POWER Ask The Experts 2013

Performance Best Practices with POWER7

Nigel Griffiths
IBM Power Systems
Advanced Technology Support, Europe

Email 1 from Jim

My VM has
- Entitlement of 8 CPUs uncapped,
- Virtual Processors (VP) at 10 and
- SMT=4
- so why with only a 20 users online out of 40,
- is it using 8 physical CPU cores already?

Answer 1 → E=8 VP=10 uncapped 20 users

- You set VP=10 which states:
  "You are happy for the VM to use 10 whole CPUs"
- AIX default behaviour is to use all the VPs for maximum performance
- If VP is 10 then as workload grows
  it will use up all 10 CPU cores quickly
- AIX first uses SMT thread 1 on all 10 CPU cores
  before allocating work to the 2nd SMT thread,
  ditto 3rd & 4th SMT threads

Answer 1 → E=8 VP=10 uncapped 20 users

With just

- 10 busy processes or
- 20 processes using an average of half a CPU each
- 30 processes using an average of a third a CPU each
- …

Then all 10 virtual processors =10 CPU cores are used

The CPU cores are 100% allocated to this VM's use

That is what you asked for (VP=10) & what you got

Answer 1 → E=8 VP=10 uncapped 20 users

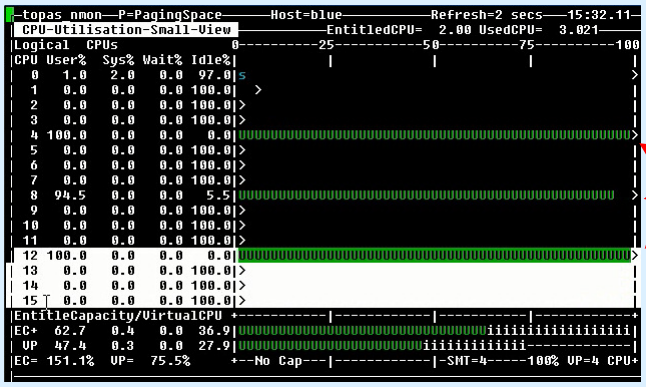
But you can still use the
2nd,
3rd & 4th
SMT threads to get more work done

Ball park guess 40% to 60% more,
depending on the application instructions

Can we see this on the machine?

- Quick reminder
  - SMT threads are reported as Logical CPUs
    - SMT = Simultaneous Multi-Threading threads
  - Virtual Processor map to physical CPU core (when running)
  - If SMT=4 then 1 VP shows up as 4 Logical CPUs
  - Intelligent SMT threads = dynamic switch SMT mode

Three spinning programs



1st thread of
3 different
Physical CPU cores

1 VP
running on
1 physical CPU core
with 4 SMT threads

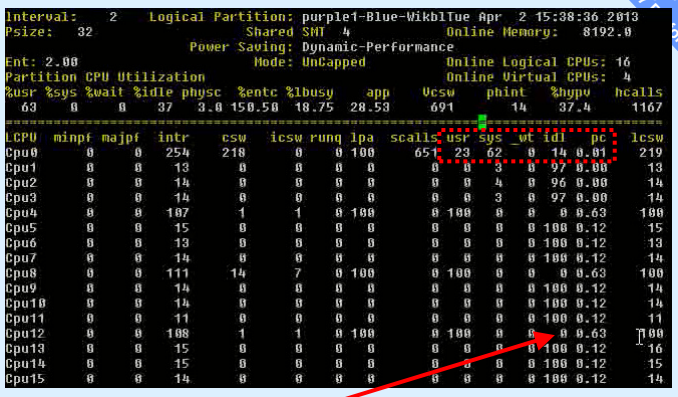
mpstat 1

cpu	min	maj	mpc	int	cs	ics	rq	mig	lpa	sysc	us	sy	wa	id	pc	%ec	lcs
0	91	0	0	319	194	7	0	1	100	611	25	58	0	17	0.01	0.2	260
1	0	0	0	11	0	0	0	0	-	0	0	2	0	98	0.00	0.1	11
2	0	0	0	14	0	0	0	0	-	0	0	3	0	97	0.00	0.1	14
3	0	0	0	10	0	0	0	0	-	0	0	1	0	99	0.00	0.1	10
4	0	0	0	105	1	1	0	0	100	0	100	0	0	0	0.63	21.0	101
5	0	0	0	12	0	0	0	0	-	0	0	0	0	100	0.12	4.1	12
6	0	0	0	12	0	0	0	0	-	0	0	0	0	100	0.12	4.1	12
7	0	0	0	12	0	0	0	0	-	0	0	0	0	100	0.12	4.1	12
8	0	0	0	104	16	8	0	0	100	0	100	0	0	0	0.63	20.9	101
9	0	0	0	11	0	0	0	0	-	0	0	0	0	100	0.12	4.1	11
10	0	0	0	11	0	0	0	0	-	0	0	0	0	100	0.12	4.1	11
11	0	0	0	11	0	0	0	0	-	0	0	0	0	100	0.12	4.1	11
12	0	0	0	107	1	1	0	0	100	0	100	0	0	0	0.63	20.9	100
13	0	0	0	11	0	0	0	0	-	0	0	0	0	100	0.12	4.1	11
14	0	0	0	11	0	0	0	0	-	0	0	0	0	100	0.12	4.1	11
15	0	0	0	11	0	0	0	0	-	0	0	0	0	100	0.12	4.1	11
ALL	91	0	0	772	212	17	0	1	100	611	63	0	0	37	3.01	150.5	699

0.63 of a CPU= 63%
Trying to show spare capacity
of other SMT threads

See later: LPAR Utilisation numbers have been Fiddled with

topas -L - - or - - topas then L



- 0.63 of a CPU= 63%
- Trying to show spare capacity of other SMT threads

Answer 1 for Jim

This is expected behaviour

Go check your spare SMT capacity + Run Queue size

I suspect you will have the resources needed for the other users

Email 2 from Jane

- Power 770 is 85+% busy
- Vital LPAR settings E=0.4 VP=4 uncapped
  - Compared to POWER6
    → E reduced & VP=Same plus consolidation
- Performance is slow, application seems to hang and the users are revolting!

