HiperDispatch – User Experiences SLA Improvements and MSU Reductions

Session LF
Tue Nov 5th, 2019:
4:45 PM - 5:45 PM
Room Woodcote

Donald Zeunert

Agenda

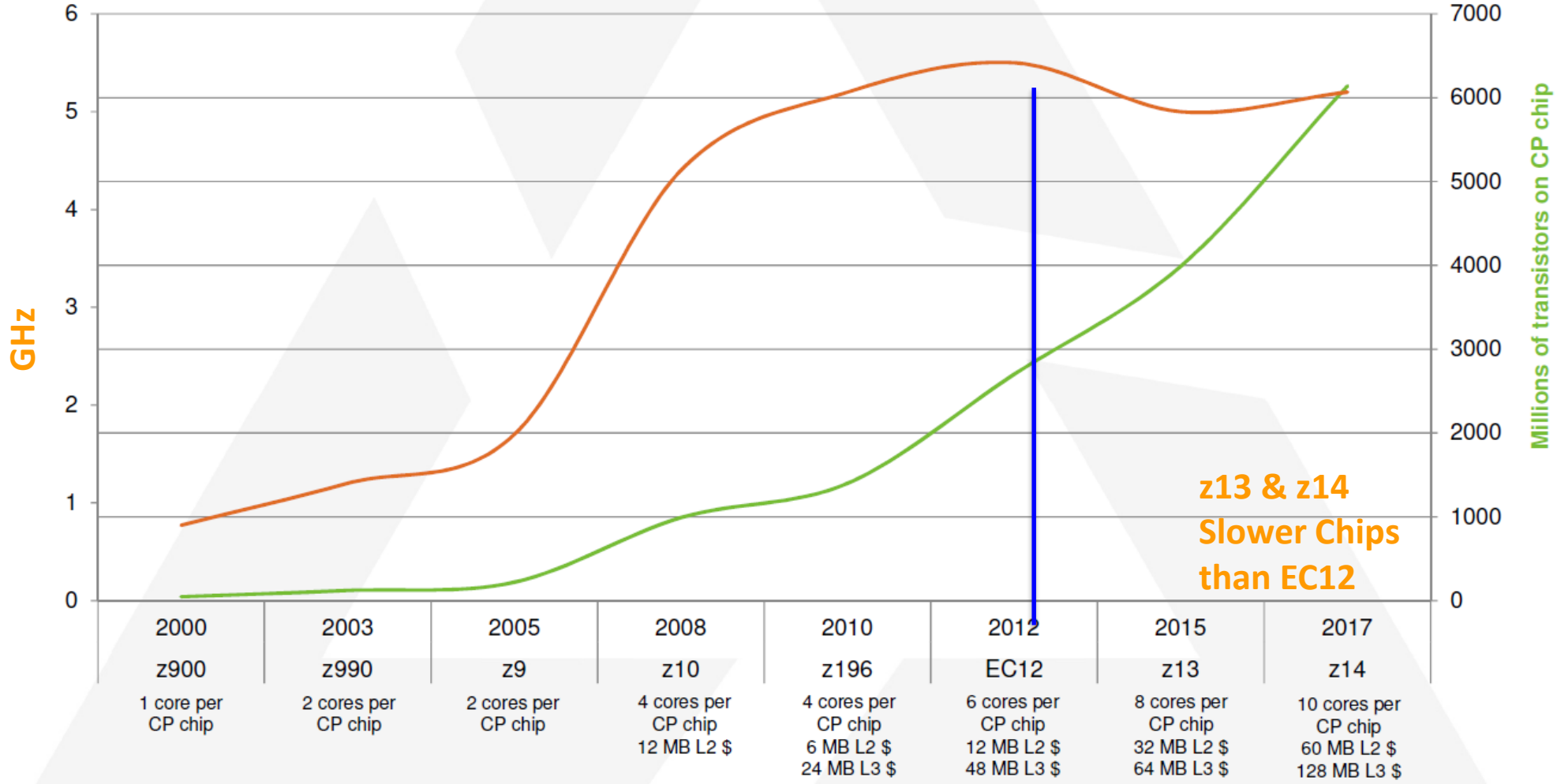
Why should I care about Chip Cache?

What issues was HiperDispatch designed to address?

