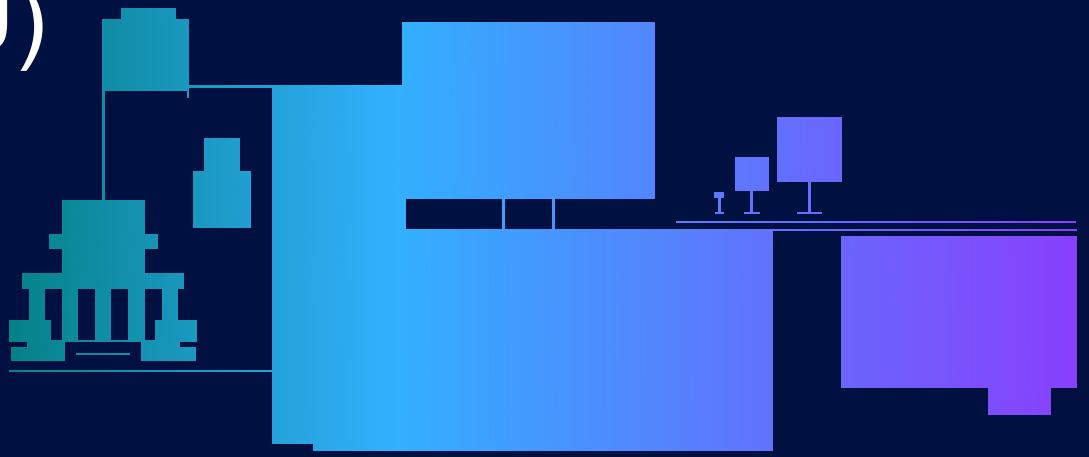
Oracle Database on IBM Power with AIX Best Practices (Part 1: Memory and CPU)



Alexander Hartmann
alexander.hartmann@de.ibm.com
IBM Systems

Slide deck provided by:
Ralf Schmidt-Dannert
dannert@us.ibm.com
IBM Advanced Technology Group

Agenda

AIX Configuration/Tuning for Oracle (Part 1)

- **Memory**
- **CPU**
- *I/O*
- *Network*
- *Miscellaneous*

The suggestions presented here are considered to be basic configuration “starting points” for general Oracle workloads

Customer workloads will vary

Ongoing performance monitoring and tuning is recommended to ensure that the configuration is optimal for your particular workload characteristics

Power9 / Power10 processor-based portfolio supports workloads of all sizes

On - Prem

Off - Prem

Power Scale-out

S922/S914/S924

AIX POWER9

Up to 24 cores, 4TB

Power Enterprise

Power E950

AIX POWER9

Up to 48 cores, 16TB

Power E980

Up to 192 cores, 64TB

Power E1080

AIX 10

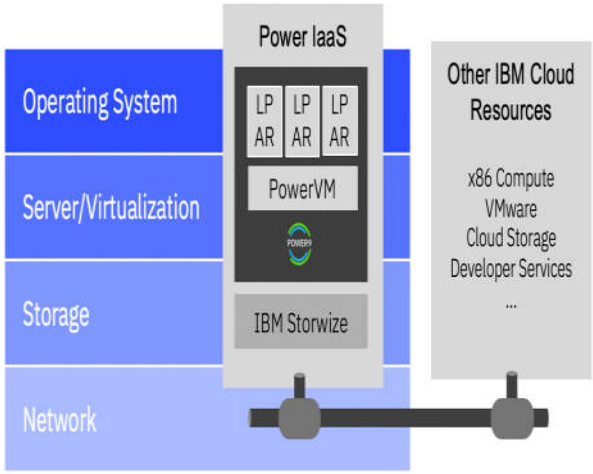
Up to 240 cores, 64TB

Capacity On Demand

Enhanced RAS

IBM Power Systems Private Cloud Solution with Dynamic Capacity (*Enterprise Pools 2.0*)

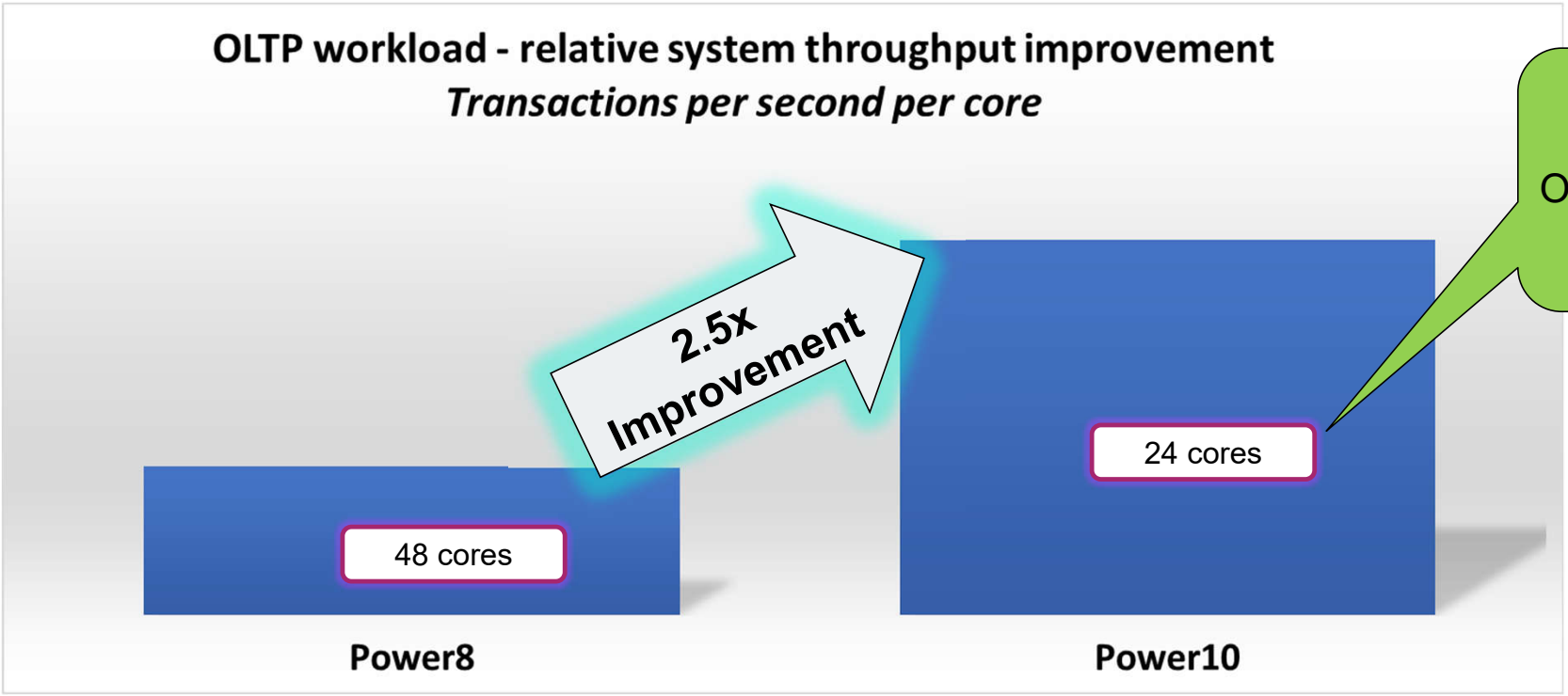
IBM Power Systems Virtual Server



Multi-tenant, self managed, Power compute as-a-service with consumption-based OpEx pricing; colocated and connected with the IBM Cloud

Mission Critical Workloads with Oracle database

Power8 to Power10 – Refresh benefit for OLTP workload



Source: IBM white paper "**Modernizing Oracle Database on IBM Power -** *Practical guidance on getting current!"* available at *https://www.ibm.com/support/pages/node/6485173*

Based on IBM internal OLTP workload performance testing. Your results may vary!

