What's Going Wrong: LPAR Weights

Brian K. Wade, Ph.D.
IBM Corporation
bkw@us.ibm.com
Version 2022-10-06.1

Abstract

In my job I often look at MONWRITE data from customer systems. Recently I have seen a rash of data showing me CPCs where the LPAR weights were not set correctly. The consequences were less-than-optimal configurations and in some cases just plain poor operation.

In this presentation I review basic concepts about LPAR weight, the notion of entitlement, how to recognize configuration problems, and how to correct them. I also point out some techniques, tools, and references that might help.

Agenda

- Basic concepts: weight, entitlement, polarization
- Ways things go wrong: problems and solutions
- Techniques: suggested practices
- Tools: some gadgets that might help
- Summary

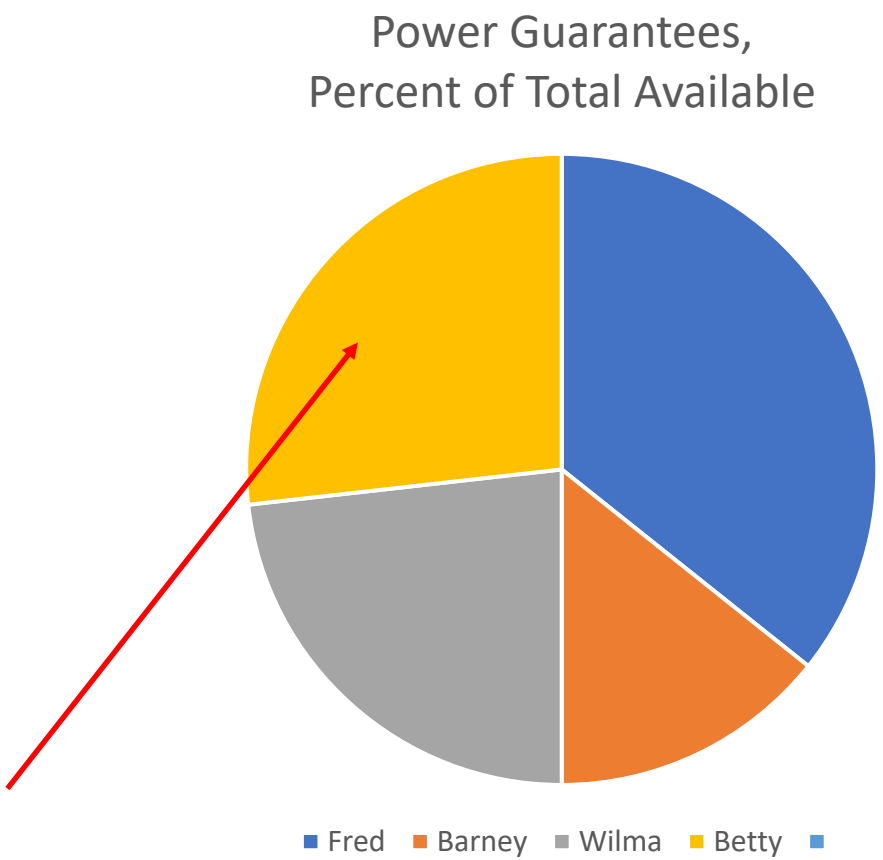
Basics

What Are LPAR Weights?

- The PR/SM hypervisor distributes computing power to the LPARs it is managing
- The weights control how much power each LPAR is guaranteed to be able to use whenever it wants.
- In other words, the weights inform PR/SM of how to compromise when there is not enough power to satisfy all partitions' demands.

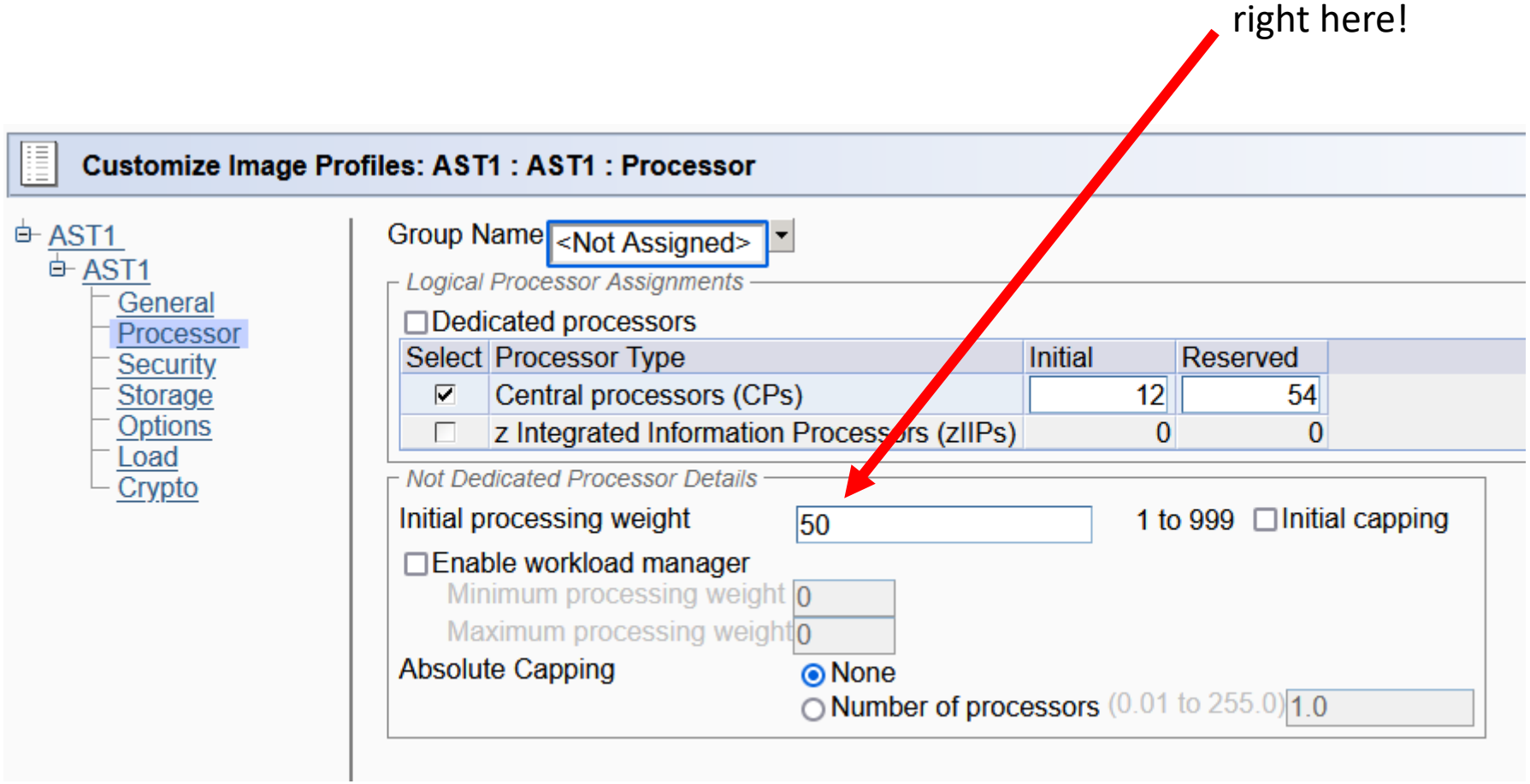
Partition	Weight
FRED	100
BARNEY	40
WILMA	65
BETTY	75
Sum	280

Betty% = 100 x (75/280) = 26.8



Where Do We Set the Weights? (partition down)