How can I configure HiperDispatch for Optimal Performance?

Customer Results from Tuning

GCP Speeds – No longer increasing



Higher MSUs / CP – is from Chip Cache

Extra cache layers / more cache memory

Unified vs separate instruction and data caches

Smaller DAT - for 1mb / 2GB page, more fit more on chip

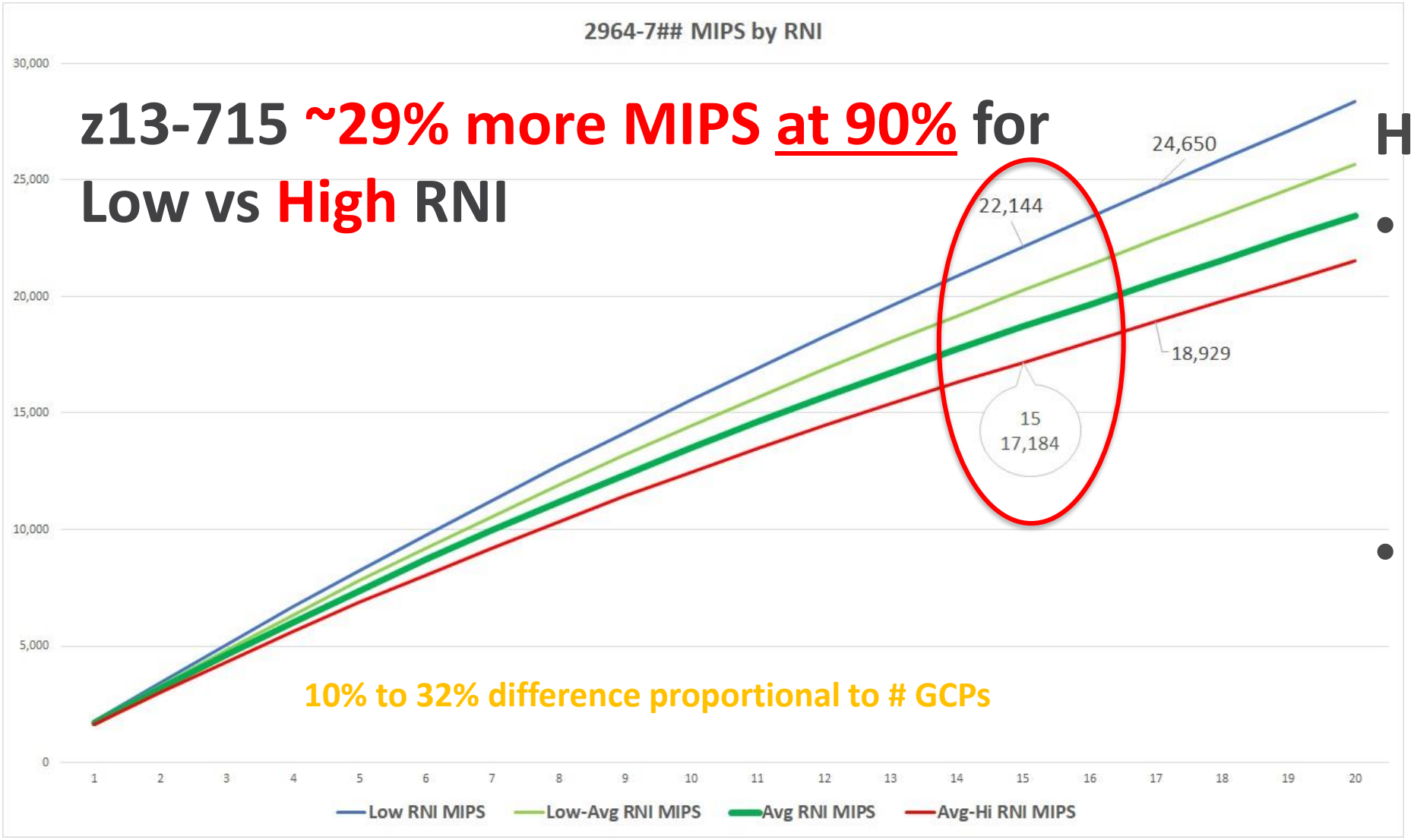
Smarter prefetching logic

Smarter branch prediction logic

Enhanced out-of-order issue/execution

New more efficient instructions (needs new compilers)

