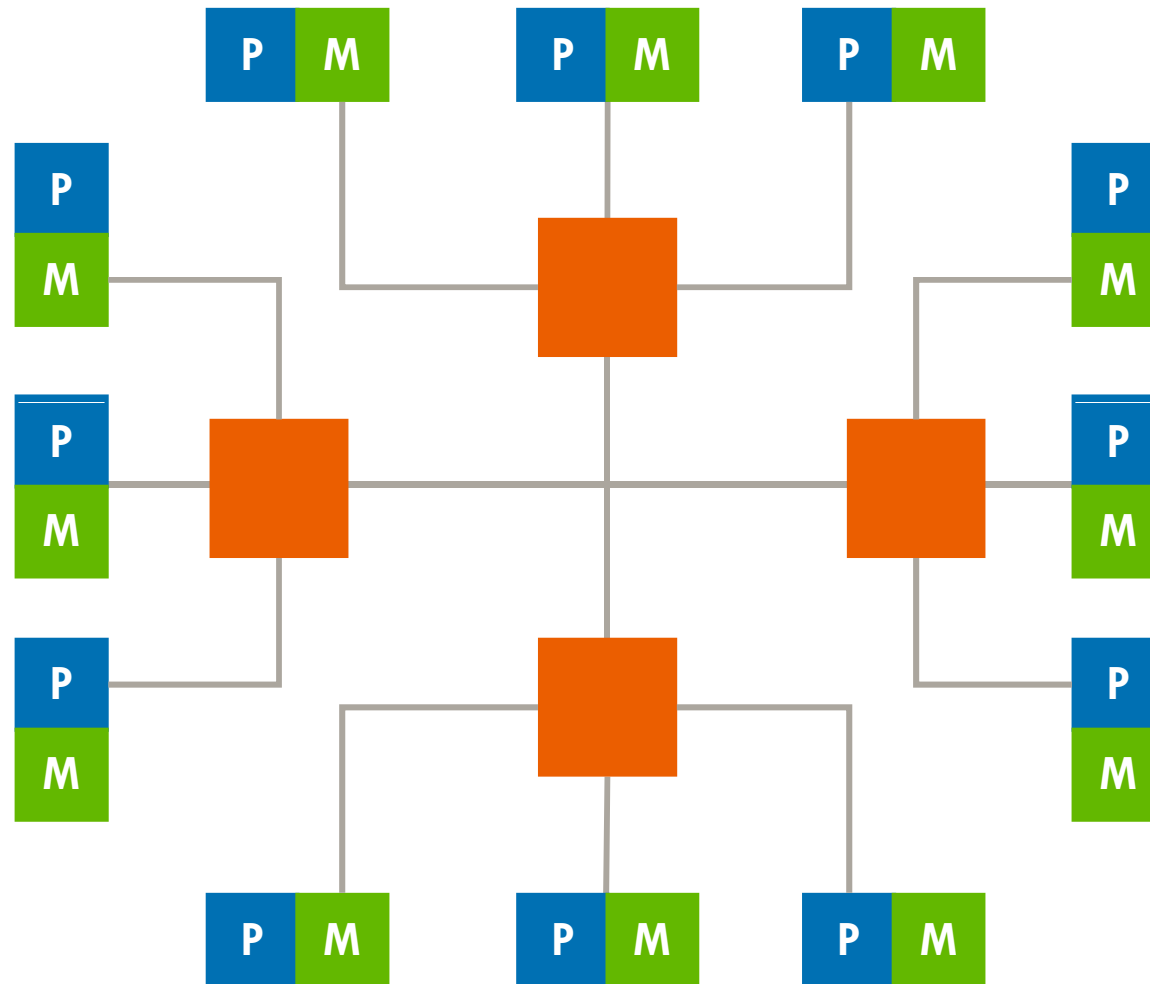
Background on NUMA platforms



Uniform Memory Access platform



Non-Uniform Memory Access platform



HP servers with NUMA structure

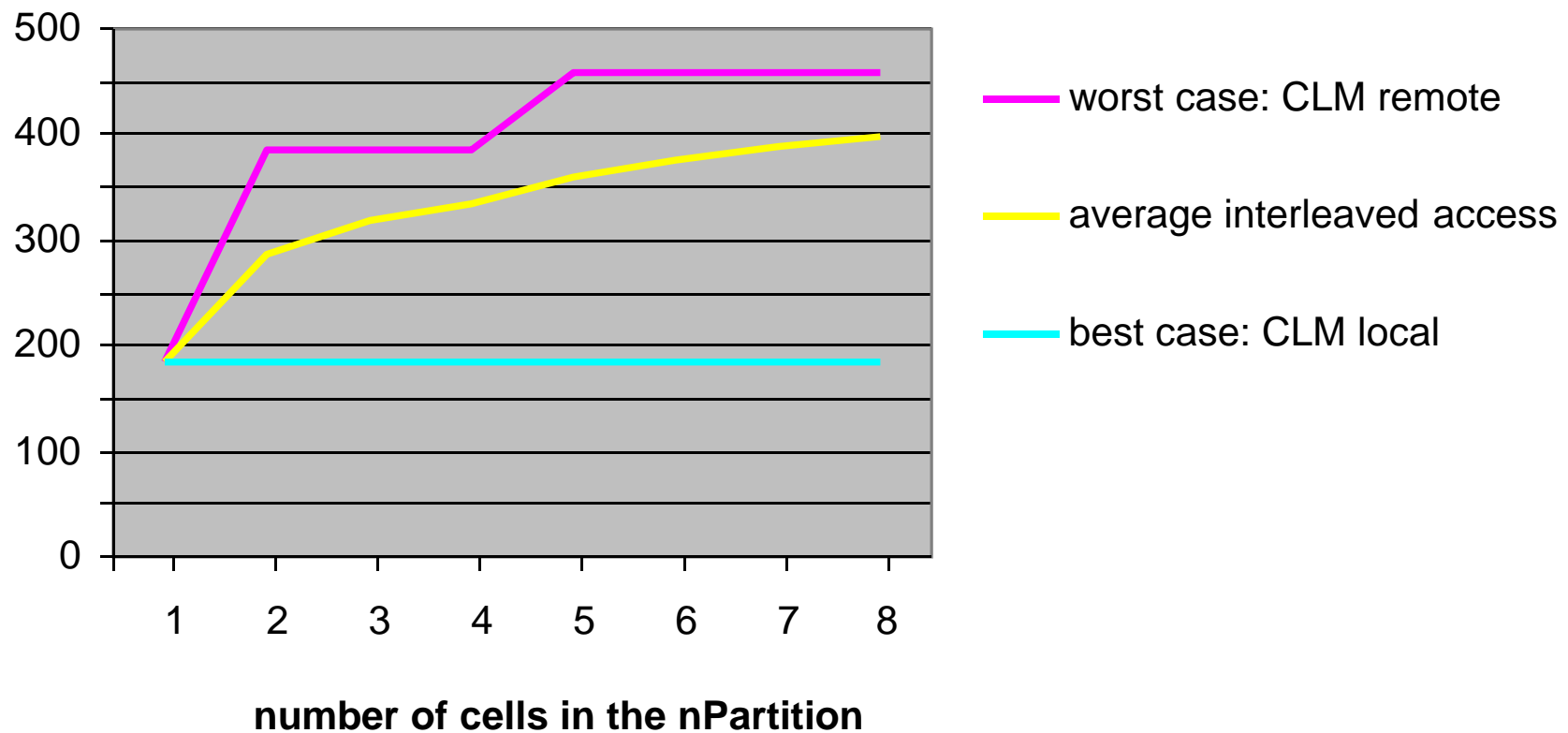
- All HP high-end and midrange servers have a Non-Uniform Memory Architecture (NUMA)
 - These are referred to as
 - Cell-based servers, cellular servers, cellular complex
 - Partitionable servers
 - Servers with sx1000 or sx2000
- The structure allows one product family to span the full performance range, permits division into independent and isolated nPartitions, and yields highly favorable price/performance metrics

Strategies for HP NUMA platforms

- Hide NUMA characteristics via memory interleaving
 - Cache line interleaving is provided by hardware
 - Interleaving is useful to distribute memory references even in uniform memory architecture platforms
 - Average memory access time is approximately uniform
 - On HP servers, it is called Interleaved Memory (**ILM**)
- Exploit memory locality for increased efficiency
 - Configure the platform with predominantly local memory
 - Processors enjoy fast access to local memory
 - On HP servers, it is called Cell Local Memory (**CLM**)

Memory access time by nPartition size for an HP NUMA server

Memory access time in nanoseconds



Impact of I/O locality

- Co-location of I/O adapters with processors and memory is an added bonus, but is a second-order effect
 - Operating system and application software employ a wide range of techniques to mask the latency of I/O operations
 - Disk access time is around 10 milliseconds; a hop through the interconnect fabric is around 100 nanoseconds
 - Processors in Integrity platforms can accomplish a lot of work in 100 nanoseconds
- The ideal configuration is to have I/O adapters distributed symmetrically across all localities, with redundant connection paths

Efficiency
advantage of
local memory



Effect of local memory on benchmark performance

Benchmark	CLM compared to ILM	%CLM
TPC-H	+10%	~80%
TPC-C high-end	+20%	~90%
TPC-C low-end	+50%	~90%
SAP SD – 2 tier	+53%	~80%
SPECjApp– multiple	+20%	~90%

- Data is for a 64-socket nPartition on Superdome running HP-UX 11i v3
- SPECjApp is a projection from a smaller configuration
- TPC-C low-end is an estimation of a 4-socket soft partition