- OLTP workload simulates stock trading transactions and was configured to drive CPU usage with a 90/10 read/write transaction mix.
- Workload was scaled in both environments to utilize all available CPU resources and then transactions per second were measured.

	Power8	Power10
System Model	Power E870	Power E1080
Cores	48	24
Clock Speed	4024 MHz	4146 MHz
SMT (Default Values Used)	4	8
OS level	AIX72 TL5 SP2	AIX72 TL5 SP2
Oracle DB	11.2.0.4	11.2.0.4

Agenda

- Memory
- CPU
- I/O
- Network
- Miscellaneous

AIX Multiple Page Size Support

(**Small**)

- Pageable; can be dynamically coalesced to 64KB pages (pmsd process)
- Default page size for processes in AIX (private data, stack, binary text)
- AIX filesystem cache (JFS, JFS2, NFS)

(**Medium**)

- Pageable; can be dynamically broken up into 4KB pages
- Default kernel page size AIX 6.1 and later
- Default for Oracle SGA in Oracle 11g or later
- Configurable for Oracle binaries; Oracle 12.1+ uses 64KB by default

  (export LDR_CNTRL=DATAPSIZE=64K@TEXTPSIZE=64K@STACKPSIZE=64K oracle)

(**Large**)

- Pinned, not pageable; can be dynamically broken up into 64KB pages in specific circumstances
- Manual allocation required; no automatic creation of persistent pages
- Option to coalesce from 64KB pages for shared memory with AIX 7.2; temporary 16MB pages (DSO)
- If improperly configured, can contribute to severe system paging and kernel panics

Tip: Read up on new vmo parameters pgz_* in AIX 7.2 TL2.

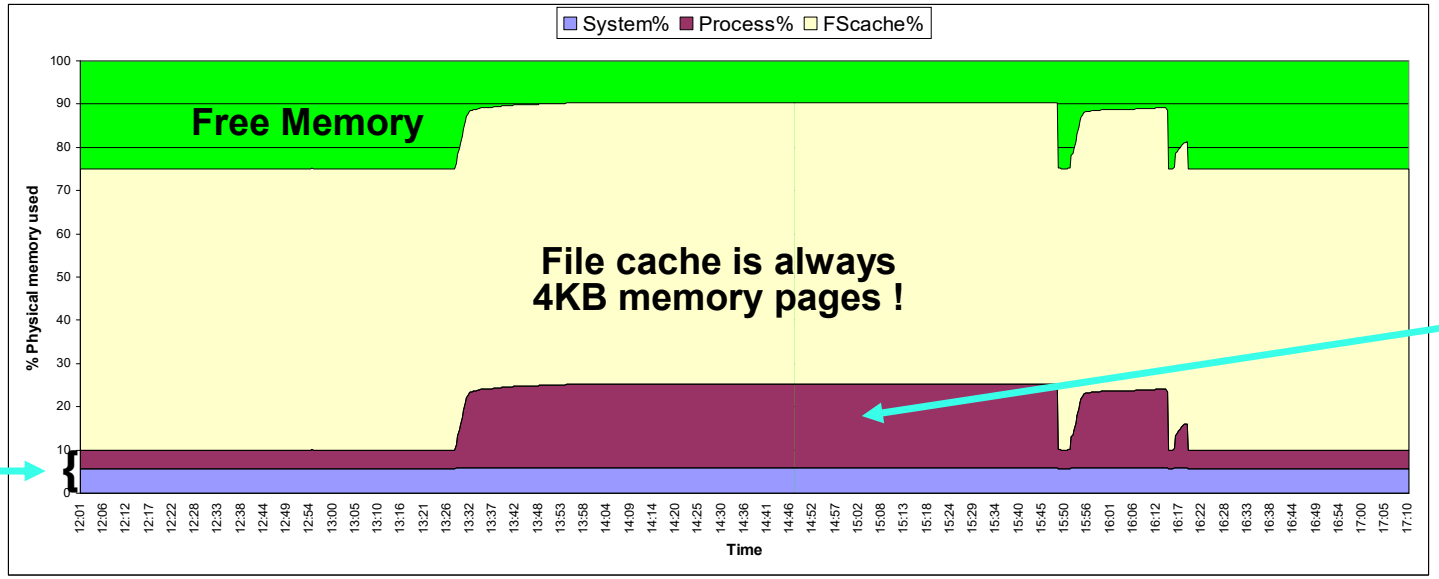
(**Huge**)

- Pinned, not pageable, never converted to other page size
- Manual allocation and assignment to specific LPAR(s) via HMC with server powered off
- 3 additional 16GB pages required in addition to specified Oracle sga_max_size value

ORACLE_SGA_PGSZ=16G

AIX Memory Management Concepts

- **Two primary categories of memory pages: Computational and File System**
- **AIX tries to utilize all available physical memory**
  - What is not required to support computational page demand will tend to be used for file system cache
- **Requests for new memory pages are satisfied from the free page list(s)**
  - Small reserve of free pages maintained by “stealing” File (typical) or Computational (bad == paging)
  - AIX uses “demand paging” algorithm – generally not written to paging space until “stolen”



Note: 16MB and 16GB pages are not included in this graph generated with nmon_analyzer!

VMM Page Stealing Process (lrud)

Definitions:

- lrud = VMM page stealing process = LRU Daemon (1 per memory pool)
- numperm, **numclient** = # pages currently used for filesystem buffer cache
- free pages = # pages immediately available to satisfy new memory requests

vmo Parameters:

- minfree = target minimum number of free memory pages (per pool and page size)
- maxfree = number of free memory pages at which lrud stops stealing pages (per pool and page size)
- minperm% = target min % real memory for any filesystem buffer cache
- maxperm%, **maxclient%** = target max % real memory for filesystem buffer cache

When does lrud (for a given memory pool and page size) start?

- When free pages < minfree (4K and 64K pages)
- When (maxclient - numclient) < minfree (4K pages only)

When does lrud (for a given memory pool and page size) stop?

- When free pages > maxfree (4K and 64K pages)
- When (maxclient – numclient) > maxfree (4K pages only)

VMM Page Stealing Thresholds (AIX 7.x, 6.1)

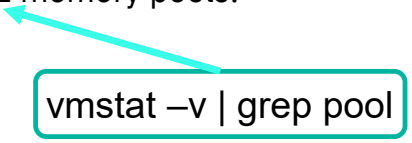
- **minfree / maxfree values are per memory pool**
  - Total system minfree = minfree * # of memory pools
  - Total system maxfree = maxfree * # of memory pools
- **AIX 7.x and 6.1 defaults are acceptable for most workloads**
  - **Consider increasing if** vmstat 'fre' column frequently approaches zero, or if "vmstat –s" shows significantly **increasing "free frame waits"** over time
- **Suggested starting points if tuning is required:**
  - minfree >= max(960,(120 x # logical CPUs)) / #mem pools
  - maxfree = minfree + ((MAX(maxpgahead, j2_maxPageReadAhead) * # logical CPUs) / # mem pools)

Example:

- LPAR with 10 VP and with SMT-2 enabled, with maxpgahead=8 and j2_maxPageReadAhead=128 and 2 memory pools:

  minfree = 1200 = max(960,(120 x 10 x 2) / 2

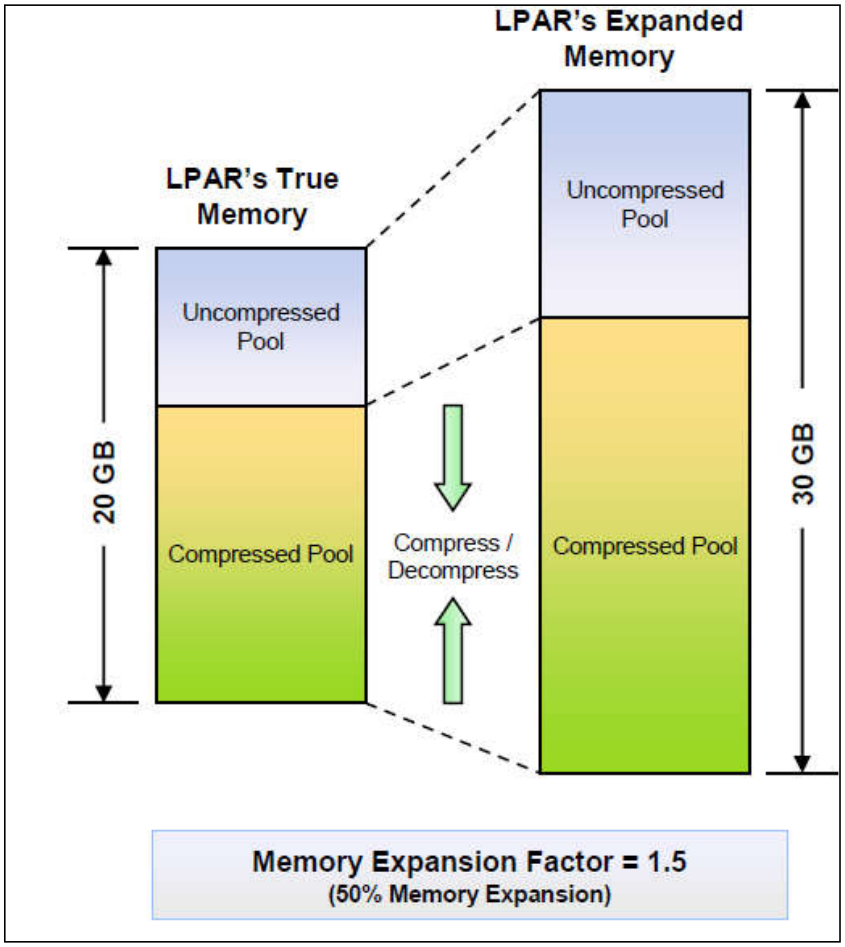
  maxfree = 2480 = 1200 + ((max(128,8) x 10 x 2) / 2)



➢ vmo –p –o minfree=1200 –o maxfree=2480

Active Memory Expansion (AME)

- Licensed feature of the physical server
- May potentially be used to increase the effective memory capacity of an LPAR without increasing the physical memory; **compresses less frequently used pages in memory**
- The AME planning tool (**amepat**) can be used to predict the amount of CPU overhead required to support varying levels of memory expansion
- AME, by default, uses only 4K memory pages.
- **64KB pages** supported on POWER8 for LPARs running in POWER8 mode, or for LPARs running on POWER9 or Power10
  - **Only supported / tested option with Oracle 12.2 or later**
  - Supported for single instance and RAC
  - Requires AIX 7.2 TL1, Firmware 860+ and AME acceleration enabled on the LPAR (ame_hw_accel=1)
  - 64K page usage in AME must be enabled by setting ame_mpsize_support=1
  - When using AME, Oracle can not be configured to use large (16MB) or huge (16GB) pages
  - Use of AME for production databases is typically not recommended; carefully test performance impact!
- Memory Expansion Factor is a DLPAR memory option



- POWER8, POWER9, Power10 have built-in hardware accelerator for AME compress/decompress

Displaying the LPAR CPU & Memory Configuration

The '**lssrad -va**' command displays a summary of the way physical processors and memory is allocated for a given LPAR:

```
# lssrad -va
    REF1   SRAD       MEM         CPU
    0
              0       110785.25   0-31
              1       125665.00   32-63
    1
              2       17430.00    64-95
              3       0.00        96-127
    2
              4       0.00        128-159
              5       0.00        160-191
```

Note the extremely poor distribution of memory in this example:

- **3 of 6 SRADs have no local memory at all.**

Every process running on a logical CPU on these SRAD will encounter memory latency induced slower performance.

- REF1: Hardware provided reference point identifying sets of resources that are near each other. e.g. socket in scale-out servers or node (CEC drawer) in scale-up servers. (In this example resources from 3 CEC in an E980 were assigned)
- SRAD: A **Scheduler Resource Affinity Domain**, i.e. an individual group of processors that all reside on the same chip
- MEM: The amount of local memory (in Megabytes) allocated to the SRAD
- CPU: The logical CPUs within the SRAD, e.g. with SMT4 enabled, 0-3 would be for the first CPU, 4-7 would be for the second CPU, etc...

Tip: Dynamic Platform Optimizer (via HMC) can help address this imbalance dynamically

Displaying the Local, Near and Far Memory Access Profile

The '**mpstat –d**' command displays statistics on local, near and far thread dispatches for every logical CPU in an LPAR:

```
# mpstat -d 1 60
cpu . . . S3hrd    S4hrd    S5hrd . . .
 0 . . .  99.5     0.1      0.4 . . .
 1 . . .  99.5     0.1      0.4 . . .
 2 . . .  98.1     0.5      1.4 . . .
 3 . . .  98.0     0.6      1.5 . . .
 4 . . .  99.5     0.1      0.4 . . .
 5 . . .  99.5     0.1      0.4 . . .
 6 . . .  98.1     0.5      1.4 . . .
 7 . . .  98.0     0.6      1.4 . . .
...
184 . . . 99.2     0.2      0.6 . . .
185 . . . 98.7     0.1      1.3 . . .
186 . . . 48.3     0.0      51.7 . . .
187 . . . 45.5     0.0      54.5 . . .
188 . . . 99.2     0.2      0.6 . . .
189 . . . 98.7     0.1      1.2 . . .
190 . . . 50.9     0.0      49.1 . . .
191 . . . 47.4     0.0      52.6 . . .
```

Excellent affinity (high percentage of local memory accesses) for the logical CPUs in the first Node/SRAD

Poor affinity (potentially high percentage of remote memory accesses) for the 3rd and 4th SMT threads for the physical CPUs in the last Node/SRAD

Note: this example is for a system running in SMT4 mode

- S3hrd: The percentage of local thread dispatches on this logical CPU
- S4hrd: The percentage of near thread dispatches on this logical CPU
- S5hrd: The percentage of far thread dispatches on this logical CPU
- '-': Indicates an SMT thread that is not currently active

Help the Hypervisor do its job

Don’t over-allocate CPUs

– If a given workload (LPAR) requires less processors than a single CEC, don’t allocate more (virtual) processors than are on a single CEC.