Answer 2a → Pool 85+% busy E=0.4 VP=4

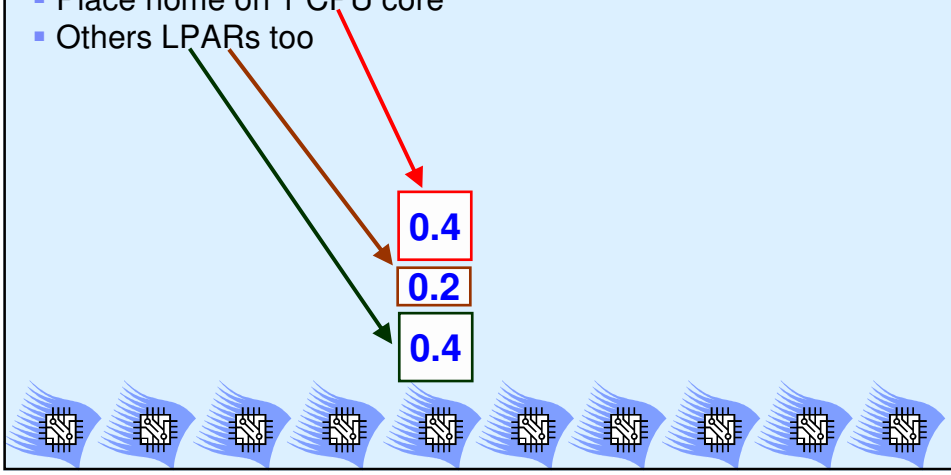
- I liken this set up to my son just passing his driving test & I would like him to stay below 40 MPH, so I set the governor on the accelerator to 400 MPH, so he can overtake safely!
- Doh!
- Obviously dumb
  - He does not need that size safety margin

Email 2 from Jane

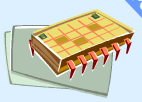
- Power 770 is 85+% busy
- Vital LPAR settings E=0.4 VP=4
  - Same as our POWER6 machine
- Only Entitlement = 0.4 → guaranteed
- Virtual Processor = 4 → LPAR can be spread out

VM placement at start up time

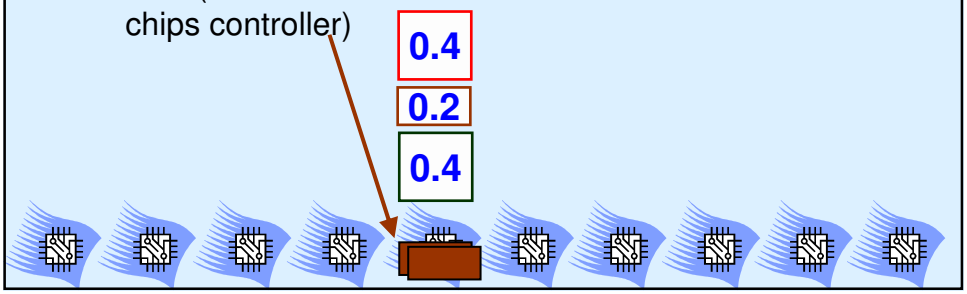
- E=0.4 and VP=4
- Hypervisor will do this …
- E = Normal usage so allocate that much now
- Place home on 1 CPU core
- Others LPARs too

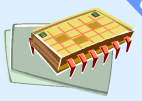


VM placement at start up time

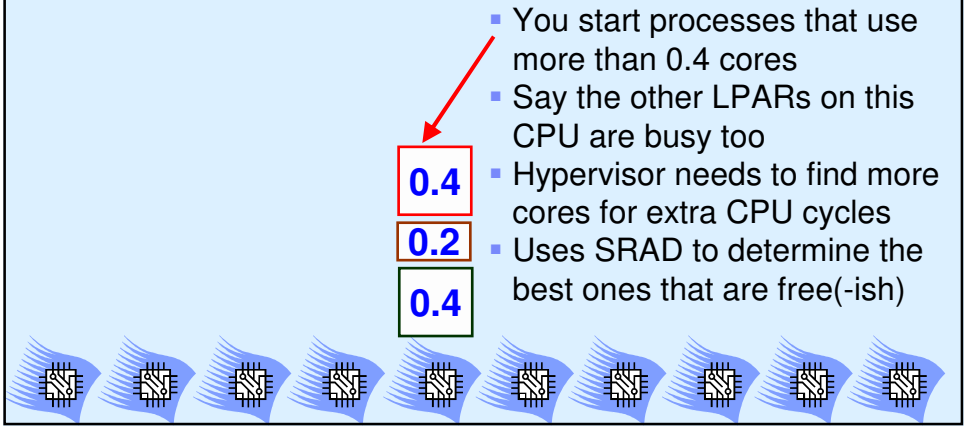


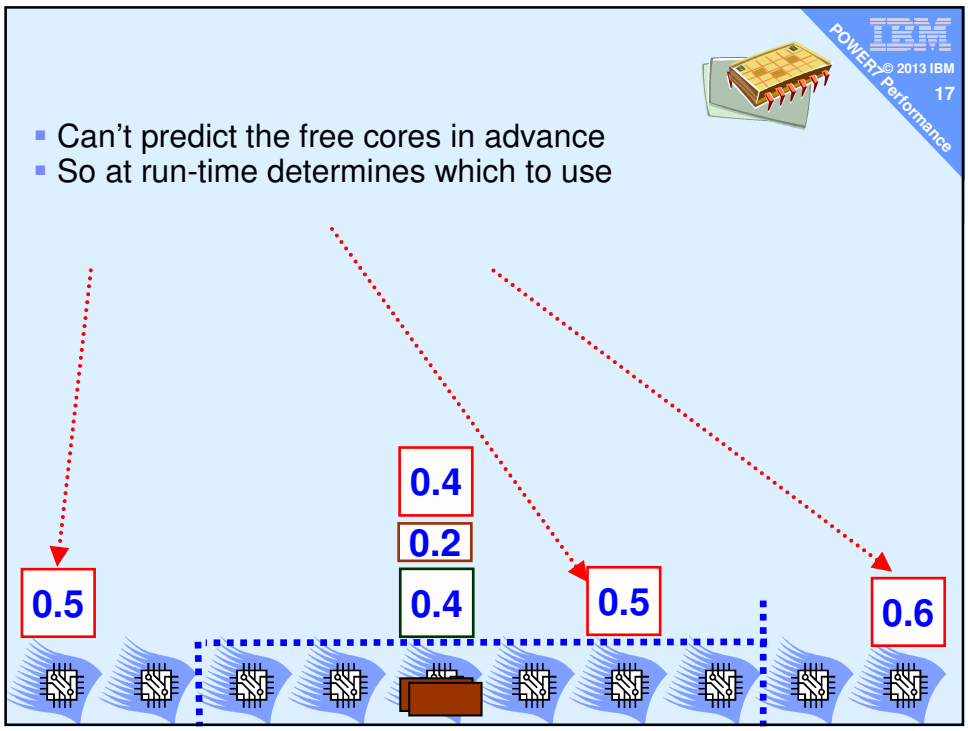
- Memory allocated on the first CPU core (POWER7 chips controller)