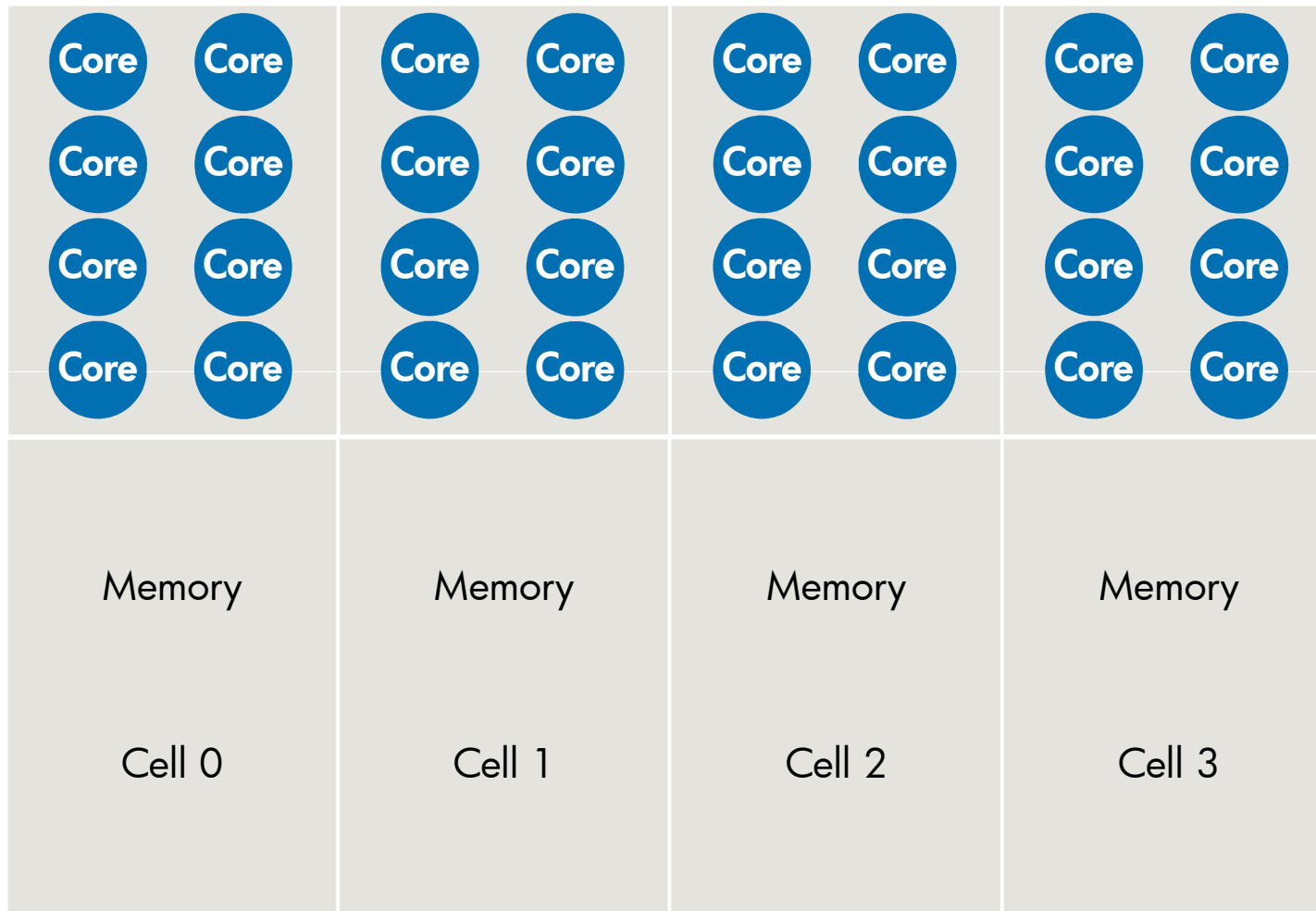


Performance experiment: SAP on vPars

- Measurement performed by the SAP team in the HP-UX Kernel Performance Section in early 2008
- SAP workload, mixture of benchmark scripts
- vPars configuration on rx8640 server
 - nPartition divided into 4 vPars instances of equal size
- Compared performance between
 - 100% Interleaved Memory (ILM)
 - 100% Cell Local Memory (CLM)