– If all the LPARs in each shared pool require (in aggregate) less processors than 2 CECs, don’t allocate more (virtual) processors than available in 2 CECs to the shared pool

– For Shared Processor LPARs, don’t over allocate vCPUs relative to Entitled Capacity (Rule of thumb - no more than 2 – 3 times entitled capacity; best practice is < 1.5x (less time slicing from hypervisor because of less processors to cycle through))

Don’t over-allocate memory

– May cause processors/memory to be allocated on additional CECs, not local to the processors assigned to the LPAR, because there wasn’t sufficient free memory available on the optimal CEC

Help the Hypervisor do its job

– Stay current on Firmware to avoid any known CPU/memory allocation or virtual processor dispatching issues

– Where appropriate, consider LPAR boot order to ensure high priority LPARs get optimal CPU to memory allocation with improved affinity (validate via *lssrad –av*)

Parameter Tuning (AIX 7.2, AIX 7.1)

Most AIX 7.x parameters configured by default to be ‘correct’ for most workloads

As of AIX 6.1, many tunables are now classified as ‘Restricted’

- Only change if AIX Support requests it
- Restricted parameters will not be displayed unless the ‘-F’ option is used for “vmo” or other commands

When migrating from AIX 5.3 to AIX 6.1 or AIX 7.x, existing parameter override settings in AIX 5.3 will be transferred to AIX 6.1 or later environment

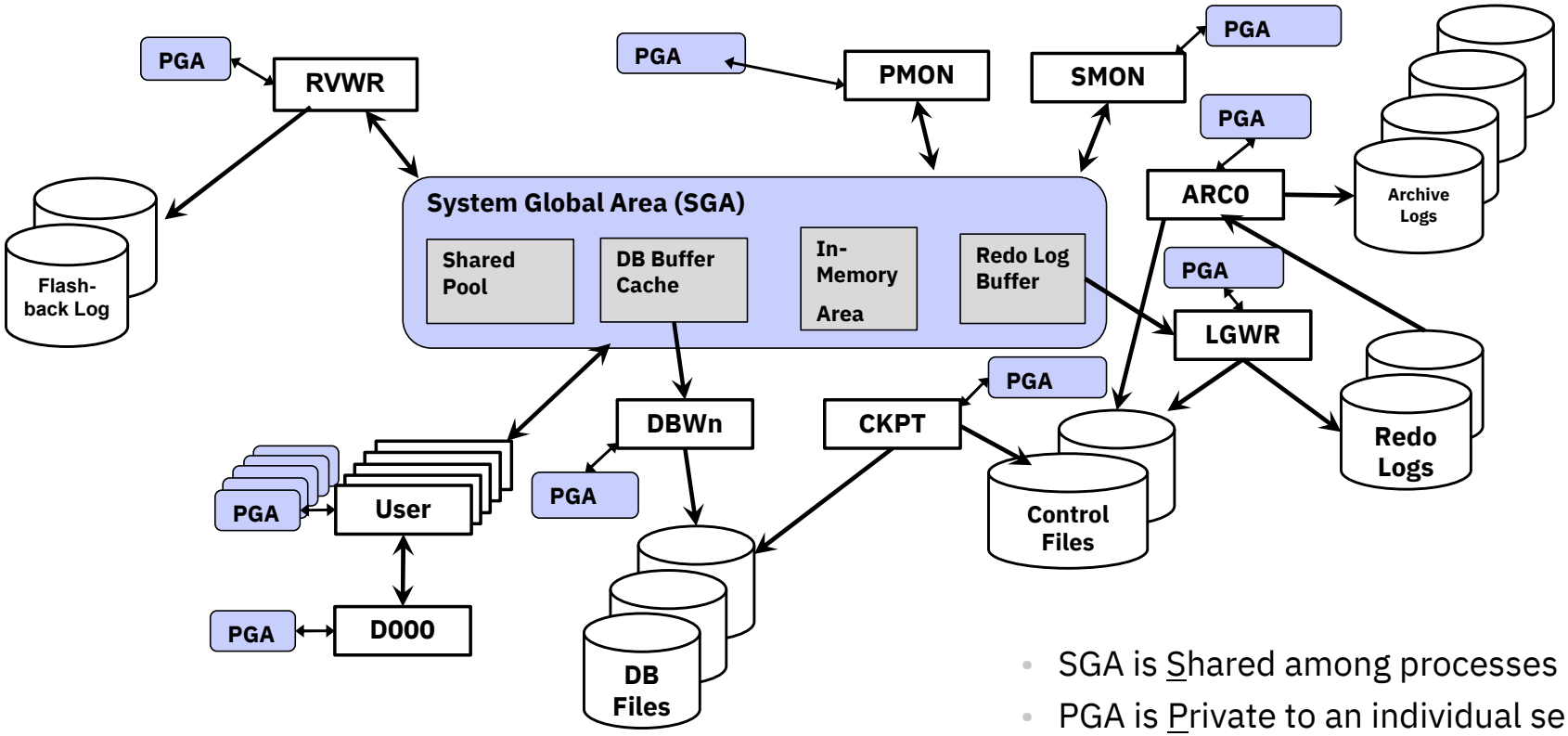
- After migration, review/verify parameter values are properly set
- … and don’t forget to look at LPARs which may have had a previous life with AIX 5.3

Recommended vmo “Starting Points” - Review

Parameter	Recommend Value	AIX 7.x Default	AIX 7.x Restricted	AIX 6.1 Default	AIX 6.1 Restricted	AIX 5.3 Default
esid_allocator	**1**	1	No (TL0:Yes)	**0**	Yes	N/A
minperm%	**3**	3	No	3	No	**20**
maxperm%	**90**	90	Yes	90	Yes	**80**
maxclient%	**90**	90	Yes	90	Yes	**80**
strict_maxclient	**1**	1	Yes	1	Yes	1
strict_maxperm	**0**	0	Yes	0	Yes	0
lru_file_repage	**0**	N/A	N/A	0	Yes	**1 or 0(*1)**
lru_poll_interval	**10**	10	Yes	10	Yes	10
minfree	**960**	960	No	960	No	960
maxfree	**1088(*2)**	1088	No	1088	No	1088
page_steal_method	**1**	1	Yes	1	Yes	**0**
memory_affinity	**1**	1	Yes	1	Yes	1
v_pinshm	**0**	0	No	0	No	0
lgpg_regions	**0**	0	No	0	No	0
lgpg_size	**0**	0	No	0	No	0
maxpin%	**Leave at Default**	90	No	80	No	80
vmm_klock_mode	**2 (see notes)**	2	No	1	Yes	N/A

*1 Depending on AIX 5.3 TL level *2 Do not reduce below default

Oracle Server Architecture – Memory Structures



- SGA is Shared among processes
- PGA is Private to an individual server or background process

Only a subset of Oracle process types is shown.

Oracle and “PRE_PAGE_SGA”

- Starting with Oracle 12.1 this settings defaults to “TRUE”. Earlier releases have set it to “FALSE”.
- If set to “TRUE”, **every** Oracle process will “touch” all SGA memory pages at startup to map them into its address space.
  - **Pros**
    - No additional AIX physical memory mapping for SGA memory pages after startup
  - **Cons**
    - Startup of any Oracle process takes longer; that includes Oracle connection shadow processes!
    - The higher new connection rates are the more contention the startup will create in AIX virtual memory management
    - Time required is directly proportional to the number of memory pages allocated for the SGA

    → memory page size and specified SGA size are directly correlated to the time required.

SGA size	4KB pages	64KB pages	16MB pages
10GB	2,621,440	163,840	640
100GB	26,214,400	1,638,400	6,400
250GB	65,536,000	4,096,000	16,000
1TB	268,435,456	16,777,216	65,536

Note: If you are planning to use the “In-Memory” feature of Oracle database 12c or later it is recommended to set pre_page_sga = TRUE (Default)

Oracle Memory Structures Allocation – 10g / 11g

Available since 10g : Automatic Shared Memory Management (ASMM)

- sga_target (dynamic) – if set, the db_cache_size, shared_pool_size, large_pool_size and streams_pool_size are dynamically sized; can dynamically grow to sga_max_size.
  - Minimum values for these pools can optionally be specified and it is recommended to do so
- *If LOCK_SGA=true then physical memory according to sga_max_size is allocated at DB startup! If sga_max_size is not set sga_target is used instead.*
- *To use ASMM, sga_target must be >0*

Available since 11g : Automatic Memory Management (AMM)

- memory_target (dynamic) – specifies the total memory size to be used by the instance **SGA and PGA**. Exchanges between SGA and PGA are done automatically according to workload requirements
- If sga_target and pga_aggregate_target are not set, the policy is to give 60% of memory_target to the SGA and 40% to the PGA
- memory_max_target (static parameter) – specifies the maximum memory size for the instance (not a hard limit!)
- *To activate Automatic Memory Management, memory_target must be >0*
- *See Metalink notes 443746.1 and 452512.1 explaining AMM and these parameters*

AMM dynamic resizing of the shared pool can cause a fair amount of “cursor: pin s” wait time. One strategy to minimize this wait time is to set minimum sizes for memory areas you particularly care about.