MIPS Rating – Cache Hit Impact



z13-715 ~29% more MIPS at 90% for Low vs High RNI

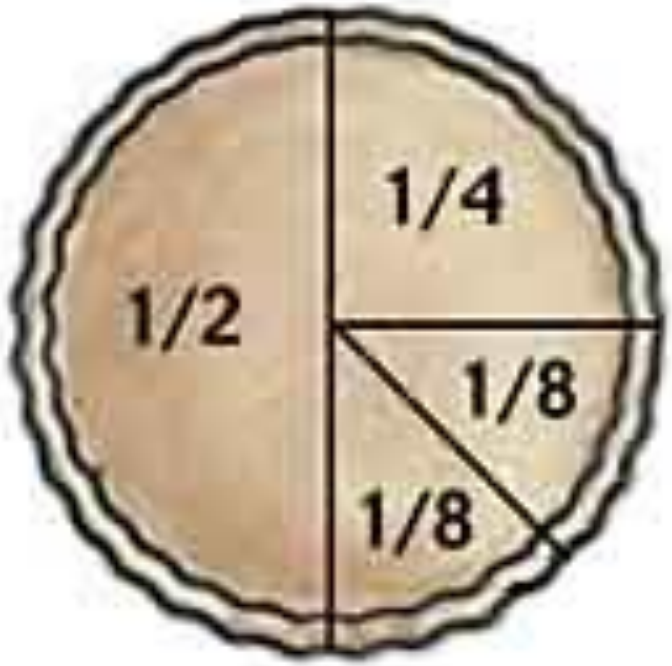
10% to 32% difference proportional to # GCPs

Higher RNI =

- **Higher CPU secs / MSUs for same workload**
- **Fewer MSUs delivered per GCP**

PR/SM Objective – Use 100%

CEC at 100%



LPAR Fair Share

If Capping is not controlling all LPARs, capping one LPAR doesn't mean it goes to other one.

HiperDispatch – Improve Cache%

Vertical vs Horizonal Polarity

- **Vertical High – Physical CP virtually dedicated to LP**
- **Vertical Medium – split of entitlement < 2 across 2 CPs**
- **Vertical Low – LPs defined not needed to satisfy entitlement**
    - Parked Engines – VLs are parked when not needed

Overcommit Logical to Physical

Without HiperDispatch or IRD (Obs)
PR/SM Share ~2 GCP,
Defined 4 LPs
Each get 50% of GCP
CP Constraints

With HiperDispatch
2 VMs, 2 VLs
Each get ~ 1 GCP
Unleash Latent Demand?

With HiperDispatch Tuned
1 VH and 1VM, 2 VLs

Calculating LPAR MSUs from Weights

LPAR SHARE = LPARs Weight / Sum of Weights

Processor guarantee =
# of Physical GCPs(4) * LPAR Share

LPAR Name	# LPs	Wgt	Rel Shr	Guar CPs
LPAR1	3	600	60.00%	2.40
LPAR2	2	300	30.00%	1.20
LPAR3	1	100	10.00%	0.40
Total		**1000**		

Controls Vertical LP type assignment

Max MSUs = MSUs per CP * #LPs

Minimum entitled MSUs = MSUs per CP * Guaranteed CP share

Example CEC 2965-V04 has 4 GCPs at 101 MSUs each, PR/SM has 6 LPs defined
LPAR 2 Min is 121 MSUs (101 MSUs * 1.20) and
Max is 202 MSUs (101*2) which is 67% more than guaranteed