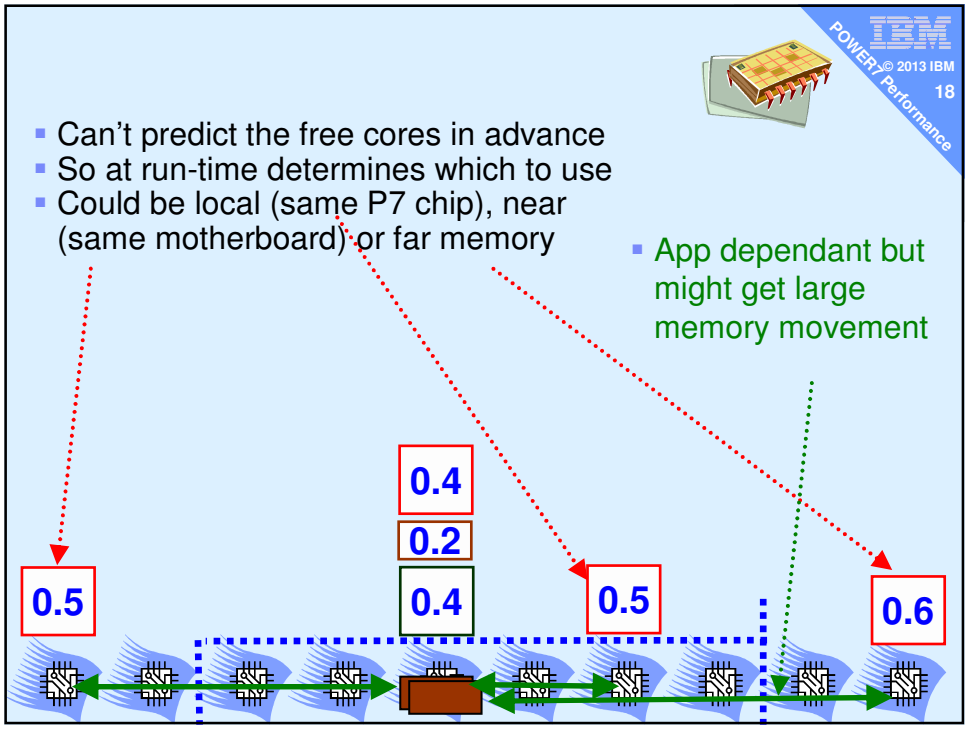


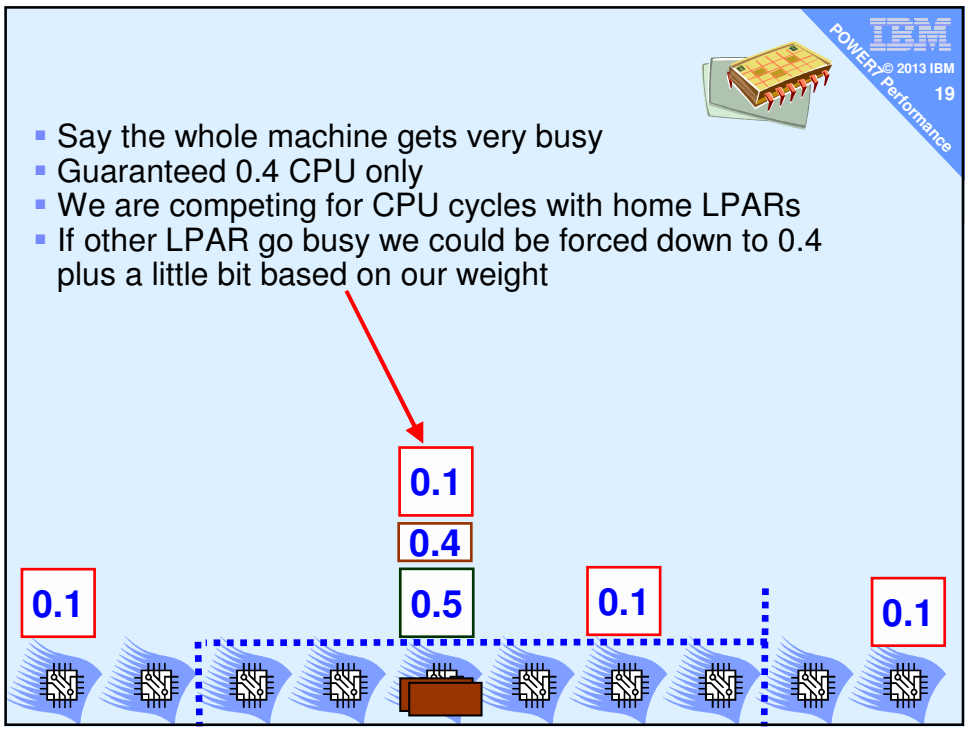


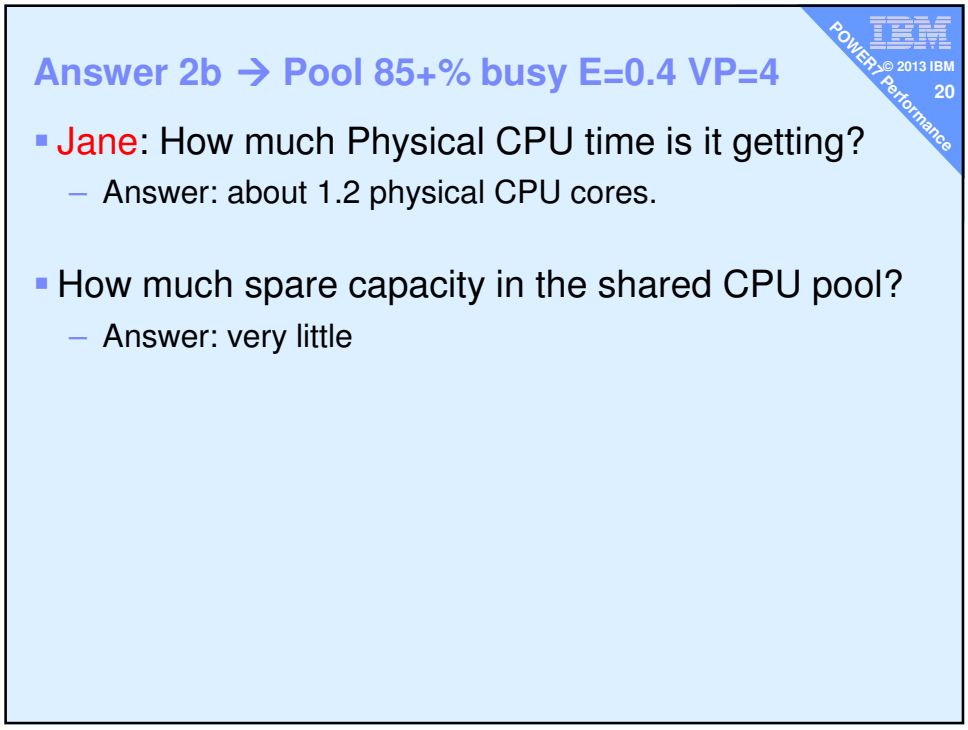
- You start processes that use more than 0.4 cores
- Say the other LPARs on this CPU are busy too
- Hypervisor needs to find more cores for extra CPU cycles
- Uses SRAD to determine the best ones that are free(-ish)