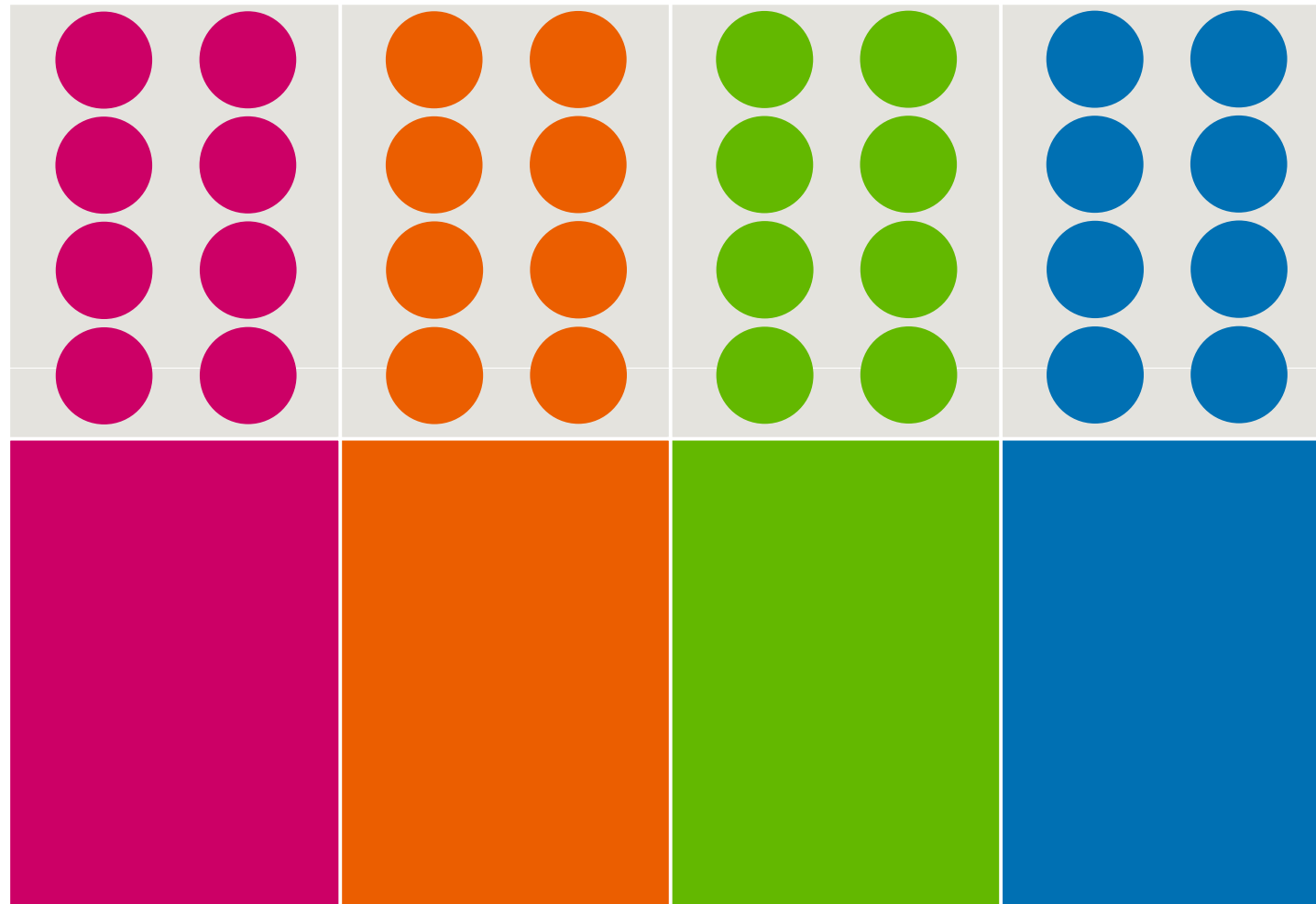
rx8640 hardware configuration



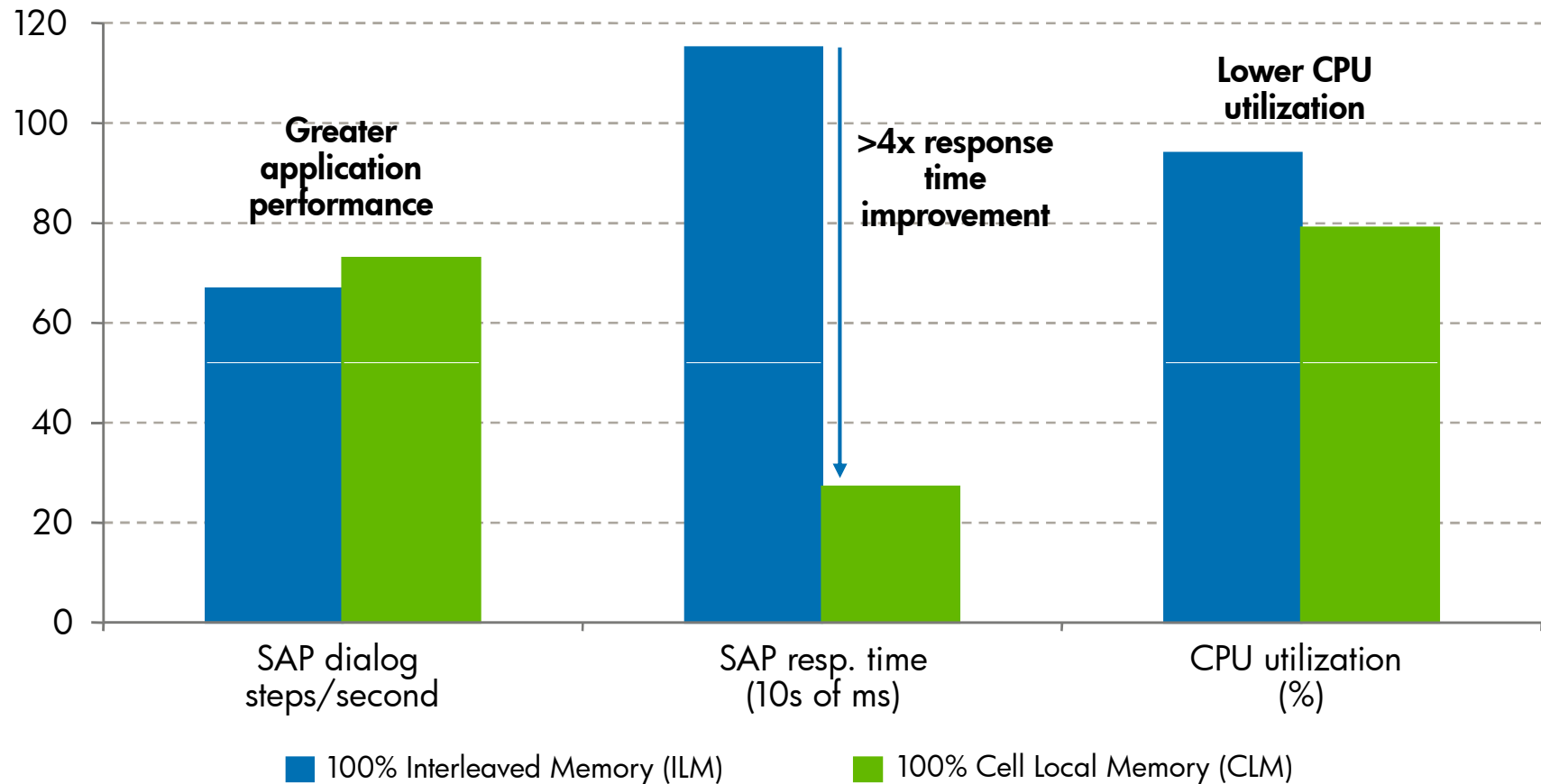
100% ILM vPars configuration



100% CLM vPars configuration



CLM gives greater performance efficiency



Alternatively, 6 CPUs using CLM = 8 CPUs using ILM



Realizing increased efficiency as performance gain or cost reduction

Adjusting resources for LORA compared to ILM

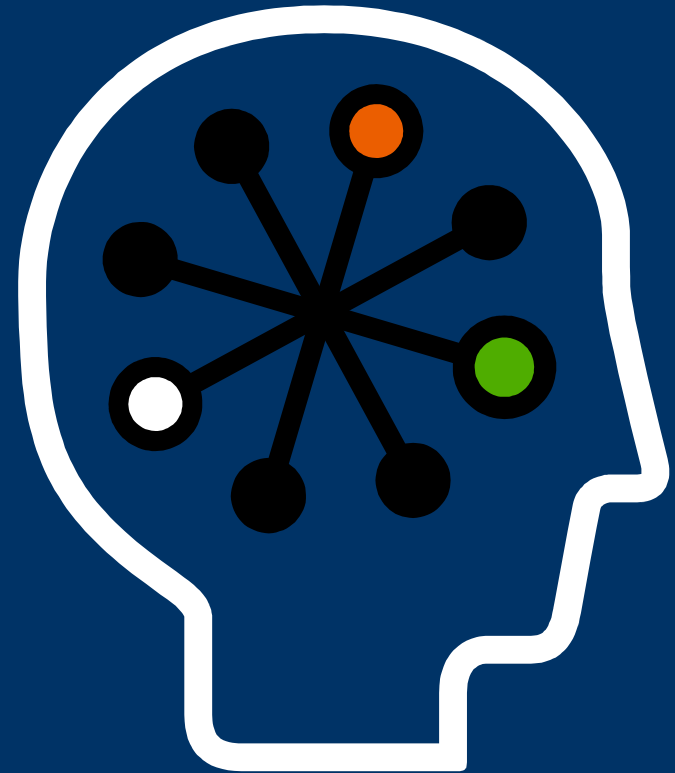
Processor type	Number of cores	Core adjustment	Memory adjustment
Intel Itanium processor 9100 series	1 to 12	None	None
	13 to 24	Reduce by 10%	Increase by 5%
	25 to 48	Reduce by 20%	Increase by 10%
	49 to 64	Reduce by 25%	Increase by 15%
Dual-core (Montecito, Montvale)	1 to 24	None	None
	25 to 48	Reduce by 10%	Increase by 5%
	49 to 96	Reduce by 20%	Increase by 10%
	97 to 128	Reduce by 25%	Increase by 15%

Note

If memory utilization is below 75%, as is often the case, it is not necessary to increase the amount of memory at all.