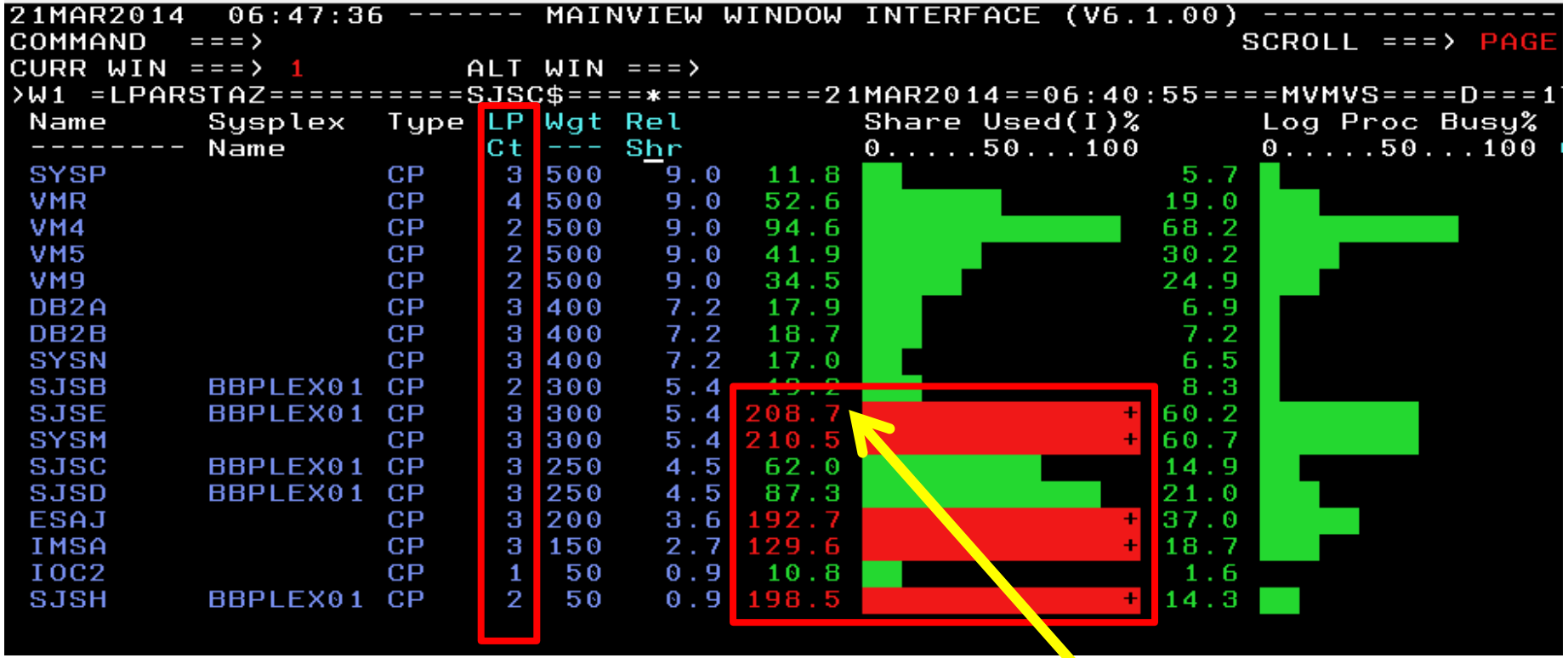
Optimize LPAR Performance

Ensure PR/SM Entitlement Adequate for Workload

- **If necessary alter weights for Shifts or Business Cycles**
- **Avoid Executing significant MSUs on VLs**
    - Check over entitlement w/ Monitors
    - Verify VLs are parked most of the time
- **Optimize LPARs PR/SM Entitlement for VH Assignment**