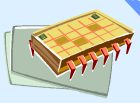


Answer 2b → Pool 85+% busy E=0.4 VP=4

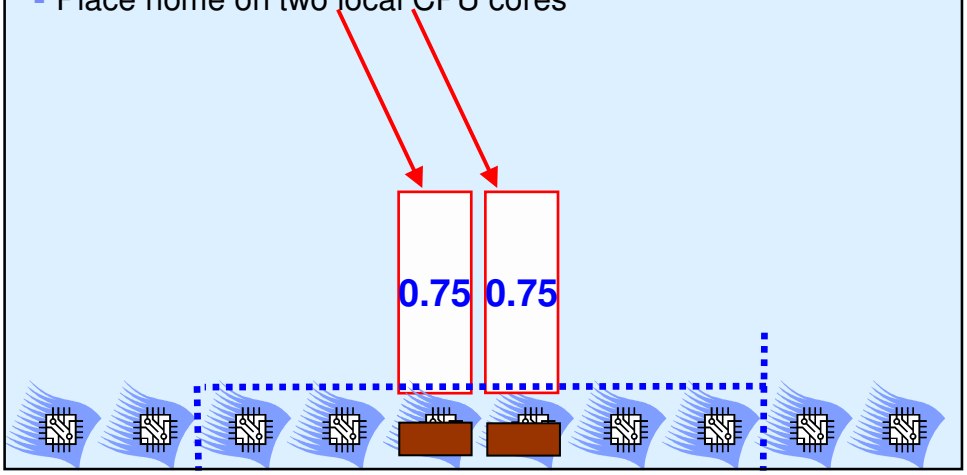
- Jane: How much Physical CPU time is it getting?
  - Answer: about 1.2 physical CPU cores.
- How much spare capacity in the shared CPU pool?
  - Answer: very little
- If this is an important LPAR put the Entitlement up to cover the demand CPU peaks like E=1.5
  - Result: sudden & dramatic leap in performance, responsiveness & zero user problems
- Next consider reducing the VP !!!!
  - Yes I am serious VP is too high = not efficient

FIXED

Eh!!!



- E=1.5 and VP=2 we have 33% headroom
- Hypervisor will do this …
- E = Normal usage so allocate that much now
- Place home on two local CPU cores



Email 5 from Bob

- Can you review our whole machines LPAR settings & recommend what to do?
- Then the details arrive in many bizarre formats
  - Spreadsheets
  - Hand written notes/documents
  - Screen grabs of HMC
  - Camera pictures of HMC screens!!!

Best tools → whole machine review

1. Systems Plans from the HMC
   - Large PDF is a bit of a pain if 100's of LPARs & profiles
2. Reports
   - Hand made or Automated via HMC commands
3. **HMCscanner** → free AIX wiki download
   - Very cool, quick, Java extracts from HMC to a spread sheet

purple-9117-MMB-SN100525P

Name	Status	Mode	Virt/Phys procs Min	Curr	Max	Entitlement Min	Curr	Max	Weight	Sharing Mode	Shared Pool Name	Resv	Max	E:VP %
purple12 IBMi	Off	shared	1	8	2	0.50	6.50	2.00	0	cap	DefaultPool			123.00
purple11-AIX7sp1	Off	shared	1	1	5	0.20	0.40	5.00	0	uncap	DefaultPool			MicroLPAR
purple10_RH55	Off	shared	1	1	2	0.20	0.80	0.50	0	uncap	DefaultPool			MicroLPAR
purple9 fresh	Off	shared	0	0	0	0.00	0.00	0.00	0	uncap	DefaultPool			Off
purple8 SLES11	Off	shared	1	0	2	0.20	0.00	0.50	0	uncap	DefaultPool			Off
purple7-AIX7_TL1_WPAR	On	shared	1	8	16	0.20	8.00	4.00	100	uncap	DefaultPool			100.00
purple6	Off	shared	1	0	10	0.20	0.00	10.00	0	uncap	DefaultPool			Off
purple5-AIX5	Off	shared	1	0	2	0.20	0.00	0.50	0	uncap	DefaultPool			Off
purple4-ISD63-NIM	On	shared	1	2	4	0.20	0.80	4.00	100	uncap	DefaultPool			250.00
purple-hpc	Off	ded	1	0	32					share_idle_procs				Dedicated
purple2-ISD63	On	shared	1	4	4	0.10	2.00	4.00	128	uncap	DefaultPool			200.00
purple1-Blue-Wiki	On	shared	1	14	32	0.50	2.00	4.00	200	uncap	DefaultPool			700.00
diamond9	On	shared	1	4	8	0.10	3.00	8.00	128	uncap	DefaultPool			133.00
mmafull	Off	ded	0	0	0					share_idle_procs				Dedicated
purplevio3	Off	shared	1	0	4	0.20	0.00	2.00	0	uncap	DefaultPool			Off
purplevio2	On	shared	1	2	4	0.20	1.00	2.00	200	uncap	DefaultPool			200.00
purplevio1	On	shared	1	2	4	0.20	1.00	2.00	200	uncap	DefaultPool			200.00
purplevio4	On	shared	1	1	2	0.10	0.50	2.00	128	uncap	DefaultPool			MicroLPAR
purple3 Repository	On	shared	1	3	6	0.10	2.00	4.00	200	uncap	DefaultPool			150.00

	Size	Assigned	Available
Active Physical Cores	32		
Dedicated Cores		0	
Shared Pool	32	28.00	4.00
Virtual Processors		50.00	

Answer 5 → whole machine review

- I normally work on “big iron” = big LPARs
  - but now seeing many micro-partition setups
- I have had to rethink what to recommend
- Lots of:
  - E=0.2 and VP=2
  - E=0.3 and VP=3
  - E=0.4 and VP=4
  - …
  - = Ghastly but Common

Answer 5 → whole machine review

- I can see the pressure
  - Loads of LPARs but limited physical processing units
  - VP is free, allocate lots of safety margins and then no need to monitor
  - = Bad thinking
- End up with total VP up to 10 times total physical CPU cores

Answer 5 → whole machine review

Just because IBM says: “You can do this”
- Does not make it a good idea
- Specially doing it everywhere and every time!