Facets of the LORA program



LORA consists of several elements

- Reference architecture
- Configuration rules and tuning recommendations
- New HP-UX 11i v3 commands
- Enhancements to vPars
- Enhancements to Integrity Virtual Machines
- Enhancements to performance analysis tools (coming soon)
- Integration with advanced power controls (coming soon)



LORA reference architecture

- Describes the design center for the locality optimizations. Compliant configurations will gain maximum performance benefit.
- Criteria
 - Integrity cellular platform
 - Running HP-UX 11i v3 Update 3 or later
 - Configured with 7/8 Local Memory
 - Dynamic platform reallocations limited to 30% variability in computing resources
 - Running vPars, Integrity Virtual Machines, or SAP, or other workloads known to exhibit strong locality



LORA configuration rules

- Configure the minimum number of distinct localities needed to supply the processor, memory, and I/O resources to provision the workload
- Allocate memory in the ratio 7/8 local memory to 1/8 interleaved memory
- Distribute all resources symmetrically across all localities
- Align the processors executing a workload with the memory that they access

Easier nPartition creation

- A new command, `parconfig`, makes creation of nPartitions easier
- Allows number of cells in nPartition to be specified, rather than naming each cell by its identifier as in `parcreate`
 - When the option to create a LORA partition is specified, it is created according to the LORA configuration guidelines

Workloads well-suited to LORA

- LORA yields best results for workloads that exhibit strong locality of memory reference pattern
- SAP
 - Deploy multiple SAP instances so that each one fits within a single locality
- Java
 - Deploy multiple Java Virtual Machines so that each one fits within a single locality
- vPars
 - Creating vPars instances that are well aligned guarantees favorable memory reference pattern
- Integrity Virtual Machines



LORA evolution over time

- LORA first introduced with September 2008 update to HP-UX 11i v3
 - Recommended for SAP, Java, and static vPars deployments
- Enhancements added in March 2009 update
 - Easier vPars configuration, also includes dynamic CPU migration
 - New `parconfig` command
- More enhancements planned for September 2009
 - Automatic workload placement

LORA with vPars



vPars partitioning model is well-suited to LORA

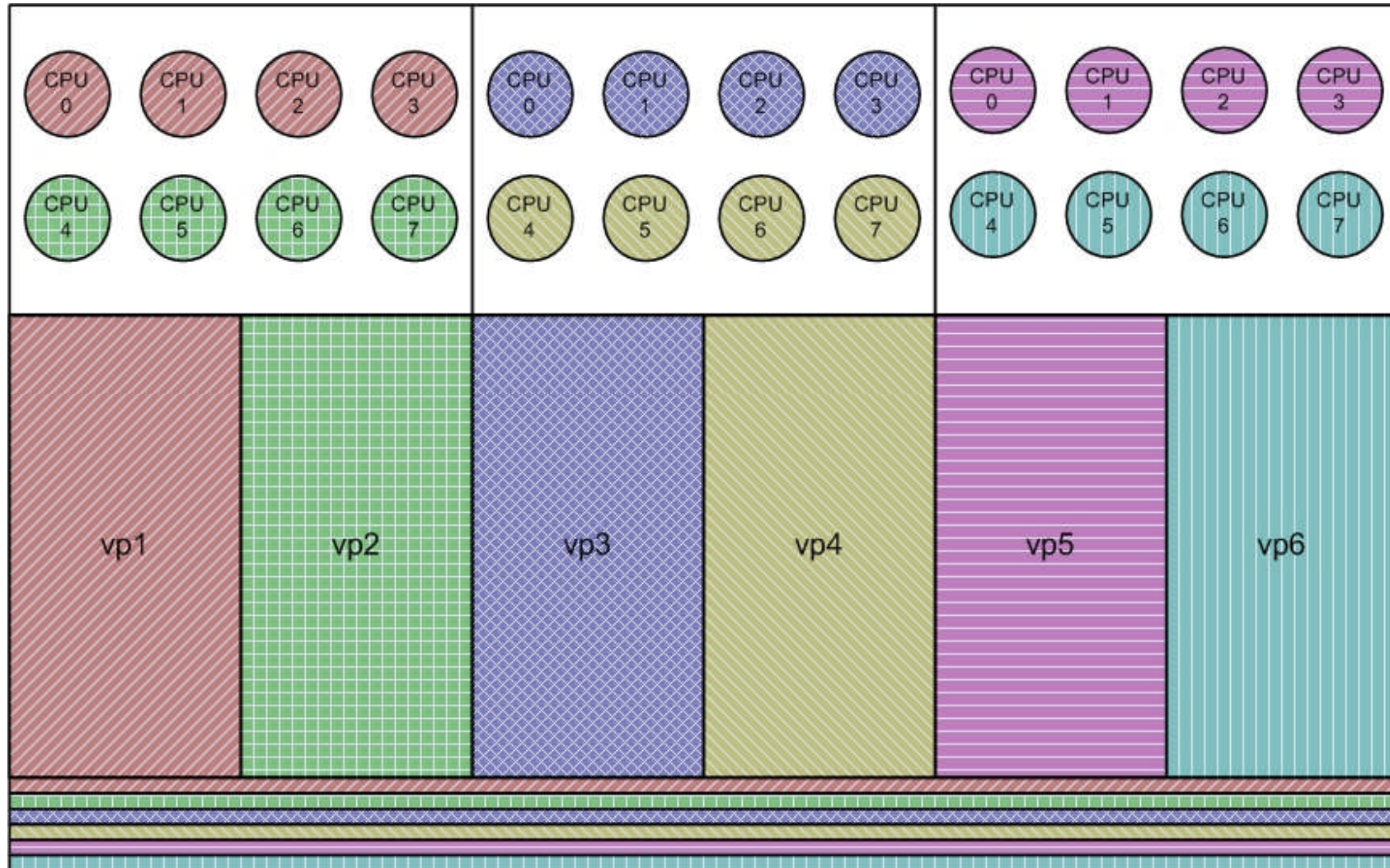
- Especially version A.05.05, delivered in the March 2009 update to HP-UX 11i v3
 - Earlier versions of vPars may also benefit from local memory, but require more user tuning, especially in dynamic platform scenarios