In addition, you can change the frequency how often AMM analyzes and adjusts the memory distribution. See: Metalink note: 742599.1 (_memory_broker_stat_interval)

Oracle Memory Structures Allocation – 12c or later

12c+: pga_aggregate_limit

- Limit of how large the aggregate PGA of all oracle processes can grow to
- Sessions with the highest amount of allocated PGA will be terminated until compliance is reached

12c+: inmemory_size

- When the Oracle “In Memory” Option is used, specifies the size within the SGA to reserve for “In-Memory” objects pinned in memory
- This parameter is not dynamic in 12.1, but can be dynamically increased (not decreased) in 12.2 or later

Note:

MEMORY_TARGET/MEMORY_MAX_TARGET are not hard limits and Oracle can utilize significantly more memory for PGA if needed. With Oracle 12c we observed some issues when calculating the active limits.

Recommended starting point for Oracle memory management setting:

1. Use SGA_TARGET and SGA_MAX SIZE rather than MEMORY_TARGET and MEMORY_MAX_TARGET
2. Specify values for critical SGA components like buffer cache in *init.ora / spfile* to ensure minimum is reserved
3. Most environments should use 64K pages rather than pinned 16MB pages if SGA < 100GB
4. If you do pin the SGA, make sure you also pin the kernel with vmm_klock_mode=2

SGA_MAX_SIZE and LOCK_SGA implications (19c, 18c,12c, 11g)

LOCK_SGA=false **Preferred** (at least for SGA < 100GB)

- Oracle dynamically allocates memory for the SGA only as needed up to the size specified by SGA_TARGET
- SGA_TARGET can be dynamically increased, up to SGA_MAX_SIZE
- 64K pages automatically used for SGA if supported in the environment.
  - If needed, 4K (or 16M) pages are converted to 64K pages.
  - Down-conversion of 16M pages to 64K pages is only triggered at DB startup if required.
  - After startup, additional unused 16M pages are not converted, even if not enough 4K or 64K pages are available → potential for paging to paging space.

Notes:

- If you utilize environment variable ORACLE_SGA_PGSZ to set SGA memory page size manually, then Oracle will allocate all memory specified via sga_max_size at startup! Memory is not pinned.
- If 16M or 16G is specified as page size, those pages will still be pinned.

LOCK_SGA=TRUE implications (19c, 18c, 12c, 11g)

LOCK_SGA=true Discouraged (at minimum requires strict change controls)

- Oracle pre-allocates all memory as specified by SGA_MAX_SIZE and pins it in memory, even if it's not all usable (i.e. SGA_TARGET < SGA_MAX_SIZE)
- If Oracle Automatic Memory Management (AMM) is used (MEMORY_MAX_TARGET), there must be enough 16M pages pre-allocated to support **both SGA and PGA**
- If sufficient 16M pages are available, those will be used. Otherwise, all the SGA memory will be allocated from 64K (if supported) or 4K pages (if 64K pages are not supported). If needed, 4K or 16M pages will be converted to 64K pages, but 16M pages are never automatically created.
- If a value for SGA_MAX_SIZE is specified larger than the amount of available memory for computational pages, the system can become unresponsive due to system paging.
- If the specified SGA_MAX_SIZE is much larger than the currently available pages on the combined 64K and 16M page free lists, the database startup can fail with error: *"IBM AIX RISC System/6000 Error: 12: Not enough space".* In this case re-try to start the database.

If SGA is > 100GB, then use of 16MB pages is recommended, **after implementing strong memory monitoring controls**, as otherwise database startup can take a very long time!

Enabling Large Pages for Oracle SGA

If you <u>MUST</u> do it, the following is required to implement use of Large Pages:

Oracle

- LOCK_SGA = TRUE

AIX

- *Calculate required # of large pages* = INT [(SGA size – 1) / 16 MB)] + 3
  - ➢ If Oracle Automatic Memory Management (AMM) is used, large pages need to be allocated based on max_memory_target, rather than SGA size
- # vmo **–r** –o lgpg_regions = <*no_of_large_pages*> -o lgpg_size=16777216
- # vmo –p –o v_pinshm = 1
- # chuser capabilities = CAP_NUMA_ATTACH,CAP_BYPASS_RAC_VMM,CAP_PROPAGATE oracle
- # chuser capabilities = CAP_NUMA_ATTACH,CAP_BYPASS_RAC_VMM,CAP_PROPAGATE grid **If RAC**
  Do not modify maxpin% default (80 for AIX 6.1, 90 for AIX 7.1, 7.2)
- # bosboot –a **This optimizes memory allocation at boot to match configuration**

To verify that Oracle is using large pages:

– svmon -P $(ps -elf | egrep " ora_smon_${ORACLE_SID} " | grep -v egrep | awk '{print $4}') | grep shmat
– Part of output will look similar to the output below (Note the L) which indicates you are using large pages:

```
5390db9 70000019 work default shmat/mmap L 16 16 0 16
4670ce7 7000000c work default shmat/mmap L 16 16 0 16
 821002 7000001e work default shmat/mmap L 16 16 0 16
```

Note: Environment variable ORACLE_SGA_PGSZ=16[M|G] also allows to set large/huge page size for SGA.

AIX nmon – “Easy” way to evaluate usage of 16MB pages

nmon -> **M**

lgpg_regions

```
-topas_nmon--j=JFS----------Host=p21a01---------Refresh=2 secs---13:35.00-
 Multiple-Page-Size (in Pages)
Page Sizes= 4   Page Size ->          4KB       64KB       16MB       16GB
  numframes memory frames       149743568    7394691     196700          0
     numfrb free list pages     109394165    6765565         91          0
  numclient client frames         1508941          0          0          0
numcompress compressed segments         0          0          0          0
    numperm non-working segments    1508941          0          0          0
  numvpages accessed virtual pages 38730579     629126     196609          0
    minfree min free list             960         60          0          0
    maxfree max free list            1088         68          0          0
    numpout page-outs                   0          0          0          0
  numremote remote page-outs            0          0          0          0
 numwseguse in use working seg    38730579     629126     196609          0
 numpseguse in use persistent            0          0          0          0
numclseguse in use client          1618824          0          0          0
 numwsegpin pinned working        38348930     263720     196609          0
 numpsegpin pinned persistent            0          0          0          0
numclsegpin pinned client seg        4183          0          0          0
numpgsp_pgs allocated PageSpace    107364          0          0          0
  numralloc remote allocations          0          0          0          0
 pfrsvdblks system reserv.blocks   11825716     739469          0          0
    pfavail pin available         111390455    7130971          0          0
 pfpinavail pinnable@apps level   116910870    7279238          0          0
 system_pgs SCBs marked V_SYS       6856739     234887          0          0
 nonsys_pgs SCBs not V_SYS          2001656     394225     196609          0
--- Below are Rates per Second ---       4KB       64KB       16MB       16GB
  numpermio non-w.s. pageouts           0          0          0          0
     pgexct Page Faults                 4         26          0          0
     pgrclm Page Reclaims               0          0          0          0
    pageins Paged in   -All             0          0          0          0
   pageouts Paged out -All              0          0          0          0
   pgspgins Paged in  -PageSpace        0          0          0          0
  pgspgouts Paged out-PageSpace         0          0          0          0
    numsios I/O Started                 0          0          0          0
  numiodone I/O Completed               0          0          0          0
  zerofills Zero filled                 0          0          0          0
    exfills Exec() filled               0         17          0          0
      scans Scans by clock              0          0          0          0
     cycles Clock hand cycles           0          0          0          0
   pgsteals Page Steals                 0          0          0          0
```