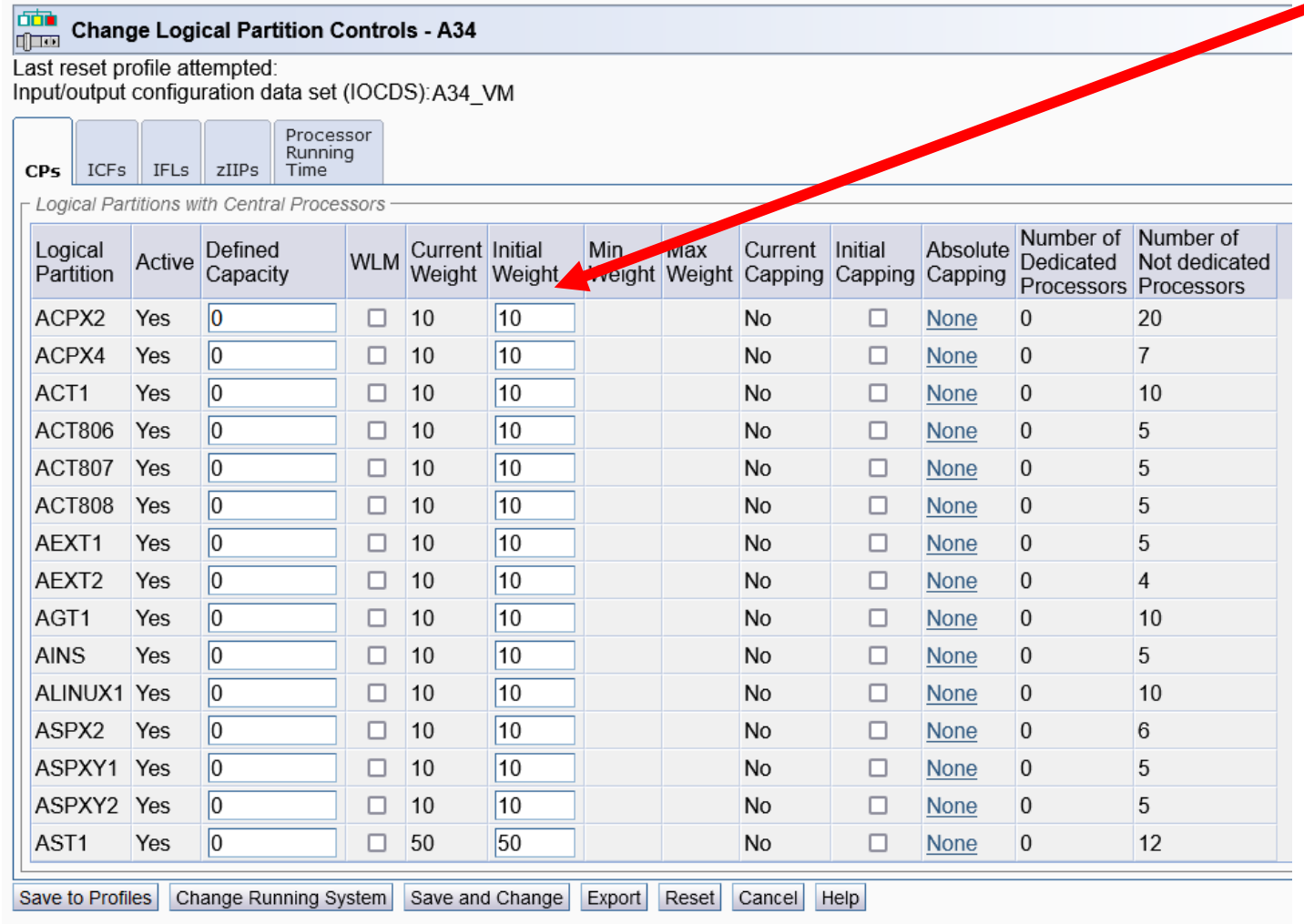
- HMC or SE: Image activation profile



Where Do We Set the Weights? (partition up)

- HMC or SE: Change LPAR Controls

right here!



Change Logical Partition Controls - A34

Last reset profile attempted:
Input/output configuration data set (IOCDS):A34_VM

CPs | ICFs | IFLs | zIIPs | Processor Running Time

Logical Partitions with Central Processors

Logical Partition	Active	Defined Capacity	WLM	Current Weight	Initial Weight	Min Weight	Max Weight	Current Capping	Initial Capping	Absolute Capping	Number of Dedicated Processors	Number of Not dedicated Processors
ACPX2	Yes	0	☐	10	10			No	☐	None	0	20
ACPX4	Yes	0	☐	10	10			No	☐	None	0	7
ACT1	Yes	0	☐	10	10			No	☐	None	0	10
ACT806	Yes	0	☐	10	10			No	☐	None	0	5
ACT807	Yes	0	☐	10	10			No	☐	None	0	5
ACT808	Yes	0	☐	10	10			No	☐	None	0	5
AEXT1	Yes	0	☐	10	10			No	☐	None	0	5
AEXT2	Yes	0	☐	10	10			No	☐	None	0	4
AGT1	Yes	0	☐	10	10			No	☐	None	0	10
AINS	Yes	0	☐	10	10			No	☐	None	0	5
ALINUX1	Yes	0	☐	10	10			No	☐	None	0	10
ASPX2	Yes	0	☐	10	10			No	☐	None	0	6
ASPXY1	Yes	0	☐	10	10			No	☐	None	0	5
ASPXY2	Yes	0	☐	10	10			No	☐	None	0	5
AST1	Yes	0	☐	50	50			No	☐	None	0	12

Save to Profiles | Change Running System | Save and Change | Export | Reset | Cancel | Help

then this

"Save and Change" to save to image activation profile and change the running system

The Notion of Entitlement

- **Entitlement:** what you can use whenever you want
- We calculate each LPAR's **entitlement** from:
  - # of shared physical cores, and
  - the LPAR's weight, and
  - the sum of the weights
- Entitlement is expressed in units of **cores' worth of power.**
- Usually we multiply by 100.

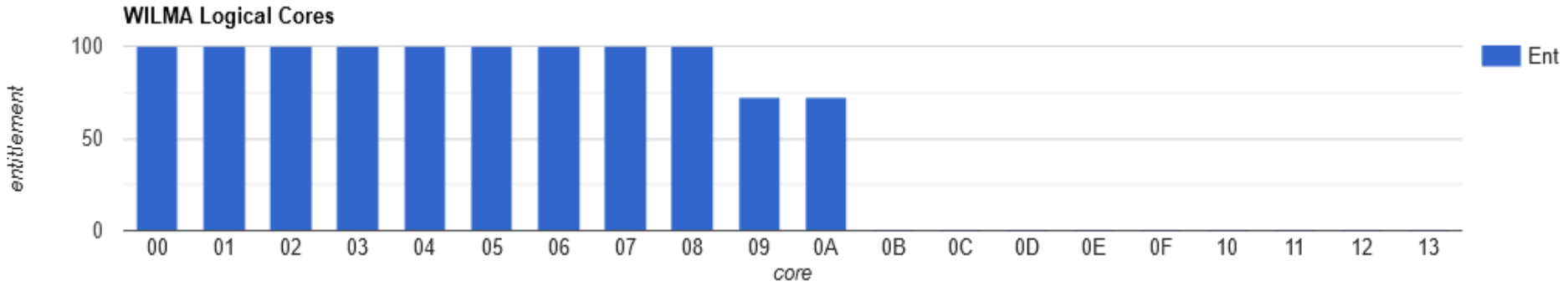
Shared Physical Cores	45		
Partition	Weight	% of Sum	Entitlement
FRED	100	35.7	16.1
BARNEY	40	14.3	6.4
WILMA	65	23.2	10.4
BETTY	75	26.8	12.1
sums -->	280	100.0	45.0

For example: E(FRED) = 45 * (100/280) = 16.1 (we usually x100 = 1607)

Polarization: Effect of Entitlement on Logical Cores

- PR/SM spreads entitlement unevenly over the logical cores
- PR/SM places the entitled cores in the machine topology in such a way that the partitions are less likely to interfere with one another's caches
- Operating system should try to run on only its entitled cores whenever possible

Partition	Weight	Entitlement	Cores	VHs	VMs	VLs
FRED	100	1607.1	20	15	2@53.6	3
BARNEY	40	642.9	20	5	2@71.4	13
WILMA	65	1044.6	20	9	2@72.3	9
BETTY	75	1205.4	20	11	2@52.7	7