Using LORA with vPars

- Configure hosting nPartition with 7/8 local memory
- Configure each vPars instance with 7/8 local memory
 - Draw the local memory from the minimum number of cells
 - Specify cells for chosen memory at vPar creation time
 - Can specify I/O as well or instead
 - Don't specify cell assignments for the processor cores
- vPars monitor chooses processor cores that are close to the memory and/or I/O that was specified
 - Processor to memory affinity is preserved throughout processor migration operations



Example LORA vPars configuration



Commands for vPars example

Use `vparcreate` to establish the processor and memory allocations

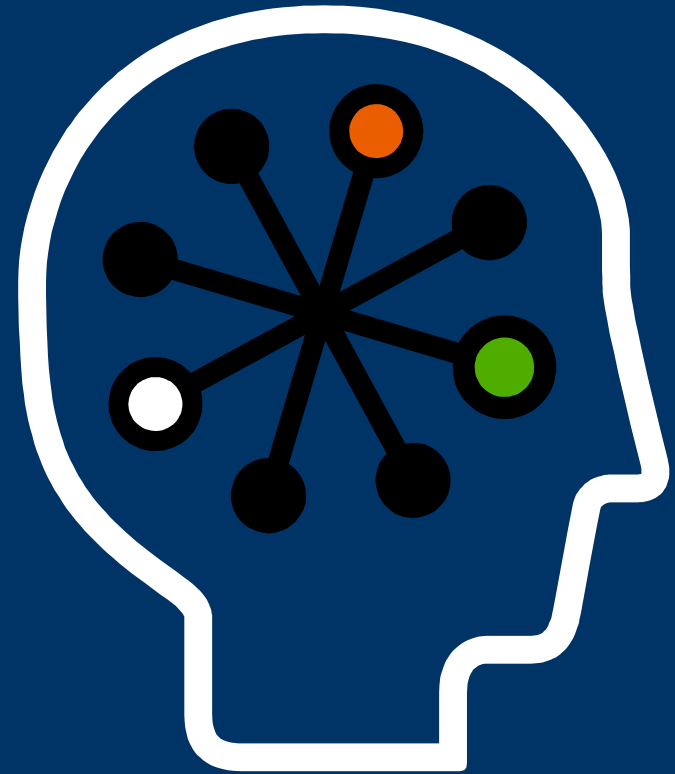
```
vparcreate -p vp1 -a cpu::4 -a mem::4096 -a cell:1:mem::28672  
vparcreate -p vp2 -a cpu::4 -a mem::4096 -a cell:1:mem::28672  
vparcreate -p vp3 -a cpu::4 -a mem::4096 -a cell:2:mem::28672  
vparcreate -p vp4 -a cpu::4 -a mem::4096 -a cell:2:mem::28672  
vparcreate -p vp5 -a cpu::4 -a mem::4096 -a cell:3:mem::28672  
vparcreate -p vp6 -a cpu::4 -a mem::4096 -a cell:3:mem::28672
```



When to use LORA with vPars

- vPars deployments will nearly always perform better in LORA configuration than with interleaved memory, especially when the nPartition is large
- Troublesome cases
 - Extreme asymmetry in vPars instances
 - One instance with 3 processors and 400 GB of memory; another instance with 20 processors and 64 GB of memory
 - Extreme dynamism
 - Continued creation and destruction of vPars instances that causes local memory to become fragmented
 - Memory needed by vPars instances relatively small in comparison to platform granule size

LORA with Integrity Virtual Machines



Integrity Virtual Machines is tuned to exploit local memory

- Use HPVM V4.00 or later so the host platform is HP-UX 11i v3, with Update 4 from March 2009 preferred