nmon -> **M M**

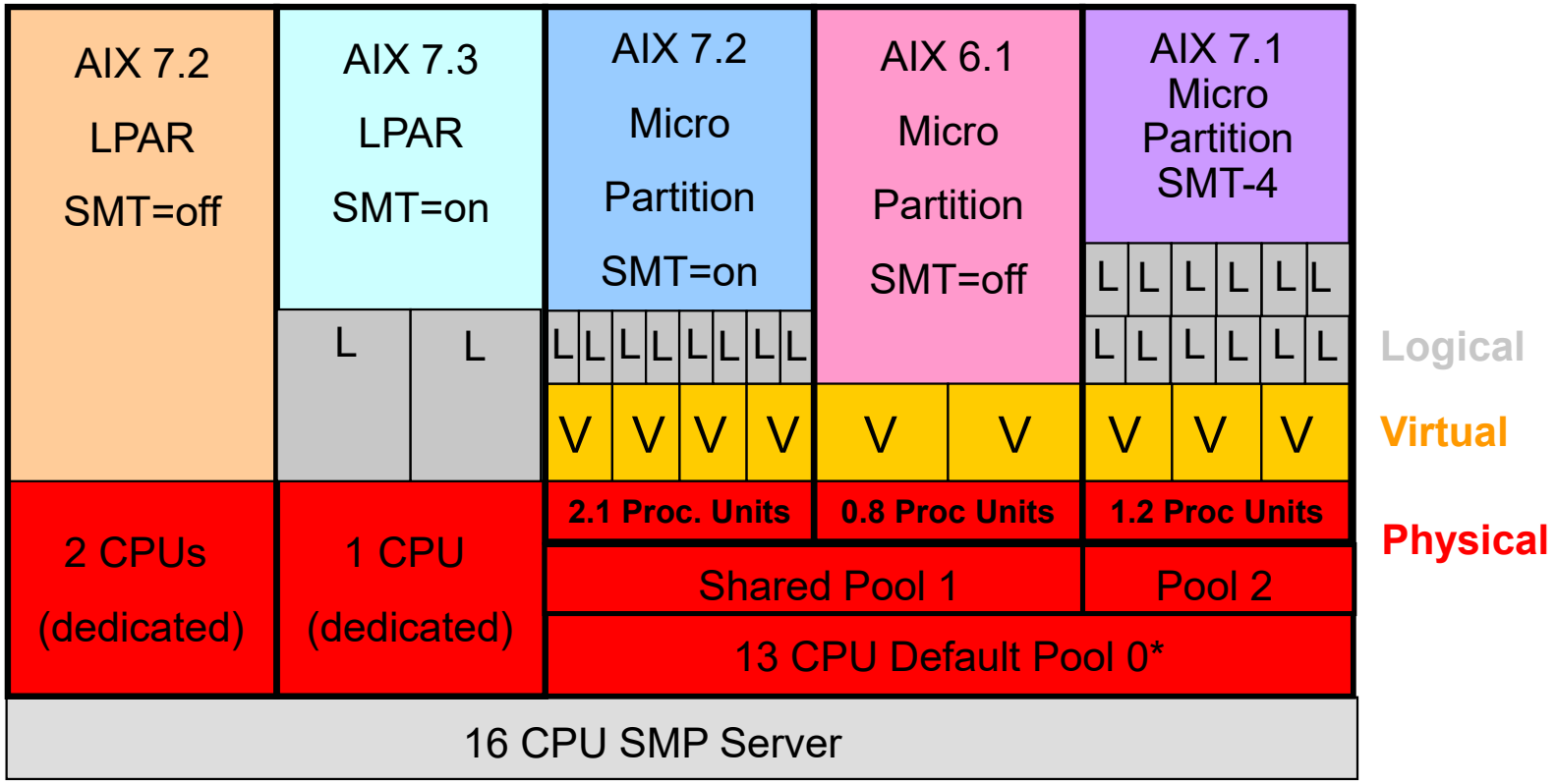
sga_max_size + 3x16MB pages to be lower than

```
-topas_nmon--D=Disk-Stats-------Host=p21a01---------Refresh=2 secs---13:35.20-
 Multiple-Page-Size (in Megabytes)
Page Sizes= 4   Page Size ->          4KB       64KB       16MB       16GB
  numframes memory frames          584936     462168    3147200          0
     numfrb free list pages        427320     422823       1456          0
  numclient client frames            5894          0          0          0
numcompress compressed segments         0          0          0          0
    numperm non-working segments       5894          0          0          0
  numvpages accessed virtual pages   151292      39345    3145744          0
    minfree min free list               4          4          0          0
    maxfree max free list               4          4          0          0
    numpout page-outs                   0          0          0          0
  numremote remote page-outs            0          0          0          0
 numwseguse in use working seg      151292      39345    3145744          0
 numpseguse in use persistent            0          0          0          0
numclseguse in use client             6324          0          0          0
 numwsegpin pinned working          149801      16483    3145744          0
 numpsegpin pinned persistent            0          0          0          0
numclsegpin pinned client seg           16          0          0          0
numpgsp_pgs allocated PageSpace        419          0          0          0
  numralloc remote allocations          0          0          0          0
 pfrsvdblks system reserv.blocks    46194      46217          0          0
    pfavail pin available          435119     445685          0          0
 pfpinavail pinnable@apps level    456683     454952          0          0
 system_pgs SCBs marked V_SYS       26784      14680          0          0
 nonsys_pgs SCBs not V_SYS           7820      24664    3145744          0
--- Below are Rates per Second ---       4KB       64KB       16MB       16GB
  numpermio non-w.s. pageouts           0          0          0          0
     pgexct Page Faults                 4          0          0          0
     pgrclm Page Reclaims               0          0          0          0
    pageins Paged in   -All             0          0          0          0
   pageouts Paged out -All              0          0          0          0
   pgspgins Paged in  -PageSpace        0          0          0          0
  pgspgouts Paged out-PageSpace         0          0          0          0
    numsios I/O Started                 0          0          0          0
  numiodone I/O Completed               0          0          0          0
  zerofills Zero filled                 0          0          0          0
    exfills Exec() filled               0          0          0          0
      scans Scans by clock              0          0          0          0
     cycles Clock hand cycles           0          0          0          0
   pgsteals Page Steals                 0          0          0          0
```