Ways Things Go Wrong

Problem 1: Unusable Entitlement

- LPAR13 has 22 logical cores but entitlement 2667
- It cannot possibly run 2667% core-busy
- Other LPARs are deprived of entitled power
  - Some VLs could have been VMs or VHs
  - Some VMs could have been VHs
- **u** finds these for you right away

From Perfkit FCX306 LSHARACT

Core counts:	CP	ZAAP	IFL	ICF	ZIIP
Dedicated	0	0	0	0	0
Shared physical	1	0	108	0	0
Shared logical	1	0	322	0	0

(edited to show IFL cores only)

Core Type	Partition Name	Core Count	Load Max	LPAR Weight	Entlment	Cap	AbsCap	GrpCapNm	GrpCap	<CoreTotal,%> Busy	Excess	Core Conf	
IFL	LPAR01	64	6400	10	133.3	No	...	...	...	.2	.0	o	
IFL	LPAR02	1	100	10	133.3	No	...	...	...	.1	.0	u	<--
IFL	LPAR03	30	3000	60	800.0	No	...	...	...	81.8	.0	o	
IFL	LPAR04	20	2000	60	800.0	No	...	...	...	57.5	.0	o	
IFL	LPAR05	20	2000	60	800.0	No	...	...	...	135.3	.0	o	
IFL	LPAR06	20	2000	60	800.0	No	...	...	...	82.2	.0	o	
IFL	LPAR07	20	2000	60	800.0	No	...	...	...	58.9	.0	o	
IFL	LPAR08	20	2000	60	800.0	No	...	...	...	199.6	.0	o	
IFL	LPAR09	12	1200	60	800.0	No	...	...	...	1.4	.0	o	
IFL	LPAR10	30	3000	60	800.0	No	...	...	...	1.2	.0	o	
IFL	LPAR11	30	3000	60	800.0	No	...	...	...	1.4	.0	o	
IFL	LPAR12	4	400	10	133.3	No	...	...	...	25.0	.0	o	
IFL	LPAR13	22	2200	200	2666.7	No	...	...	...	602.1	.0	u	<--
IFL	LPAR14	6	600	10	133.3	No	...	...	...	2.9	.0	o	
IFL	LPAR15	8	800	10	133.3	No	...	...	...	176.6	43.3	o	
IFL	LPAR16	7	700	10	133.3	No	...	...	...	7.4	.0	o	
IFL	LPAR17	8	800	10	133.3	No	...	...	...	1.7	.0	o	

Solution 1: Change the Weights

- I changed the weights of several LPARs just a little bit
- This left LPAR02 and LPAR13 fully entitled and increased the entitlements of the other LPARs

Phys		LPAR	Cores	Old W	Old E		New W	New E
108		lpar01	64	10	133.3333		10	143.8083
		lpar02	1	10	133.3333		7	100.6658
		lpar03	30	60	800		59	848.4687
		lpar04	20	60	800		59	848.4687
		lpar05	20	60	800		59	848.4687
		lpar06	20	60	800		59	848.4687
		lpar07	20	60	800		59	848.4687
		lpar08	20	60	800		59	848.4687
		lpar09	12	60	800		59	848.4687
		lpar10	30	60	800		59	848.4687
		lpar11	30	60	800		59	848.4687
		lpar12	4	10	133.3333		10	143.8083
		lpar13	22	200	2666.667		153	2200.266
		lpar14	6	10	133.3333		10	143.8083
		lpar15	8	10	133.3333		10	143.8083
		lpar16	7	10	133.3333		10	143.8083
		lpar17	8	10	133.3333		10	143.8083

Problem 2: Excess Logical Cores

- LPAR03 has 11 logical cores but entitlement of only 406.4
- Its logical cores:
  - 3 VH
  - 2 VM of 53 each
  - **6 VL**
- **o** finds these for you right away
- Notice column **"Excess"**. LPAR03 is nontrivially running on unentitled power.

From Perfkit FCX306 LSHARACT

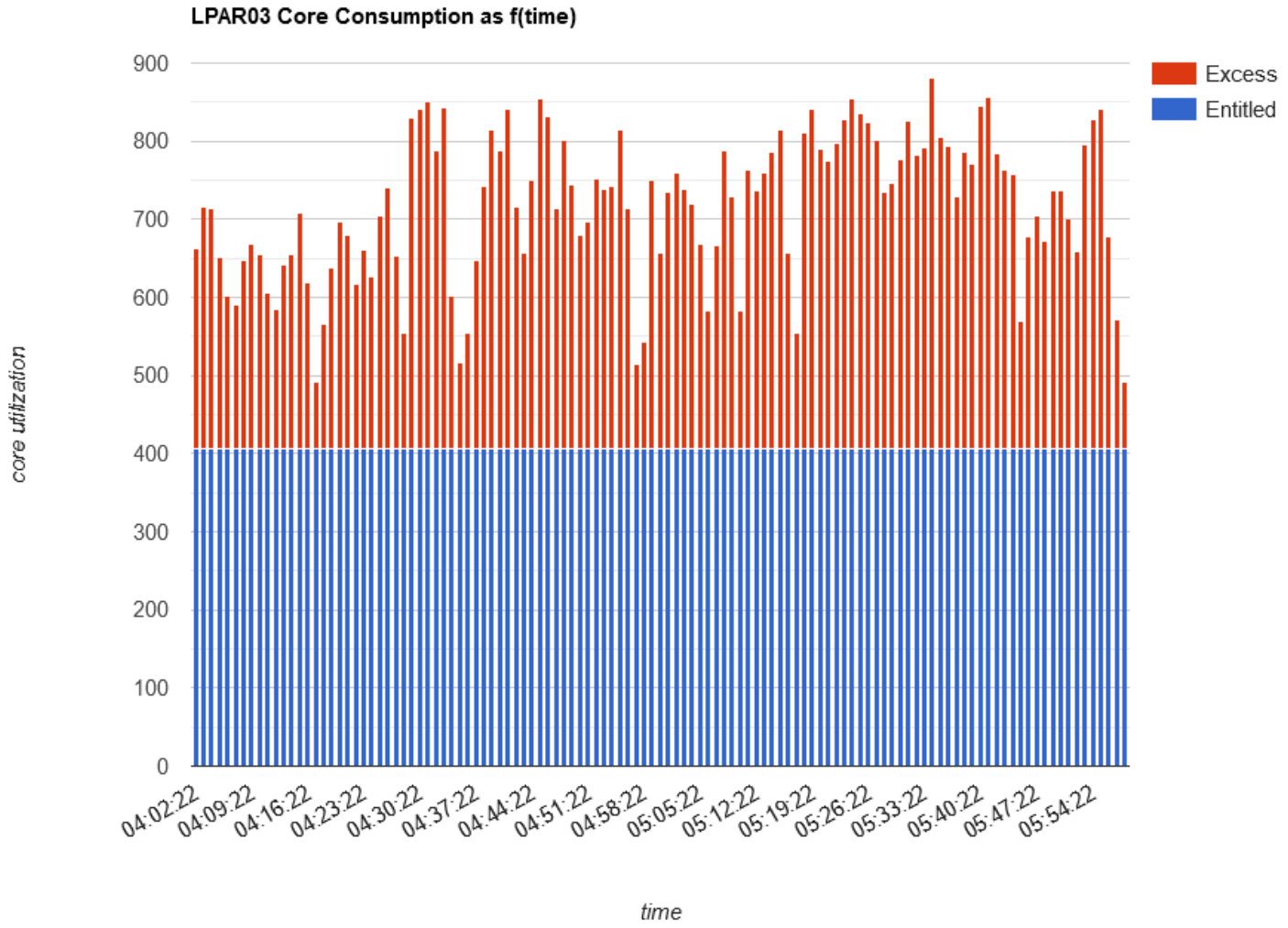
LPAR Data, Collected in Partition LPAR03

Core counts:	CP	ZAAP	IFL	ICF	ZIIP
Dedicated	0	0	0	0	0
Shared physical	0	0	29	0	0
Shared logical	0	0	42	0	0