Logical / Physical – Entitlement

PR/SM Defs - Do not create a maximum other than 100% of # of LPs

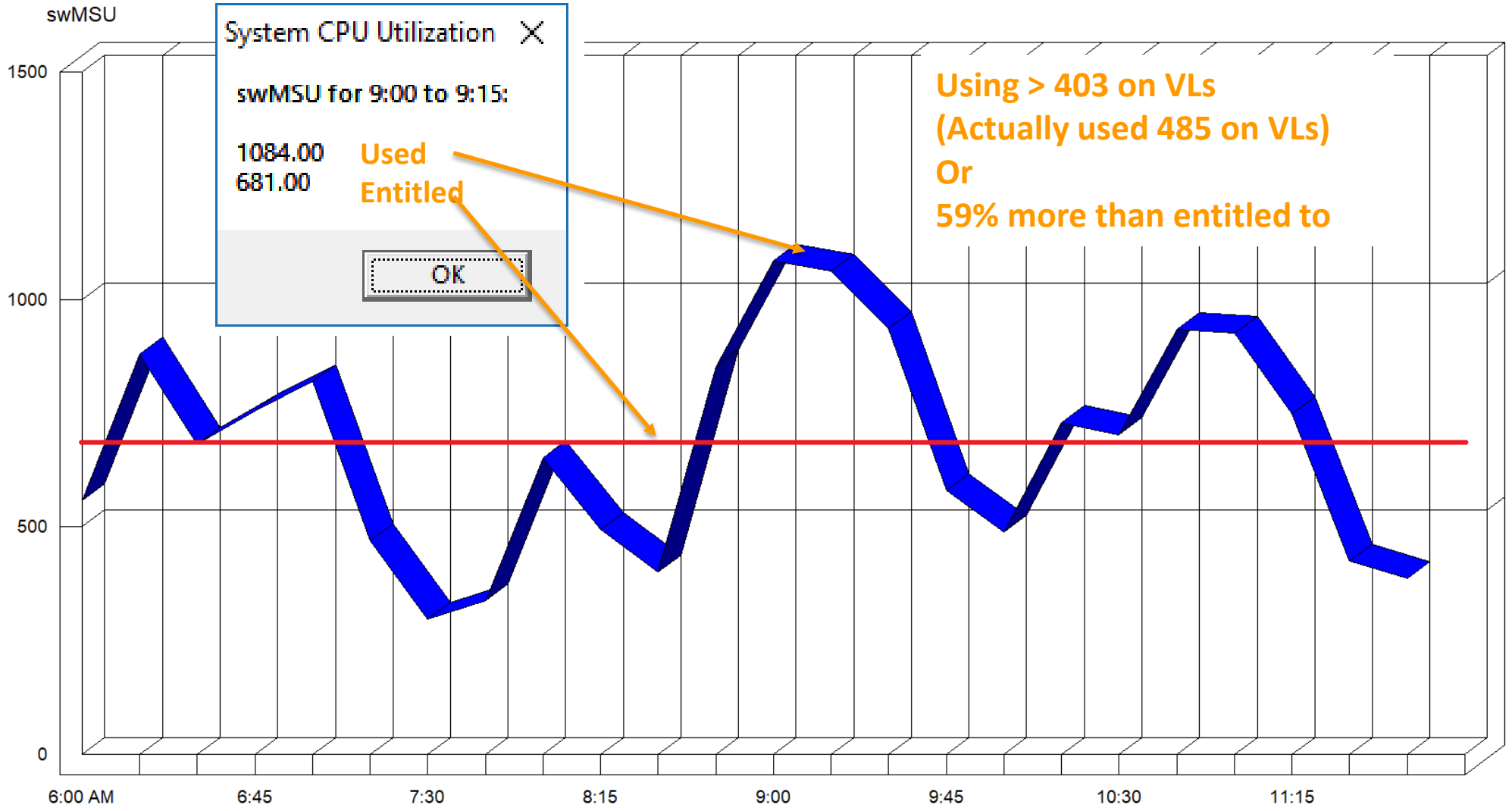


Everyone's share < 10%
of 16 CPs so Entitlement
< 1.6 CPs

Executing > Entitlement = using VLs

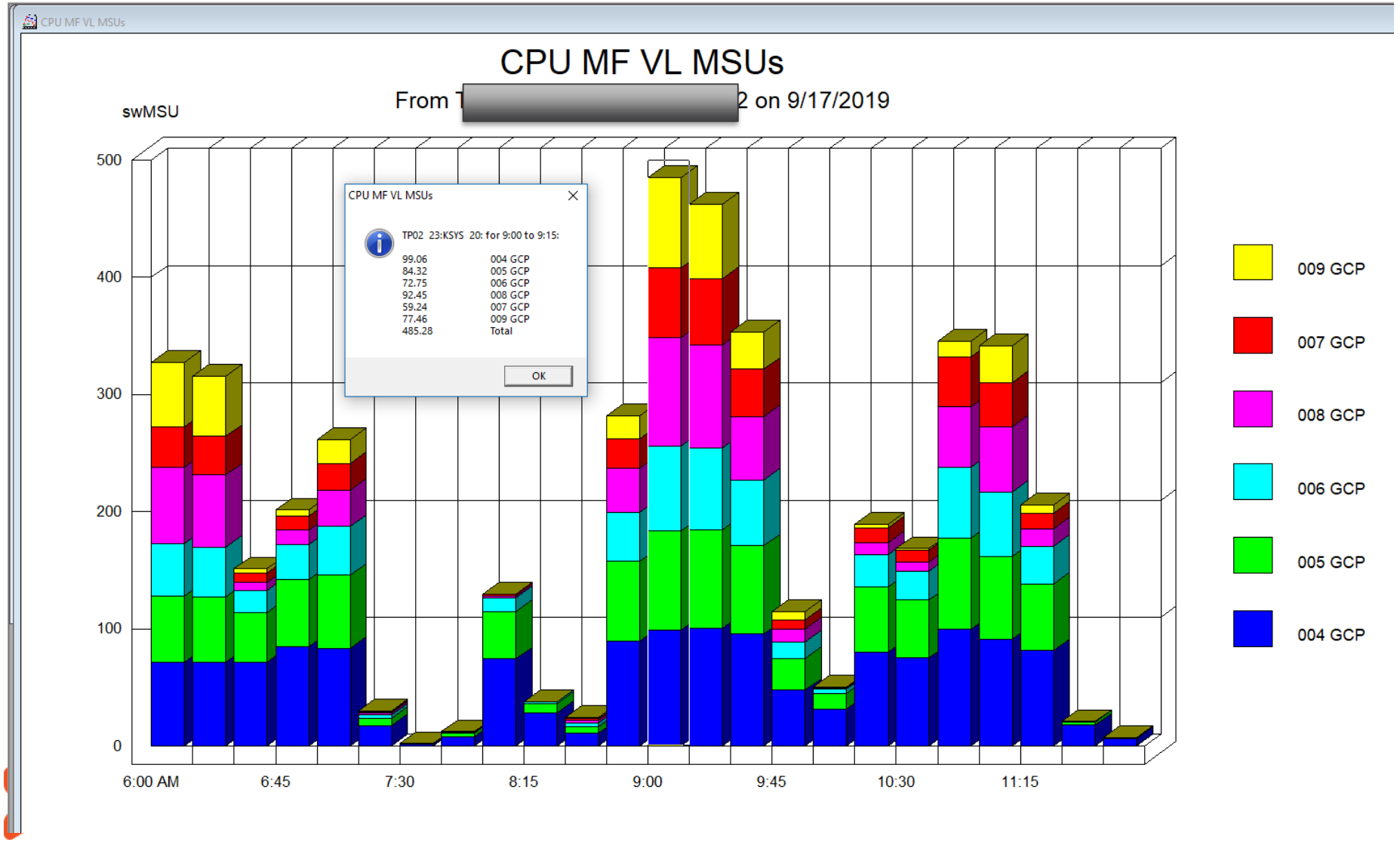
Using ~6% of CPC > Entitlement

Capacity Used vs Entitled



Recommend
PR/SM Changes
to Entitlement

6% Wasted MSUs on VLs



485 VL MSU Inefficient

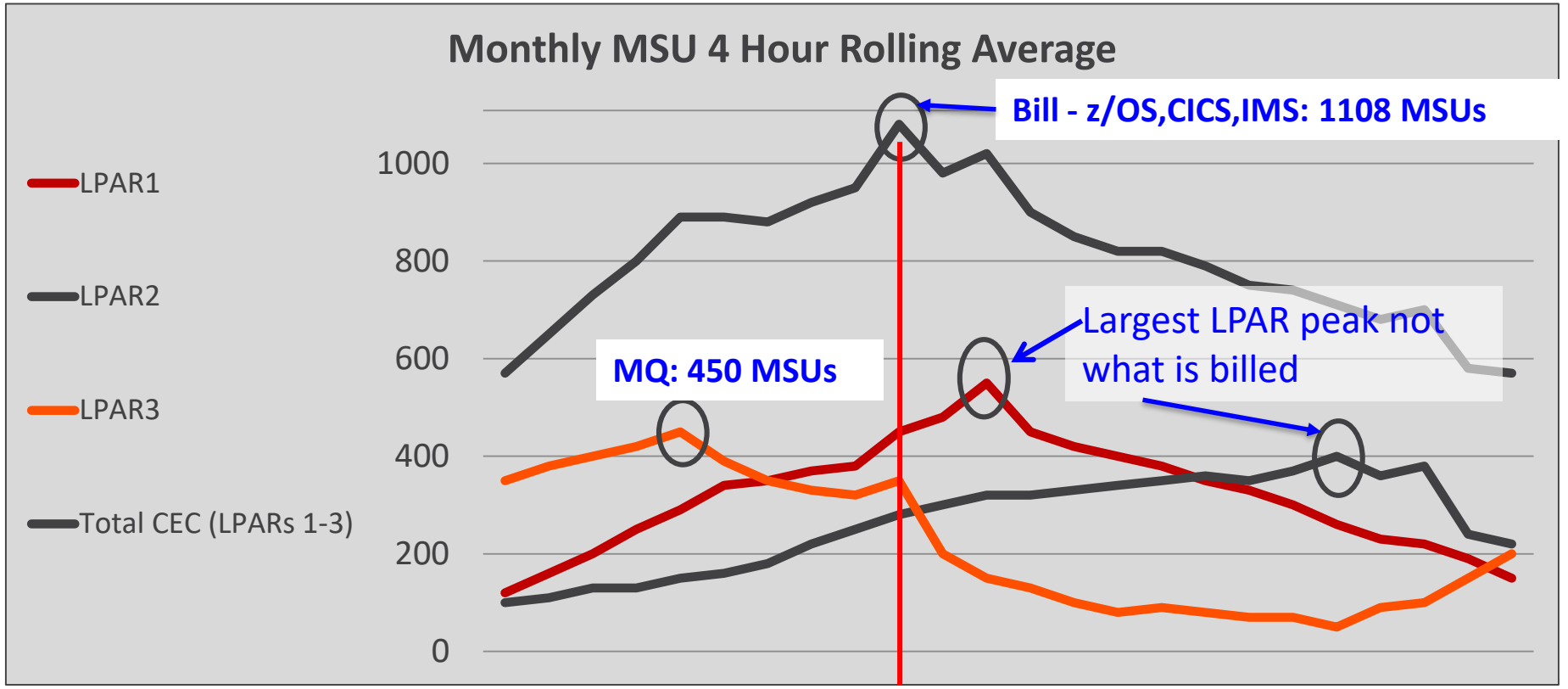
- 100 waste of 1700 Used
- Means 5.88% more MSUs

26% Less Efficient Calcs

385 * 1.26=485