Agenda

- Memory
- CPU
- I/O
- Network
- Miscellaneous

Physical, Virtual, Logical Layers



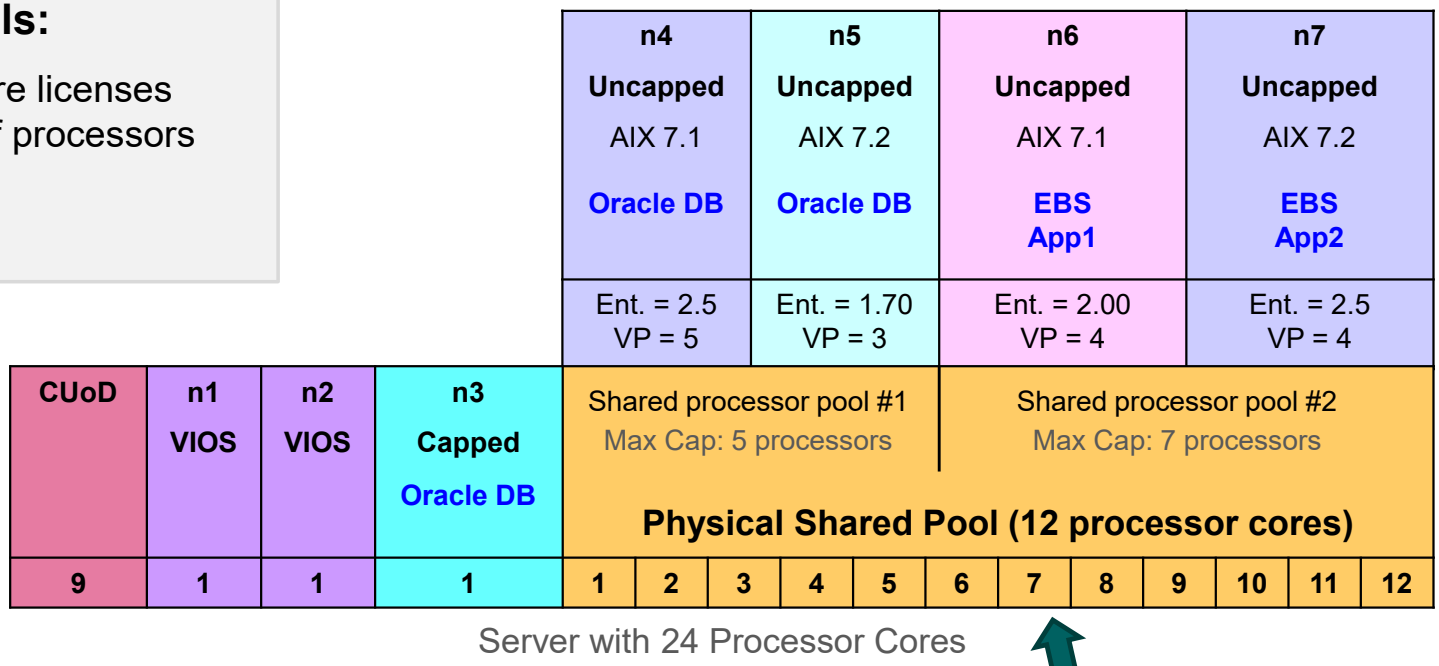
Think "PVL " **P=Physical** **V=Virtual** **L=Logical (SMT)**

* All activated, non-dedicated CPUs are automatically placed into the shared processor pool 0.
Only 2.1+0.8+1.2 = 4.1 processor units of "desired capacity" has been allocated from the pool of 13 CPUs

PowerVM Shared Processor Pools – Licensing Benefits

Multiple shared processor pools:

- Can reduce the number of software licenses by putting a limit on the amount of processors an uncapped partition can use
- Create up to 64 shared pools



Server with 24 Processor Cores

Incrementally activate and license only 1 core at a time!

Oracle DB core – license factors:

POWER7/7+:	1.0
POWER8:	1.0
POWER9:	1.0
POWER10:	1.0

Cores to License in this Example:

Oracle Database: 6
- 5 from shared processor pool #1
- 1 from LPAR n3

Oracle EBS: 7
- 7 from shared processor pool #2

Note: IBM Power Systems Private Cloud Solution with Dynamic Capacity (*Enterprise Pools 2.0*) activates all resources on all servers in the server pool but does not affect capacity limits of shared processor pools > #0 or capped CPU LPARs.

CPU Related Oracle Parameters

Oracle Parameters based on the # of **logical** CPUs (# of VP times SMT mode [1,2,4,8])

- **Parameters**
  - CPU_COUNT = # of **logical** CPUs
  - DB_WRITER_PROCESSES = 1 per every 8 logical CPUs
  - GCS_SERVER_PROCESSES
- **Degree of Parallelism**
  - Can be set at the user level, table level, or query level
  - Restricted by PARALLEL_MAX_SERVERS
  - Default setting = 1 (NOPARALLEL)
  - Default degree = (CPU_COUNT * PARALLEL_THREADS_PER_CPU)
- **Cost Based Optimizer (CBO)**
  - execution plan may be affected by changes in # of logical CPU; check explain plan

Notes:

- Available Oracle releases (at least till 19.11) do not handle more than ~ 1300 CPU in AWR performance statistics and related output in AWR reports is blank or incorrect. A one-off patch is available for 19.7 and 19.11 with BUG # 31844249
- "GCS_SERVER_PROCESSES" needs to be the same in all instances of a RAC cluster. If a RAC node has a different number of processors configured, this parameter needs to be set manually to match in all RAC nodes.

Oracle - CPU related options (not typically modified)

Instance Caging

- Available with Oracle 11gR2 or 12c and later
- Requires Oracle Resource Manager with CPU directives
- CPU_COUNT set manually
- Sometimes used for prioritizing CPU allocation where multiple databases share one OS image

11g/12c Non-Uniform Memory Access (NUMA) Optimization

- Disabled by default (on AIX)
- Not generally recommended for use on POWER
- In some cases, with very large P8/P9 implementations, it may provide performance benefits

12c+ – threaded_execution parameter

Beware of APAR: IJ21353: PTHREAD_MUTEX_DESTROY() CAN COREDUMP

- Feature directing Oracle to utilize a threaded execution model (like on Windows) instead of process-based execution model. Feature is disabled by default (set to False)
- Typically not set in AIX environments and not heavily tested – suggest heavy performance testing before deploy in production environment!

12c+ - processor_group_name parameter *(expert tuning, only in very specific cases)*

- Tells the database instance to run within the specified OS processor group
- An AIX processor group is called an ‘rset’ and is configured using the ‘mkrset’ command
- Allows to restrict the number of processors Oracle processes can be deployed to (not a recognized license restriction option on AIX!)

CPU Recommendations

Micropartitioning Guidelines

– Virtual CPUs (vCPUs) should always be <= physical processors in shared CPU pool

– Use default processor folding behavior unless IBM AIX support recommends otherwise; but suggest to set vpm_xvcpus=2 For RAC must be set at minimum to 2

CAPPED LPAR

– vCPUs should be the nearest integer >= capping limit / entitlement

UNCAPPED LPAR

– vCPUs should be set to the max peak demand requirement

– Best practice, number of vCPUs should not be more than 1.5x to 2x specified entitlement

DLPAR considerations

– Oracle CPU_COUNT dynamically recognizes change in # cpus (physical and logical)

- Max CPU_COUNT limited to 3x CPU_COUNT at instance startup for Oracle pre-12c

  This restriction does not exist in 12c or later

Note bug in Oracle 12.1 related to configured number of logical processors –
Patch 18775971: ORA-04031: UNABLE TO ALLOCATE ("SHARED POOL","UNKNOWN OBJECT","PDB DYN ...), see note: 2225248.1

CPU Recommendations

Test and evaluate SMT4 or SMT8 respectively prior to selecting SMT mode

- For SMT8, consider reducing DB_WRITER_PROCESSES
- When using parallel query, consider reducing or restricting default parallel degree
- RAC environments may also benefit from tuning GCS_SERVER_PROCESSES
- SMT-8 mode should be the starting point for Power9 and Power10 processor-based systems

PARALLEL_THREADS_PER_CPU=1 (default for Oracle on AIX, should be set at least with SMT 8, maybe also SMT4)

Dedicated vs. Shared CPU LPAR

- Dedicated CPUs may provide better performance, lower latency, for single or lightly threaded workloads
- Shared CPUs may provide better price/performance and/or greater ability to support peak workload demand

Note:

Even with dedicated CPU LPARs you can donate unused CPU resources to other LPARs, but it is not as fine grained as with Shared CPU LPARs. (***Dedicated, donating*** LPAR). Donating resources may reduce the benefits of the dedicated mode.