Core Type	Partition Name	Core Count	Load Max	LPAR Weight	Entlment	Cap	AbsCap	GrpCapNm	GrpCap	<CoreTotal,%> Busy	Excess	Core Conf
IFL	LPAR01	2	200	9	26.1	No	...	...	...	1.2	.0	o
IFL	LPAR02	29	2900	850	2467.5	No	...	...	...	1507.6	.0	o
IFL	LPAR03	11	1100	140	406.4	No	...	...	...	717.7	311.3	o

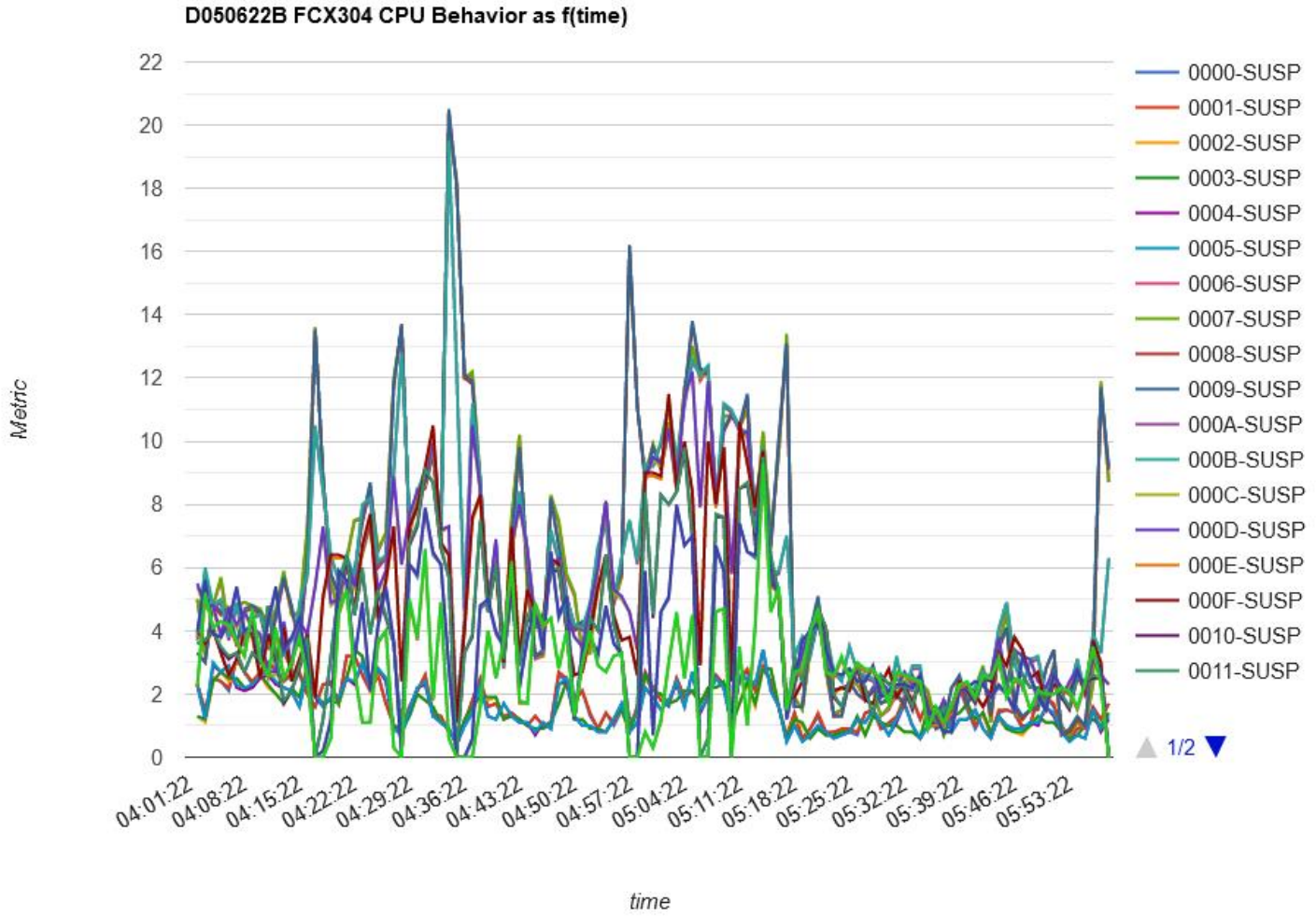
LPAR03 and Unentitled Power

- I produced this from FCX306 INTERIM LSHARACT, plotting the "Entlment" and "Excess" values for LPAR03
- We can see LPAR03 is habitually getting its work done on VLs
- This is bad because VLs are exposed to:
  - PR/SM dispatch delay
  - Suddenly having no power at all
- This partition probably needs more entitlement