485 – 385 = 100 MSUs

Why Should I Tune PR/SM?

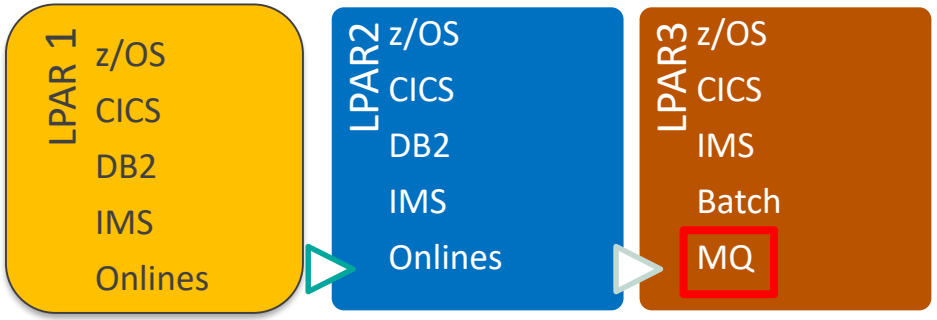


LPAR1 Peak 4HRA
550 MSUs

LPAR3 Peak 4HRA
450 MSUs

LPAR2 Peak 4HRA
400 MSUs

CPC Peak 4HRA
1108 MSUs



Reduces MSUs
Postpone Upgrades
Improves SLAs
Reduces MLC

- **Care about VLs when Impacts 4HRA peak**

Vertical Assignment Rules

Entitled CPs

- <=1 = 1 VM
- z13 - > 1 to 2.0 = 2 VM equal share
- z14 - > 1.5 to 2.0 = 1 VH and 1 VM w/ remaining share
- > 2 = whole #s are VH, remainder is VM

LPs > Entitled CPs

- All are VLs, large #s create unnecessary overhead

Typical Issues

PR/SM Settings not touched in years

LPARs MSU consumption peak utilization different times of day

GDPS LPARs Weight inconsistent w/ normal consumption

McLane – HiperDispatch Tuning Results

McLane Co. Inc.

z13s (2965-X05)

3 Production LPARs

- 1 Batch Only (DB2)
- 1 Batch DB2 Server
- 1 Online CICS only

McLane Company, Inc. is one of the largest supply chain services leaders, providing grocery and foodservice supply chain solutions for convenience stores, mass merchants, drug stores and chain restaurants throughout the United States.