But but but IBM promised this over-commit was OK
- True it is OK

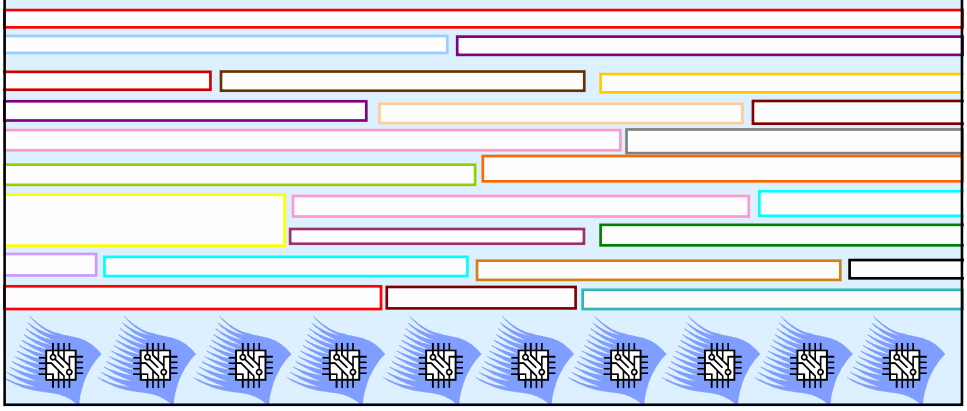
Answer 5

Just because IBM says: “You can do this”
- Does not make it a good idea
- Specially doing it everywhere and every time!

But but but IBM promised this over-commit was OK
- True it is OK but it is just like:
  - Over-commit virtual → physical memory = paging & everyone hurts!
  - Over-committing disk space with thin provisioning is OK … provided users don’t all demand their max disk space (another nightmare)
- If lots of LPARs demand their all their VP then they have to compete for CPU cycles
- Only Entitlement is guaranteed plus memory affinity side effects hurts too

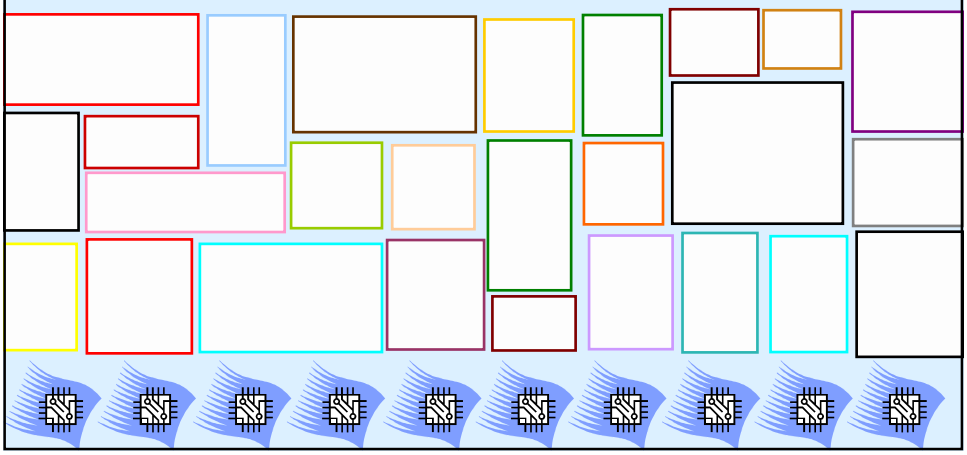
Low Entitlement + High Virtual Processors

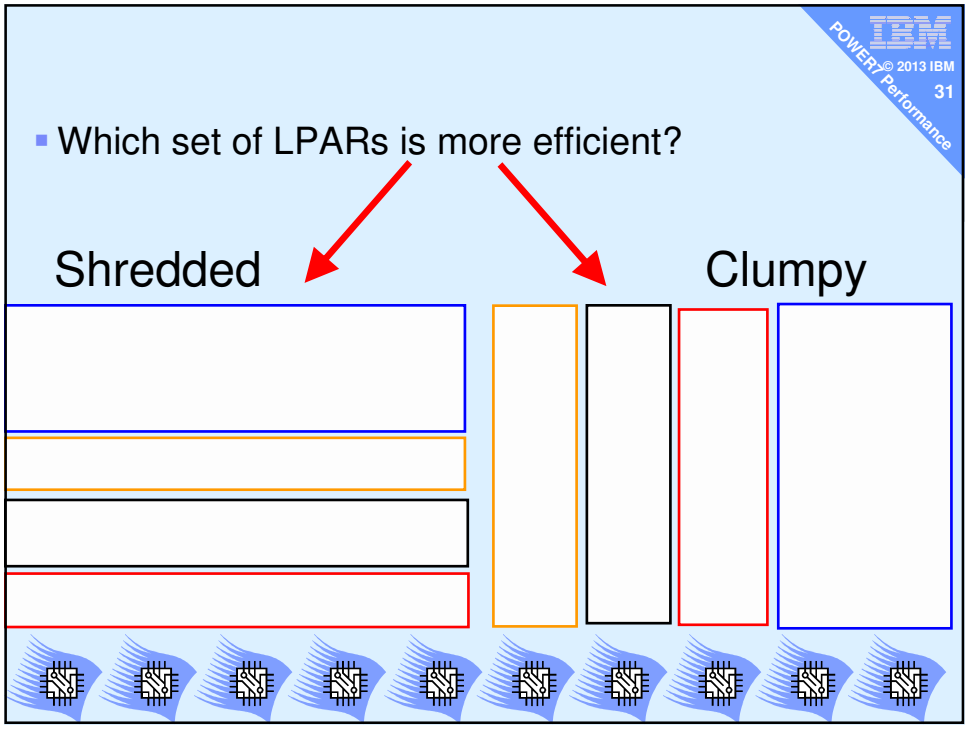
- Good for flexibility but I call this LPAR “shredding”
- Can cause unnecessary CPU cache misses & memory bus transfers