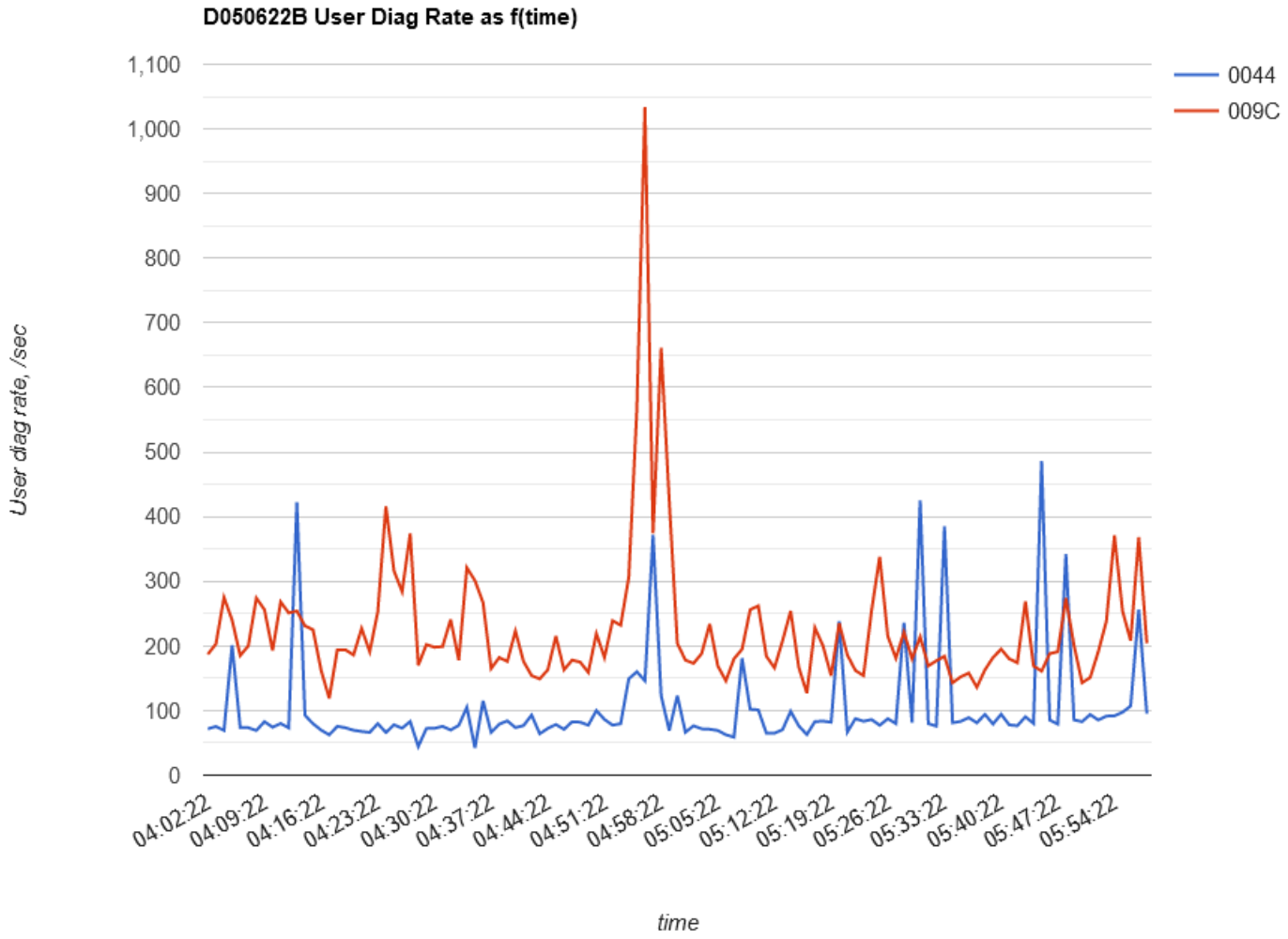
Hazard of Running on Unentitled Power

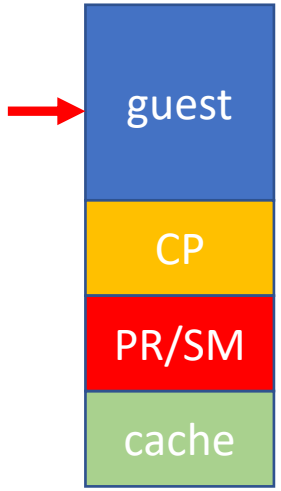
- PR/SM does not guarantee a VM or VL a premium dispatch experience
- Workload suffers "suspend time"
- "Suspend time" is time the logical core wanted to run but PR/SM didn't run it
- The workload gets delayed



Suspend Time: Consequence for Guests

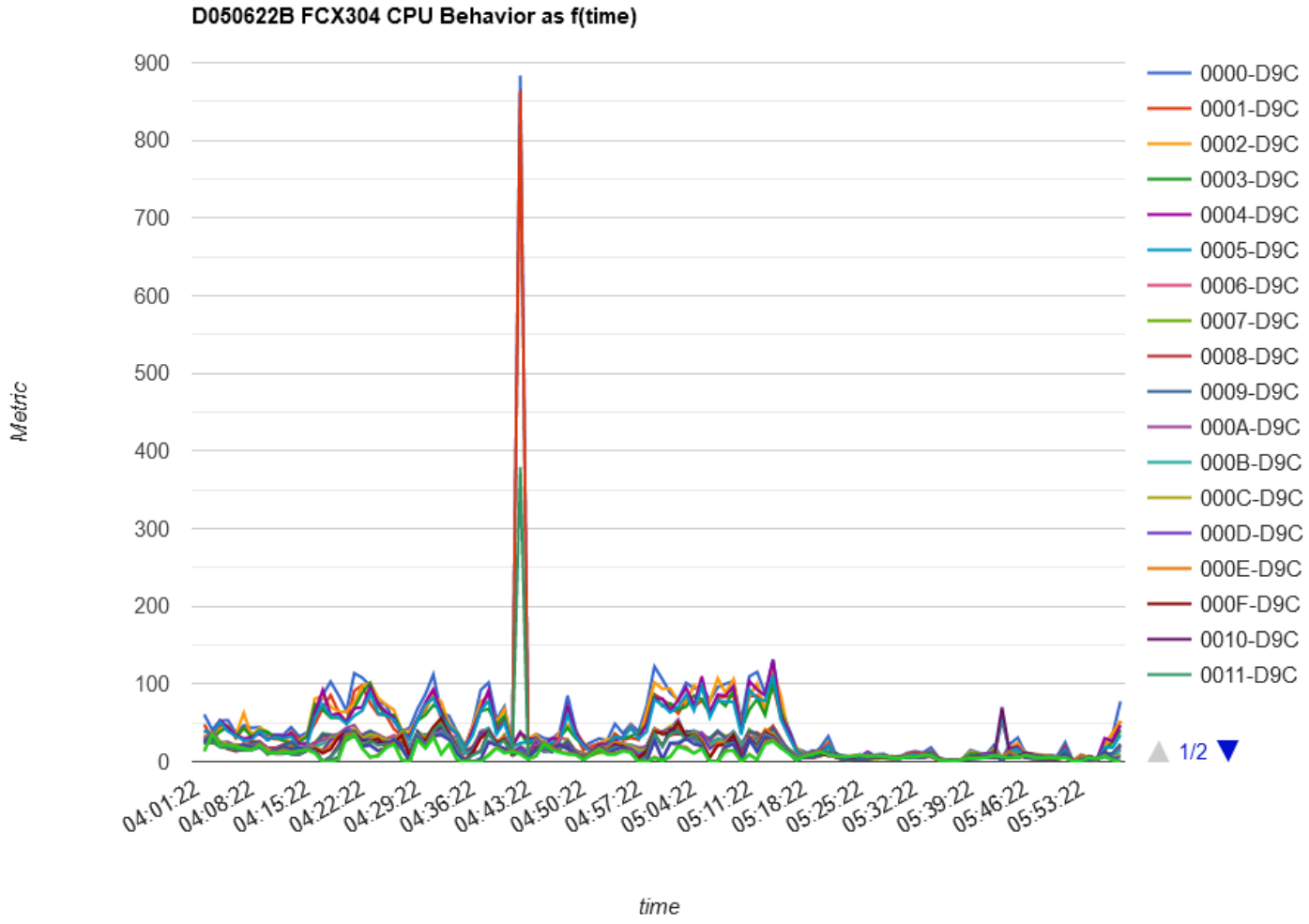
- The guest Diag x'9C' rate peak exactly lines up with the moment at which logical CPUs 06, 07, 08, and 09 experienced a suspend peak
- Those logical CPUs are vertical-mediums
- One or more virtual CPUs were stuck on those logical CPUs
- Their friends tried to unstick them
- This is pure overhead
  - Guest overhead: looping and issuing Diag x'9C'
  - CP overhead: excess simulation