Virtual Processors - Folding

Dynamically adjusts actively used Virtual Processors

- System consolidates loads onto a minimal number of VPs
  - Scheduler computes utilization of VPs every second and calculates needed VP as: ceiling(physCPU_util + vpm_xvcpus)
    - If active VPs is greater than calculated needed VP, a VP is put to sleep
    - If calculated VPs needed are greater than the current active VPs, more are enabled (can be more than 1 at a time!)

- Folding active by default and controlled by schedo tunables:
  - **`vpm_xvcpus`**, **`vpm_fold_policy`**

Goal is to increase processor utilization, affinity and reduce overhead in Hypervisor

- Inactive VPs don't get dispatched and waste physical CPU cycles
- Fewer VPs can be more accurately dispatched to physical resources by the Hypervisor

Monitoring

- mpstat -s, **mpstat -v**

Oracle database LPARs should typically run with VP folding active

- **Setting vpm_xvcpus=2 typically addressed any unexpected side effects and is suggested as default for Oracle DB LPARs**
- **vpm_fold_policy should be left on default (1)**

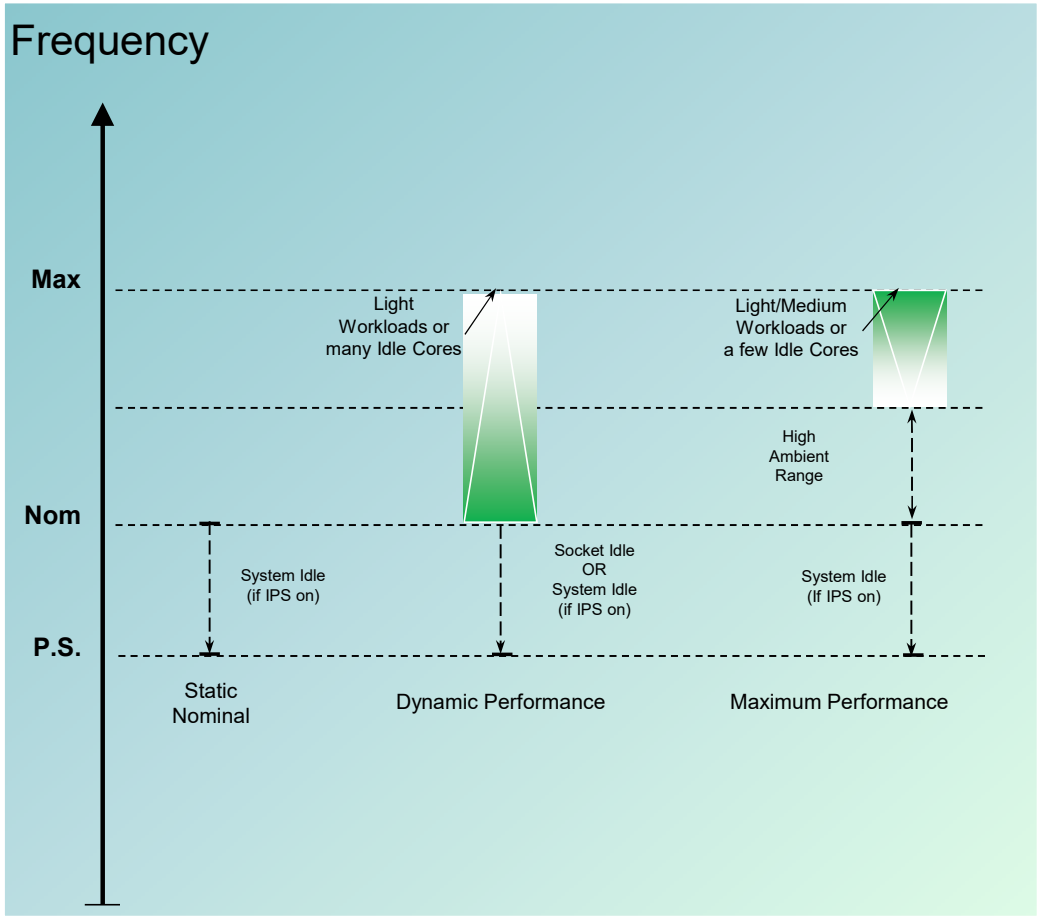
When to adjust folding behavior

- For Oracle RAC environments vpm_xvcpus=2 is minimum required value
- Burst/Batch workloads with short response time requirements may need sub-second dispatch latency
  - Option to disable processor folding or, preferred, manually tune the # of VPs
  - # schedo –o vpm_xvcpus=[-1 | N]
    - Where N= # of VPs to enable in addition to VPs for physical CPU utilization
    - -1 disables folding

EnergyScale Mode Operation – PowerVM – POWER9/Power10

- **Static Power Saver Mode**
- **Static Nominal Mode**
  - Idle Power Saver (IPS) can be on or off
- **Dynamic Performance Mode**
  - Workloads run at highest frequency possible
  - If all cores/socket idle for 100s of ms, drop to power save frequency
  - CPU managed to nominal power draw
  - Max Workload/Max Cores runs >= nominal in all environments
    - **Deterministic across full environment range**
    - The system performance for a given workload will be the same regardless of ambient temperature
  - Idle Power Saver (IPS) can be on or off*
- **Maximum Performance Mode** (POWER9/Power10 default)
  - Workloads run at highest frequency possible
  - No frequency drop due to cores/socket idleness unless IPS is enabled
  - CPU managed to higher power draw level – Higher acoustics
  - System performance is increased at lower ambient temperatures
  - Idle Power Saver (IPS) can be on or off*

* IPS mode lowers the frequency to minimum if the entire system (all CPUs in all sockets) is idle for minutes – even If MPM is enabled



Notices and disclaimers

U.S. Government Users Restricted Rights — use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM.

Information in these presentations (including information relating to products that have not yet been announced by IBM) has been reviewed for accuracy as of the date of initial publication and could include unintentional technical or typographical errors. IBM shall have no responsibility to update this information. **This document is distributed "as is" without any warranty, either express or implied. In no event, shall IBM be liable for any damage arising from the use of this information, including but not limited to, loss of data, business interruption, loss of profit or loss of opportunity.** IBM products and services are warranted per the terms and conditions of the agreements under which they are provided.

IBM products are manufactured from new parts or new and used parts. In some cases, a product may not be new and may have been previously installed. Regardless, our warranty terms apply."

Any statements regarding IBM's future direction, intent or product plans are subject to change or withdrawal without notice.

Performance data contained herein was generally obtained in a controlled, isolated environments. Customer examples are presented as illustrations of how those

customers have used IBM products and the results they may have achieved. Actual performance, cost, savings or other results in other operating environments may vary.

References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business.

Workshops, sessions and associated materials may have been prepared by independent session speakers, and do not necessarily reflect the views of IBM. All materials and discussions are provided for informational purposes only, and are neither intended to, nor shall constitute legal or other guidance or advice to any individual participant or their specific situation.

It is the customer's responsibility to insure its own compliance with legal requirements and to obtain advice of competent legal counsel as to the identification and interpretation of any relevant laws and regulatory requirements that may affect the customer's business and any actions the customer may need to take to comply with such laws. IBM does not provide legal advice or represent or warrant that its services or products will ensure that the customer follows any law.

Notices and disclaimers

- Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products about this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products. IBM does not warrant the quality of any third-party products, or the ability of any such third-party products to interoperate with IBM’s products. **IBM expressly disclaims all warranties, expressed or implied, including but not limited to, the implied warranties of merchantability and fitness for a purpose.**
- The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents, copyrights, trademarks or other intellectual property right.
- IBM, the IBM logo, ibm.com and [names of other referenced IBM products and services used in the presentation] are trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at: www.ibm.com/legal/copytrade.shtml