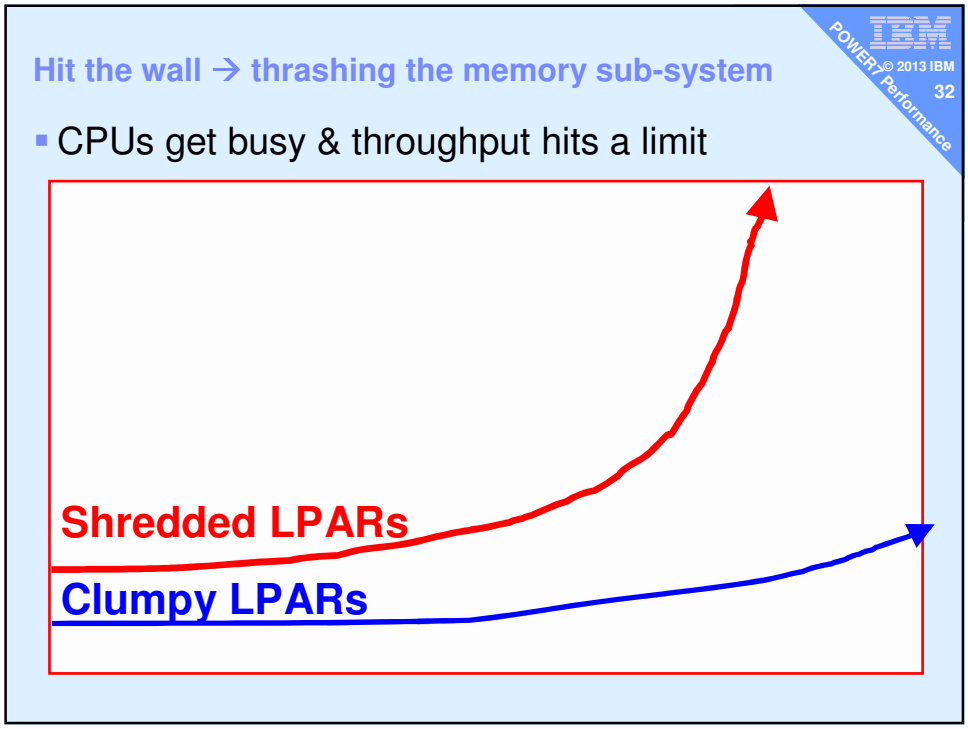


Same Entitlement + Lower Virtual Processors

- More work = performance monitoring
  - Set E (normal peak) and VP correctly (a little higher)
- Factor of ten better affinity “Clumpy”

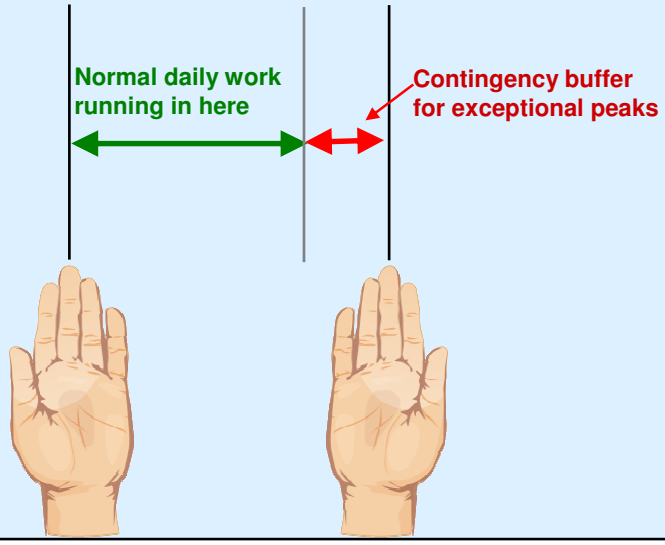




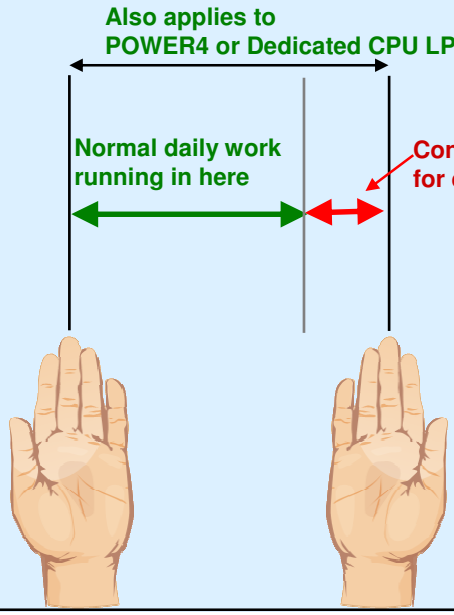


Revisit - Entitlement to Virtual Processor ratio

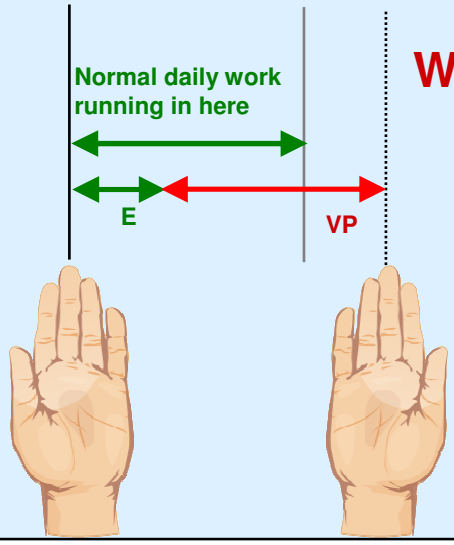
"Goode olde days" – 80:20 rule or Dedicated CPU LPAR



Shared CPU LPAR but Capped with 80:20 rule



Shared CPU LPAR but Uncapped
- what gets implemented all too often
- actually worse as E is < typical CPU use

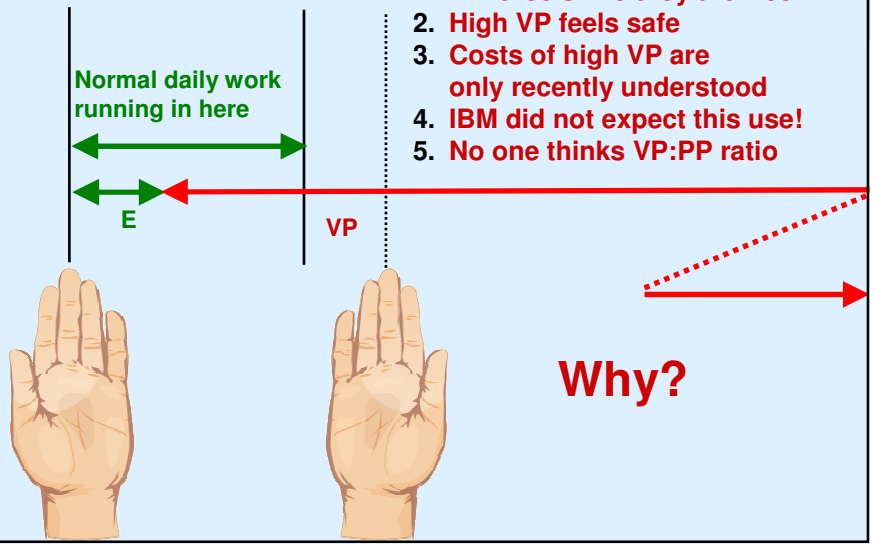


Why?