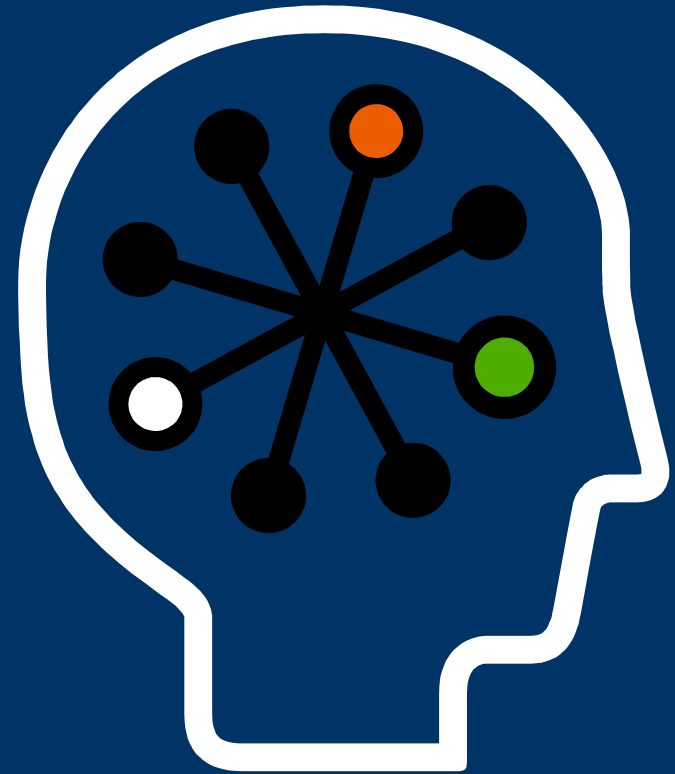


Using LORA with Integrity Virtual Machines

- Configure the host platform with 7/8 local memory
- Binding of virtual resources to physical resources is flexible and fluid
 - Performed automatically by the Platform Manager to gain best processor to memory affinity
 - Guest instances see a uniform memory environment



LORA in
non-virtualized
environments



Using LORA in non-virtualized environments

- Workload placement aligns the processor and memory resources used by an application
- Manual workload placement
 - Use the `mpsched` command
 - Create multiple processor sets
- Automatic workload placement
 - Available in a future version of LORA



SAP on LORA

- Size the SAP instances to fit within a single locality
 - Approximately 4 processor cores
 - Between 4 GB and 12 GB of memory per core
 - Approximately 2000 Sales and Distribution (SD) users
- More details available in the white paper
 - Title: Using HP-UX 11i v3 features to enhance the performance and scalability of an HP Integrity Superdome SD64B server running SAP
 - URL: <http://h71028.www7.hp.com/ERC/downloads/4AA1-3728ENW.pdf>



Manual placement of SAP with LORA

- Use `mpsched` to place the SAP instances

```
mpsched -P PACKED -l 2 sapstart r3 D00
```

```
mpsched -P PACKED -l 2 sapstart r3 D01
```

```
mpsched -P PACKED -l 3 sapstart r3 D02
```

```
mpsched -P PACKED -l 3 sapstart r3 D03
```

- Compared to 100% interleaved memory, performance is 20% better on a 4 cell server – or the number of processor cores could be reduced by 20%

Java on LORA

- Size the Java Virtual Machine instances to fit within a single locality
 - Between 2 and 4 processor cores
 - Approximately 4GB of memory per JVM instance
- Use `mpsched` with packed launch policy
- Set `numa_policy` parameter to 3



Oracle on LORA

- Oracle Database Management System can operate in two modes on NUMA platforms
 - NUMA enabled and NUMA disabled
 - In NUMA mode, the Oracle application is sensitive to the processor and memory localities in the platform
 - Dynamic platform operations that cause new localities to appear or existing localities to vanish can cause Oracle to become unstable
 - An Oracle Metalink entry (761065.1) describes the situation and suggests remedies
 - Running Oracle with LORA may increase performance, especially for multiple instances of Oracle
 - Run Oracle with NUMA disabled when using 100% ILM

Comparison to Linux

- Linux generally configured with 100% CLM
- HP-UX 11i v3 can handle any mix of CLM and ILM
 - Benchmarks executed with approximately 90% CLM
- Representative TPC-C results

<u>System</u>	<u>tpmC</u>	<u>Price/tpmC</u>	<u>System Availability</u>	<u>Database</u>	<u>Operating System</u>
<u>HP Integrity Superdome-Itanium2/1.6GHz/24MB</u>	4,092,799	2.93 USD	08/06/07	Oracle Database 10g R2 Enterprise	HP-UX 11i v3
<u>PRIMEQUEST 580A 32p/64c</u>	2,382,032	3.76 USD	12/04/08	Oracle Database 10g R2 Enterprise	Red Hat Enterprise Linux 4 AS



When to use LORA with nPartitions

- LORA usually performs as well as or better than the 100% interleaved configuration
- Troublesome cases
 - Technical applications
 - Extremely large data sets, little spatial locality, demands maximum memory bandwidth
 - Database consuming the entire nPartition
 - Extremely large data sets, referenced globally by all processors
 - Workloads that create and destroy many short-lived processes
 - Many hundreds of processes per second

Summary

- LORA is a simple technique that can be used to gain an immediate and significant performance increase for many important workloads
 - SAP, Java, vPars will show strong benefits
 - Other workloads may also perform well
- Further LORA enhancements will expand the range of suitable applications and will make management and tuning easier



LORA: More information

- White paper: Locality-Optimized Resource Alignment <http://docs.hp.com/en/14655/ENW-LORA-TW.pdf>
- White paper: Using HP-UX 11i v3 features to enhance the performance and scalability of an HP Integrity Superdome SD64B server running SAP <http://h71028.www7.hp.com/ERC/downloads/4AA1-3728ENW.pdf>
- White paper: The Oracle database on HP Integrity servers <http://h20195.www2.hp.com/V2/GetPDF.aspx/4AA2-0547ENW.pdf>



Technology for better business outcomes