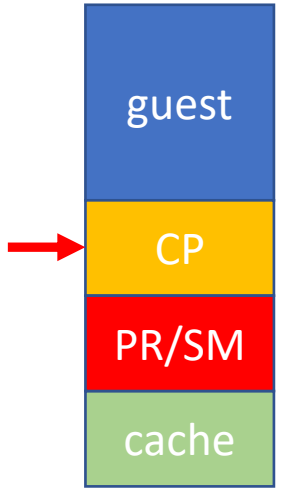




Suspend Time: Consequence for CP

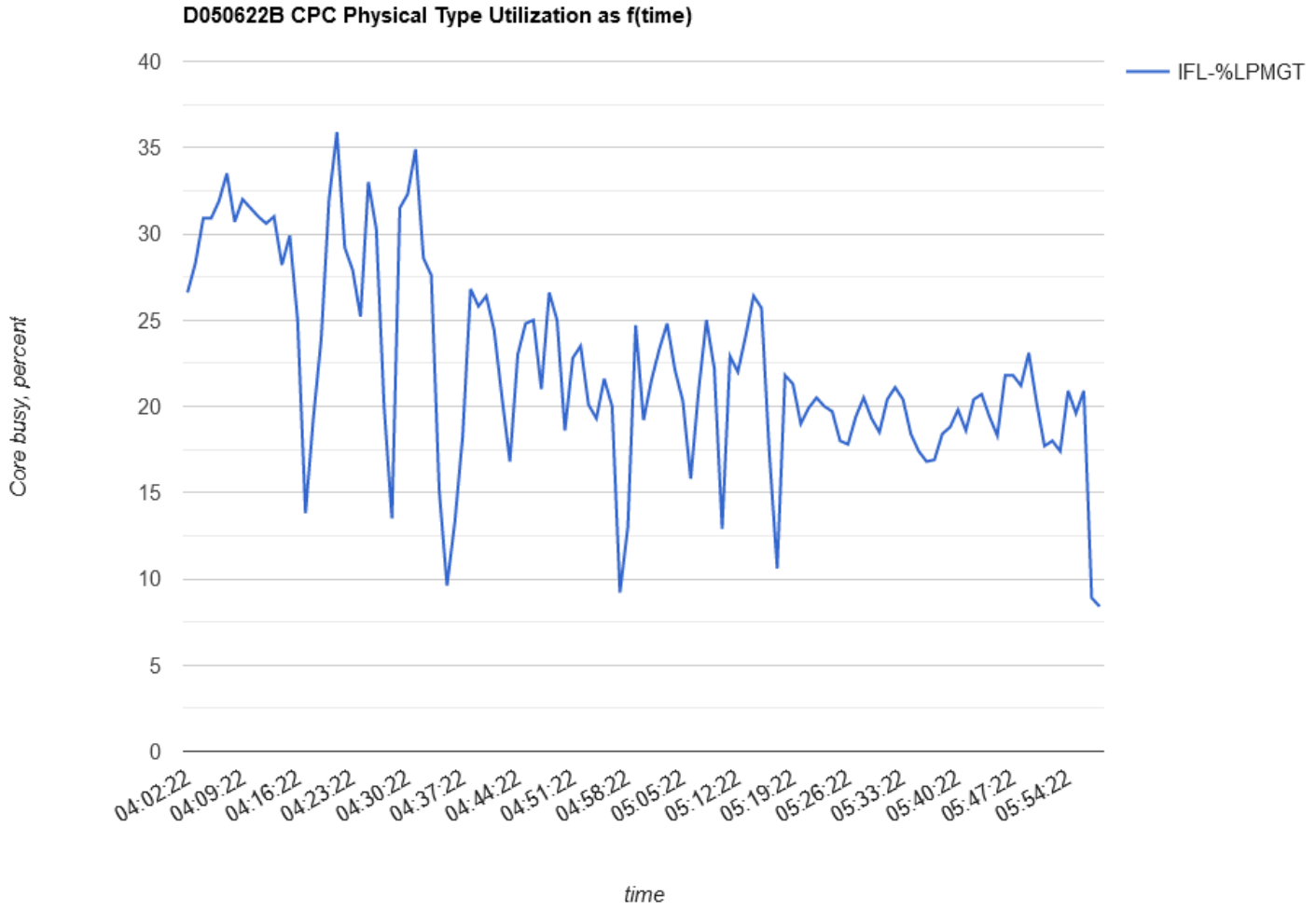
- CP has spin locks too and itself issues Diag x'9C' to wake up nonrunning holders
- In a partition running on entitled power, the CP Diag x'9C' rates are usually near zero
- When CP has to use Diag x'9C' nontrivially, CP efficiency drops
- This is pure overhead
  - CP spinning and issuing Diag x'9C'
  - PR/SM handling the excess simulation

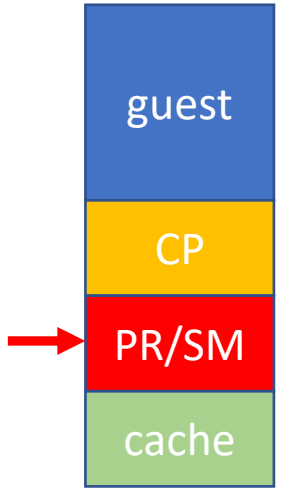




Suspend Time: Consequence for PR/SM

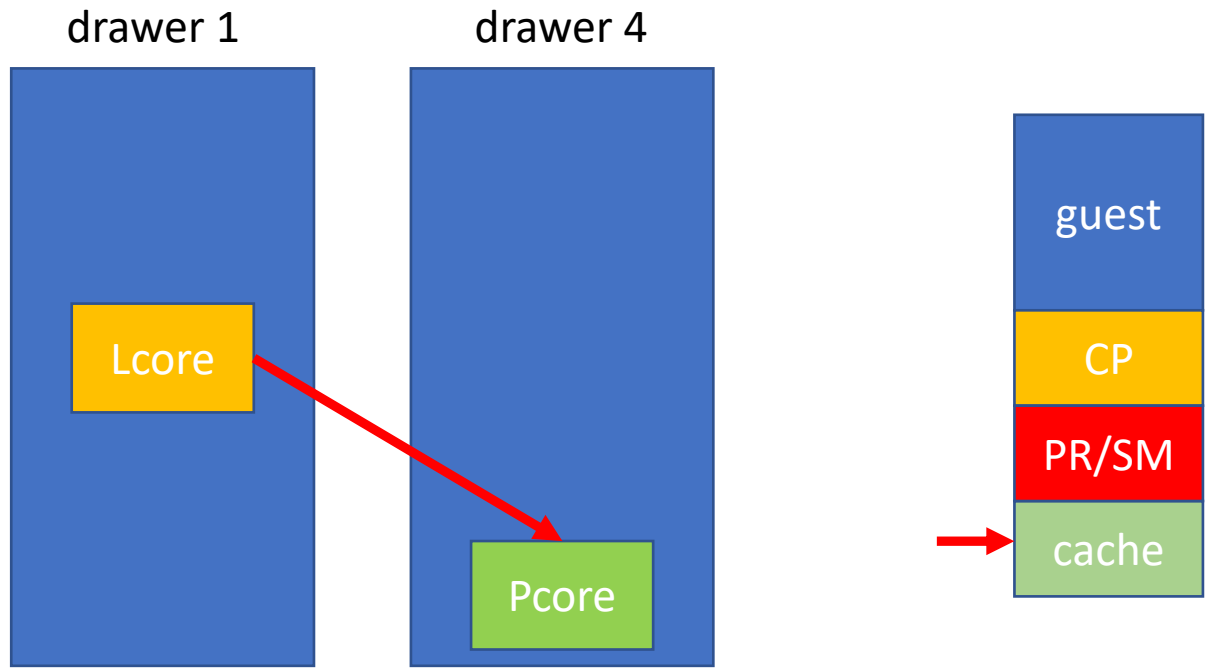
- PR/SM has to field all those Diag x'9C' invocations CP is doing
- Dispatching VLs is more complex than dispatching VHs
- All of this increases PR/SM overhead
- In a well-configured system the PR/SM overhead is about zero
- This CPC is using about 0.23 of a core on PR/SM overhead





Cache Impact of Using Unentitled Power

- Non-VH logical cores:
  - Have a preferred dispatch location, but
  - They are exposed to being dragged elsewhere to be run
- This dragging can decrease cache effectiveness, thereby increasing CPI, thereby decreasing efficiency
- This too is pure overhead



Solution 2: Adjust Something!

- **Are weights too small?**
  - Partitions should be getting their work done on entitled power
  - Set entitlements correctly for demands of workload
- **Are logical core counts too large?**
  - Set logical core counts according to entitlements
  - No more than 1-2 VL logical cores per partition
- **Maybe move some weight from LPAR02 to LPAR03**

From Perfkit FCX306 LSHARACT

LPAR Data, Collected in Partition LPAR03

Core counts:	CP	ZAAP	IFL	ICF	ZIIP
Dedicated	0	0	0	0	0
Shared physical	0	0	29	0	0
Shared logical	0	0	42	0	0

Core Type	Partition Name	Core Count	Load Max	LPAR Weight	Entlment	Cap	AbsCap	GrpCapNm	GrpCap	<CoreTotal,%> Busy	Excess	Core Conf
IFL	LPAR01	2	200	9	26.1	No	...	...	...	1.2	.0	o
IFL	LPAR02	29	2900	850	2467.5	No	...	...	...	1507.6	.0	o
IFL	LPAR03	11	1100	140	406.4	No	...	...	...	717.7	311.3	o

LPAR02: high E, low util
LPAR03: low E, high util
Maybe move some weight?