Answer: we run out of Entitlement
(HMC: Desired Processing units)
as we over-commit the CPUs

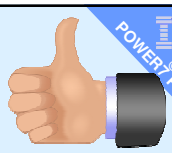
Shared CPU LPAR but Uncapped
- what gets implemented all too often
- VP = 10 times E

1. VP it feels like they are free
2. High VP feels safe
3. Costs of high VP are only recently understood
4. IBM did not expect this use!
5. No one thinks VP:PP ratio



Why?

New Role of Thumb (ROT)

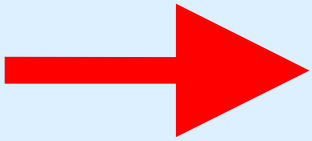


- Small shared uncapped LPARs hard to assign sensible VP numbers

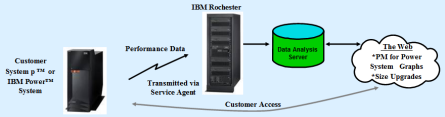
- Rule E:VP ratio — No real choice as 1 is the minimum
  - 0.05 to 0.6 VP=1 — VP head room = 100% to 33%
  - 0.7 to 1.4 VP=2 — VP head room = 65% to 30%
  - 1.5 to 2.3 VP=3 — VP head room = 100% to 24%
  - 2.4 to etc.

Policy:
- E = regular in busy peaks = guaranteed
- VP allows some head room like ~25-50% more

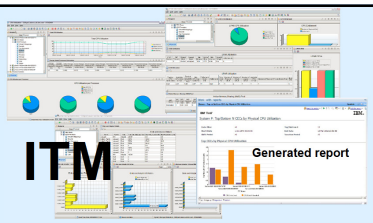
Hence the need for:
whole machine config docs
whole machine monitoring

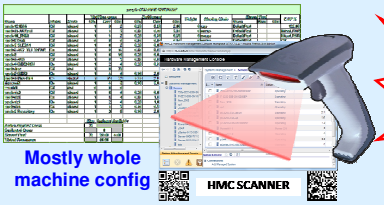


Whole Machine Monitoring

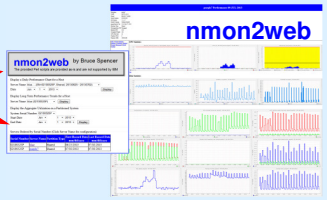


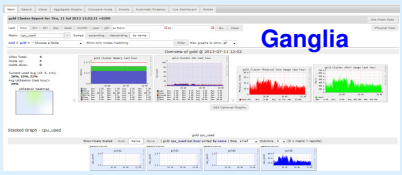
PM for Power Systems (service)

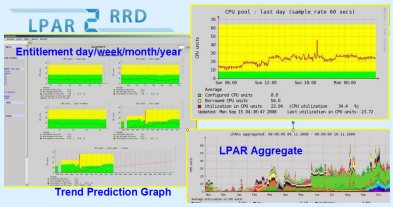




The popular tools I find at my customers

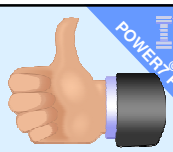






Also many 3rd party Performance Tools

Rules of Thumb

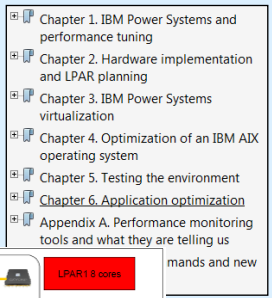


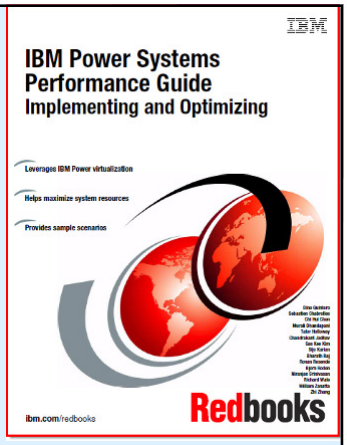
- Production LPARs
  - Entitlement (E) to cover your regular peaks = SLA
  - Virtual Processor (VP) a little bonus to handle short peaks
  - LPAR level check the E : VP ratio below 125%
  - Monitor/Alert on over E use to avoid issues

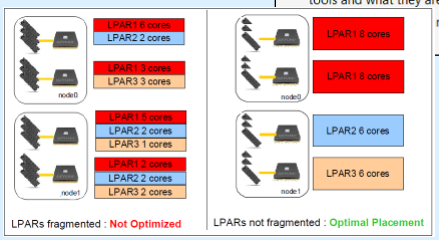
- Over committing your CPUs?
  - LPAR level check the E : VP ratio
  - Box level check the Total VP : Physical CPU ratio
  - Monitor all LPARs all the time for anomalies
  - Monitor unused Shared CPU pool (app)
    go proactive when unused pool below 15% or 3 CPUs

Power Systems Performance Guide Implementing & Optimizing

- Updated recently
- 360 pages
- SG24-8080







Shredded = not optimised
Clumpy = optimal

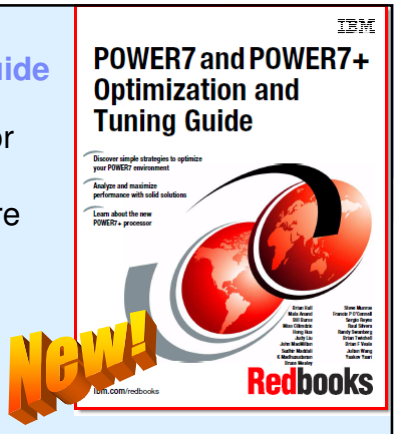
- http://www.redbooks.ibm.com/abstracts/sg248080.html