Techniques

Technique: Use Obvious, Intuitive Weights

- Some people just assign weights based on "feel"
  - They do what "feels right"
- Some people try for sum=1000
  - This makes the weights portray percentages
  - They think this helps
- Both techniques hurt us because we can't immediately see entitlement problems

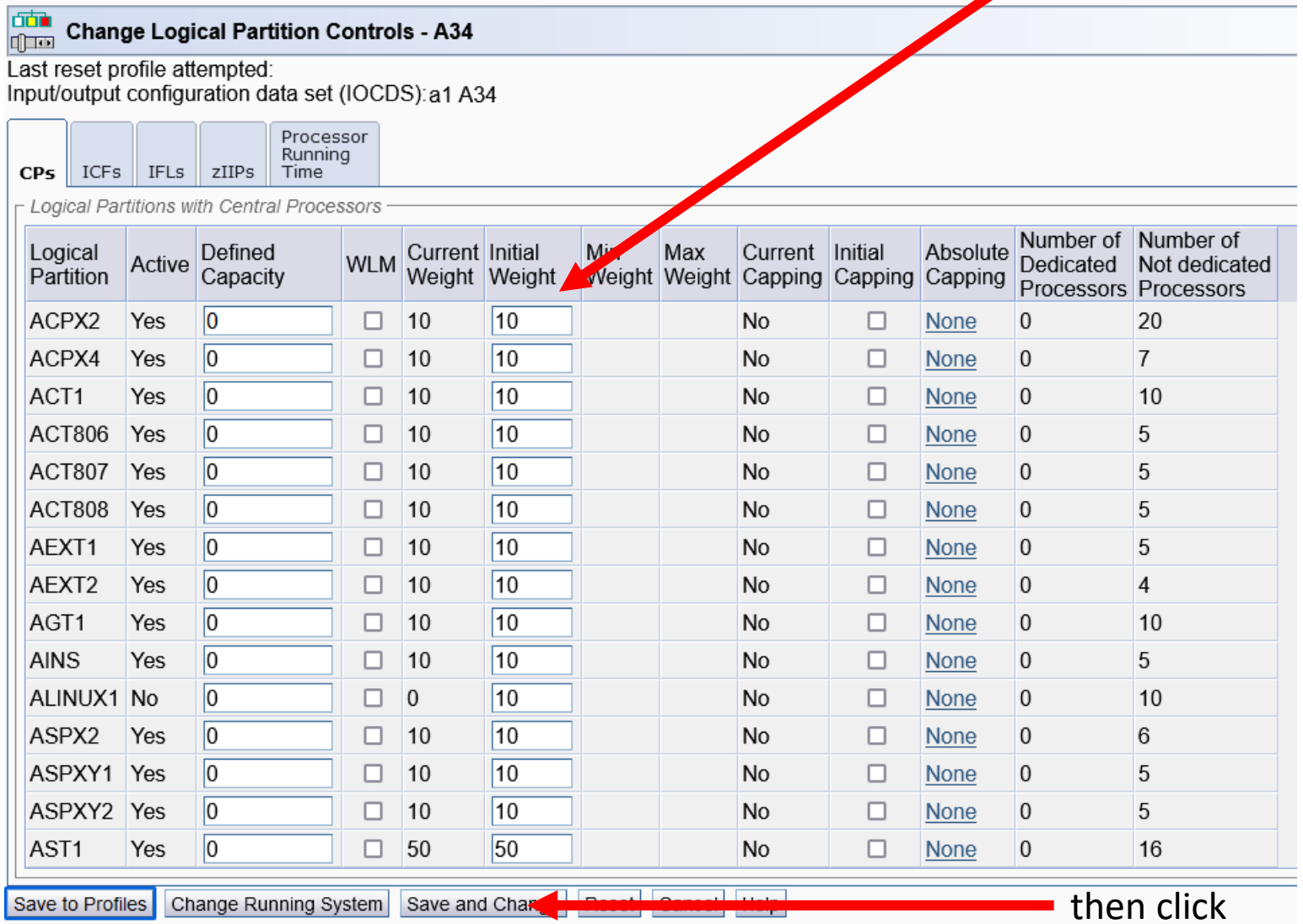
- **Make sum of weights = 10 * (# of shared physical cores)**
- **This makes each entitlement = weight/10**

Shared Physical Cores	45			
Partition	Logical Cores	Weight	% of Sum	Entitlement
FRED	20	161	35.7	16.1
BARNEY	20	64	14.3	6.4
WILMA	5	104	23.2	10.4
BETTY	20	121	26.8	12.1
sums -->		450	100.0	45.0

BARNEY might have a problem
WILMA definitely has a problem

Technique: Change When Needed

- HMC or SE: Change Logical Partition Controls



Technique: Use Medium Unparking

- z/VM 6.4 and 7.1 were *too aggressive* in unparking VLs
- Unnecessary use of VLs contributes to suspend time and PR/SM overhead
- On z/VM 7.1, configure for *medium unparking*
  - In system configuration file, code `SRM UNPARKING MEDIUM`
  - Via CP command, issue `CP SET SRM UNPARKING MEDIUM`
- z/VM 7.2 and later use medium unparking by default

Technique: Best Practices

- Know your workloads' needs for power
  - This requires observation and iteration
- Set entitlements to match workloads' needs
- Use obvious, intuitive weights
- Define only 1-2 VL cores per partition
- Check your work: FCX306 LSHARACT

- Run with medium unparking (if z/VM 7.1 or earlier, you must set this)
  - Keeps unnecessary VLs parked

- Measure and adjust as needed
  - Do not be afraid to change

Tools

Tool: LPAR Entitlement Calculator

- www.vm.ibm.com hosts an entitlement calculator
- Tool for you to do what-ifs
- Type in some basic data
- Click "Calculate"

Fill out the form, then click here --> Calculate

Number of **shared** physical cores: 45

Tell us about your LPARs:

Name	Cores	Weight	Polarity
fred	20	100	Vertical
barney	20	40	Vertical
wilma	8	65	Vertical
betty	20	75	Vertical
			Vertical

<-- 8 cores on purpose

The tool is here: https://www.vm.ibm.com/perf/tips/calcent.cgi

LPAR Entitlement Calculator: Output

LPAR Entitlement Calculator

Here is the result of your what-if.

To try another what-if, just use the browser's BACK button, change your form, and calculate again.

Shared physical cores: 45

Name	Cores	Weight	Polarity	Entitlement	# VHs	# VMs or HZs	Ent(VM or HZ)	# VLs	Unusable Ent
fred	20	100	V	1607.1	15	2	53.6	3	0.0
barney	20	40	V	642.9	5	2	71.4	13	0.0
wilma	8	65	V	1044.6	8	0	0.0	0	244.6
betty	20	75	V	1205.4	11	2	52.7	7	0.0
Totals ->	68	280		4500.0	39	6		23	244.6

In this case we see Wilma has too few logical cores for its entitlement.

Tool: CALCENT, A Better FCX306 LSHARACT

```
For type IFL:  29 physical cores, 0 dedicated, 29 in shared pool, 42 shared logical cores, WS 999

Type __LPAR__ LogCores _Weight_ Entlment CD WC GP MT IC AC AC-value GC GC-name_ GC-value P _HZ_ _VH_ _VM_ __EVM___ _VL_ _E-left_
IFL  LPAR01         2        9     26.1  0  0  1  1  0  0      0.0  0 ........      0.0 V    0    0    1     26.1    1       0.0
IFL  LPAR02        29      850   2467.5  0  0  1  1  0  0      0.0  0 ........      0.0 V    0   24    1     67.5    4       0.0
IFL  LPAR03        11      140    406.4  1  0  1  1  0  0      0.0  0 ........      0.0 V    0    3    2     53.2    6       0.0
```