POWER7 Optimization & Tuning Guide

A single “first stop” definitive source for a wide variety of general information and guidance, referencing other more detailed sources on particular topics

- Redbook SG24 8079

- Lots of guru level
  Advanced Technical content



- http://www.redbooks.ibm.com/abstracts/sg248079.html

Email 6 - Sue asks for Hints & Tips when using rPerfs for Sizing new machines or server consolidation

Email 6 - rPerf Sizing hints

- Website about rPerf to POWER7
  - http://www.ibmsystemsmag.com/aix/tipstechniques/Migration/rperf_metric/

1. Sizing by adding up old box rPerf's scaled to LPAR and scaled down based on utilisation
2. Add guestimate of new workloads
3. Add guestimate of growth
4. Add comfort factor

A. Find suitable matching box or boxes
B. Decided sensible config
C. Ask for price

BUT

- There are a large number of assumptions being made here
- These can catch you out

Ten Golden rules of using rPerf for sizing (avoiding a performance mess-up by assuming to much)

1. **Highly threaded workloads** - 2 to 3 times total SMT threads
2. **Well tuned system** - not out of the box settings
3. **Full Spec RAM** - all slots used & lots of memory
4. **No Disk Issues**
5. **No Network Issues**
6. **Current app, RDBMS, middle-ware & web servers software levels** - not what the old box ran
7. **Latest AIX with Service Packs -** like benchmarks
8. **Large LPARs -** rPerfs NOT based on micro-LPARs
9. **Firmware is Current**
10. **Bug Free** - user MUST upgrade FW, AIX and Apps.

- Find this info on http://tinyurl.com/AIXpert

POWER7 Performance FAQ Summary

1. You need to monitor SMT use
2. Set Entitlement to typical use & monitor/tune it
3. Lower the VP to get SMT threads working for you
4. Tool up for machine monitoring
5. rPerf sizing is fine but watch those assumption

- Watch those ratios
  - LPAR Entitlement : Virtual Processor
  - Machine Total VPs : Physical CPUs in the Pool

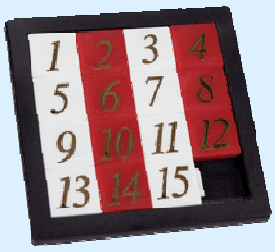
Four “Get out of Jail Free” cards

- VIOS, LPAR and Java Advisors
- Free download
  - VIOS now part of VIOS → see the “part” command
- Run the advisor data collector
- Read the report for hints and Best Practice
- More to come

Dynamic Platform Optimizer (DPO)

POWER Virtual Machines before DPO →

After DPO →



Cool right ☺

If you suspect bad placement! So what can you do?

- Use lssrad -av to build a picture
- Restart the machine from cold = total rethink then start large & important LPARs first

  - The Hypervisor does the right thing
  - Can be painful to schedule
- Start LPAR with 0.1 CPU & 1GB RAM profile & no adapters then restart the regular profile

  - This gets the Hypervisor to rethink placement
- Use DPO – needs 760+ firmware*
- Use Affinity Group – needs 730+ firmware*
- If you have bad performance – raise a PMR

* also needs matching HMC version

Scaled Throughput

Scaled Throughput?

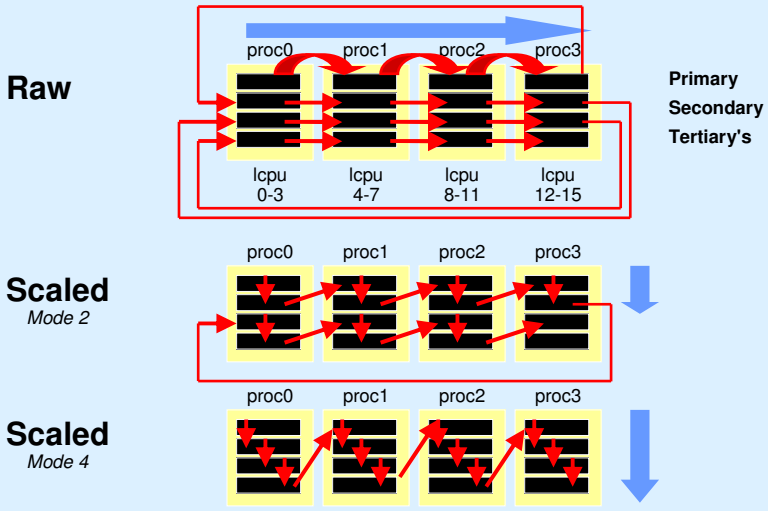
POWER7 & POWER7+ with
AIX 6.1 TL08 & AIX 7.1 TL02

- It will dispatch more SMT threads to a VP core before unfolding additional VPs
- Considered a bit more like POWER6 unfolding but is a *generalization*, not a technical statement

What is Scaled Throughput?

- **Raw** provides the highest per-thread throughput and best response times at the expense of activating more physical core
- **Scaled** provides the highest core throughput at the expense of per-thread response times and throughput.
  It also provides the highest system-wide throughput per VP because tertiary thread capacity is “not left on the table.”

Raw vs Scaled Throughput



Scaled Throughput: Tuning

- Not restricted, but
  anyone experimenting without understanding may suffer significant performance impacts

- `schedo -p -o vpm_throughput_mode=`
  - `0` Legacy Raw mode (default)
  - `1` "Enhanced Raw" mode with a higher threshold than legacy
  - `2` Scaled mode, use primary and secondary SMT threads
  - `4` Scaled mode, use all four SMT threads

- Dynamic tunable

Scaled Throughput: Workloads

- **Workloads**
  - Workloads with many light-weight threads with short dispatch cycles and low IO (the same types of workloads that benefit well from SMT)
  - Customers who are easily meeting network & I/O SLA's may find the tradeoff between higher latencies & lower core consumption attractive
  - Customers who will not reduce over-allocated VPs & prefer to see behavior similar to POWER6
- **Performance**
  - *It depends,* we can't guarantee what a particular workload will do
  - Mode 1 may see little or no impact but higher per-core utilization
  - Workloads that do not benefit from SMT & use Mode 2 or Mode 4 could easily see double-digit per-thread performance degradation (higher latency, slower completion times)

Are you keeping up to date?

mr_nmon on twitter

- Only used to POWER / AIX news, technical content, hints, tips and links

YouTube™ 125 techie hands-on videos on **YouTube** at http://www.youtube.com/user/nigelargriffiths

AIXpert Blog

- Lots of mini articles & thoughts
- http://tinyurl.com/AIXpert

AIX & PowerVM **Virtual User Groups** → ~monthly webinars