- Uses MONWRITE file as input
- Writes a nice table
- Shows core polarities

CD	collected the data
WC	wait-completion assist
GP	Global Performance Data Control setting
MT	multithreading enabled?
IC	initial cap?
AC	absolute cap?
GC	group cap?
P	polarization, H or V
HZ, VH, etc.	numbers of logical cores of these types
EVM	entitlement of a VM or HZ
E-left	leftover (excess) entitlement

https://www.vm.ibm.com/download/packages/descript.cgi?CALCENT

Tool: D0R16TOP, Machine Topology

- Uses MONWRITE file as input
- Displays actual locations of Pcores
- Displays preferred dispatch locations of Lcores
- Requires z16 or later and CP with VM66532

```
================================================================================
Report of Core Placement 2021-12-16 08:01:00 system local time

Top-level ctr 1                          Top-level ctr 2

Container 1.1.1                          Container 2.1.1
 1.1.1 PCore 0002.CP                      2.1.1 PCore 0030.CP
 1.1.1 PCore 0003.CP                      2.1.1 PCore 0031.CP
 1.1.1 PCore 0004.CP                      2.1.1 PCore 003F.CP
 1.1.1 PCore 0005.CP                      2.1.1 PCore 0048.CP
 1.1.1 PCore 0006.CP                      2.1.1 PCore 0049.CP
 1.1.1 PCore 0007.CP                      2.1.1 LCore AGT1.0006.CP.Vl
 1.1.1 LCore ACT1.0000.CP.Vh              2.1.1 LCore AGT1.0007.CP.Vl
 1.1.1 LCore ACT1.0001.CP.Vh              2.1.1 LCore AGT1.0008.CP.Vl
 1.1.1 LCore ACT1.0002.CP.Vh              2.1.1 LCore AGT1.0009.CP.Vl
 1.1.1 LCore ACT1.0003.CP.Vh              2.1.1 LCore AST3.0000.CP.Hz
 1.1.1 LCore ACT1.0004.CP.Vm              2.1.1 LCore AST3.0001.CP.Hz
 1.1.1 LCore ACT1.0005.CP.Vm              2.1.1 LCore AST3.0002.CP.Hz
 1.1.1 LCore ACPX2.0004.CP.Vm             2.1.1 LCore AST3.0003.CP.Hz
 1.1.1 LCore ACPX2.0006.CP.Vl             2.1.1 LCore AST3.0004.CP.Hz
```

much wider

much longer

https://www.vm.ibm.com/download/packages/descript.cgi?D0R16TOP

When Using SMT-2, Remember This

"Core" Is Not The Same As "CPU"

Core

- Machine has physical **cores**
  - Two CPUs per core
- Partition has logical **cores**
  - Two CPUs per core
- In the image activation profile, you are giving the partition **logical cores**
- Entitlements are in terms of **cores' worth of power**
- PR/SM dispatches logical **cores** onto physical **cores**
- z/VM parks **cores** based on **core utilization** and **available core power**
- These utilizations are **core** utilization:
  - FCX302 PHYSLOG
  - FCX126 LPAR
  - FCX202 LPARLOG
  - FCX306 LSHARACT
  - FCX299 PUCFGLOG

CPU

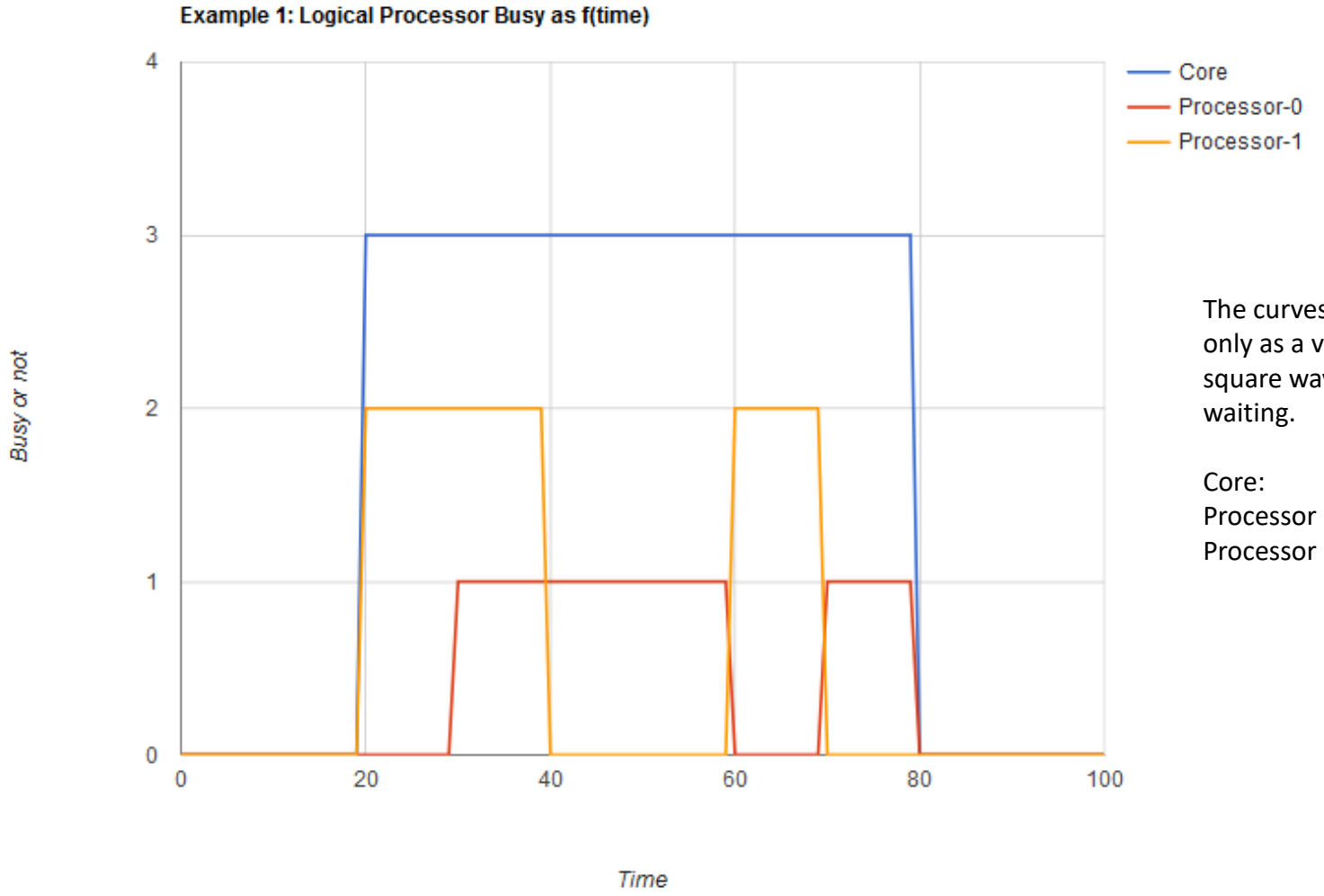
- These utilizations are **CPU** utilization:
  - FCX100 CPU
  - FCX144 PROCLOG
  - FCX225 SYSSUMLG
  - FCX304 PRCLOG
  - All the user utilization reports (FCX112 USER, FCX162 USERLOG, FCX288 USRMPLOG, etc.)

References:

- "SMT Vocabulary Tips", https://www.vm.ibm.com/perf/tips/smtvocab.html
- "CPU Utilization in an SMT World", https://www.vm.ibm.com/perf/tips/smtutil.html

Core-Busy vs. CPU-Busy

- While the core is dispatched, its CPUs go in and out of busy
- **Core utilization is not the same as CPU utilization**
- Use the correct Perfkit reports



The curves are separated by height only as a visual aid. Look only at the square wave shapes: running vs. waiting.

Core:	60 busy intervals
Processor 0:	40 busy intervals
Processor 1:	30 busy intervals

Summary

Summary

- Know what LPAR weight is and where to change it
- Know what entitlement is
- Know how entitlement relates to logical core count
- Know the common pitfalls
- Do not be afraid to make changes

References

- "Topics in LPAR Performance", https://www.vm.ibm.com/library/presentations/lparperf.pdf
- "Brian's z/VM Performance Best Practices", https://www.vm.ibm.com/perf/tips/bestp.html
- "Controlling Vertical-Low Logical Cores", https://www.vm.ibm.com/perf/tips/unpark.html
- "LPAR Entitlement Calculator", https://www.vm.ibm.com/perf/tips/calcent.cgi
- "Understanding z/VM HiperDispatch", https://www.vm.ibm.com/perf/tips/zvmhd.html
- "Understanding z/VM CPU Utilization", https://www.vm.ibm.com/perf/tips/lparinfo.html
- "SMT Vocabulary Tips", https://www.vm.ibm.com/perf/tips/smtvocab.html
- "CPU Utilization in an SMT World", https://www.vm.ibm.com/perf/tips/smtutil.html