McLane, through McLane Grocery and McLane Foodservice, operates over 80 distribution centers across the U.S. and one of the nation's largest private fleets. The company buys, sells and delivers more than 50,000 different consumer products to nearly 110,000 locations across the U.S.

In addition, McLane provides alcoholic beverage distribution through its wholly owned subsidiary, Empire Distributors, Inc.

McLane - Environment

Before

CICS LPAR –

- High weight 24hrs a day
- Only needed 8am-6pm

Batch LPAR –

- Used > Entitlement as plenty of spare CICS LPAR MSUs

After

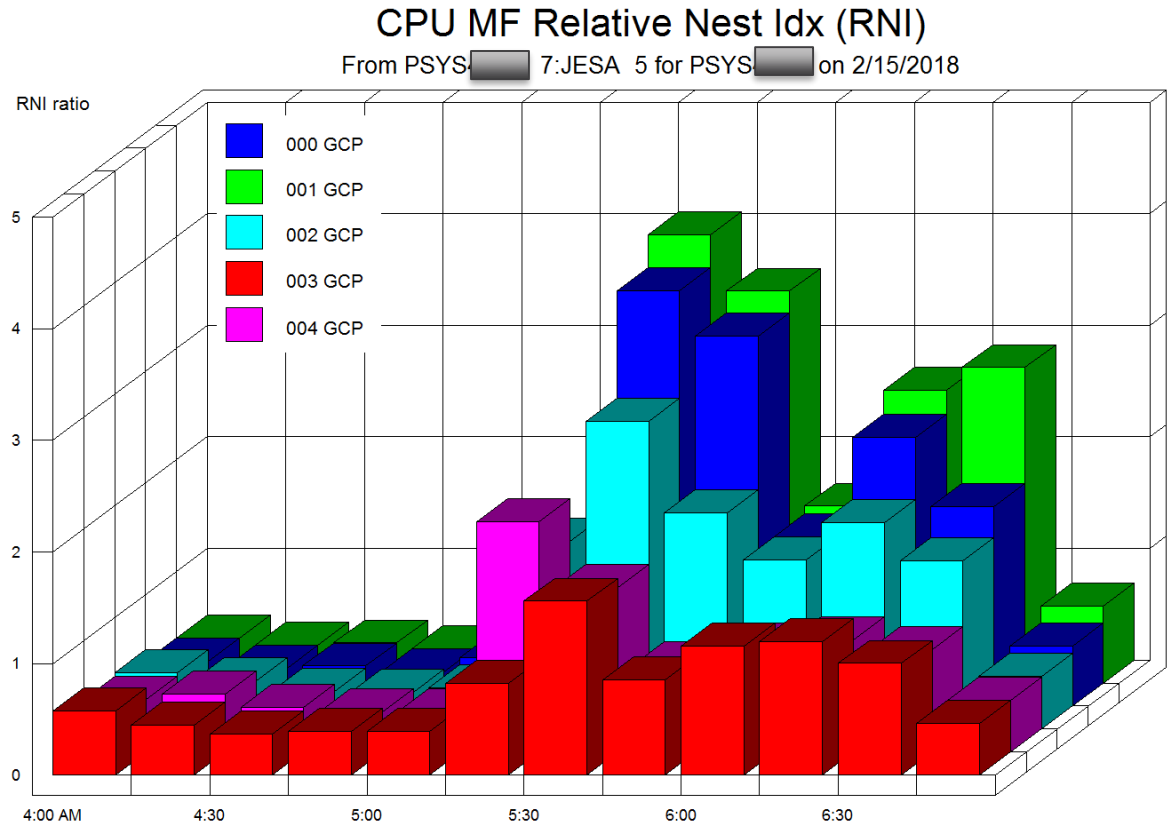
CICS LPAR –

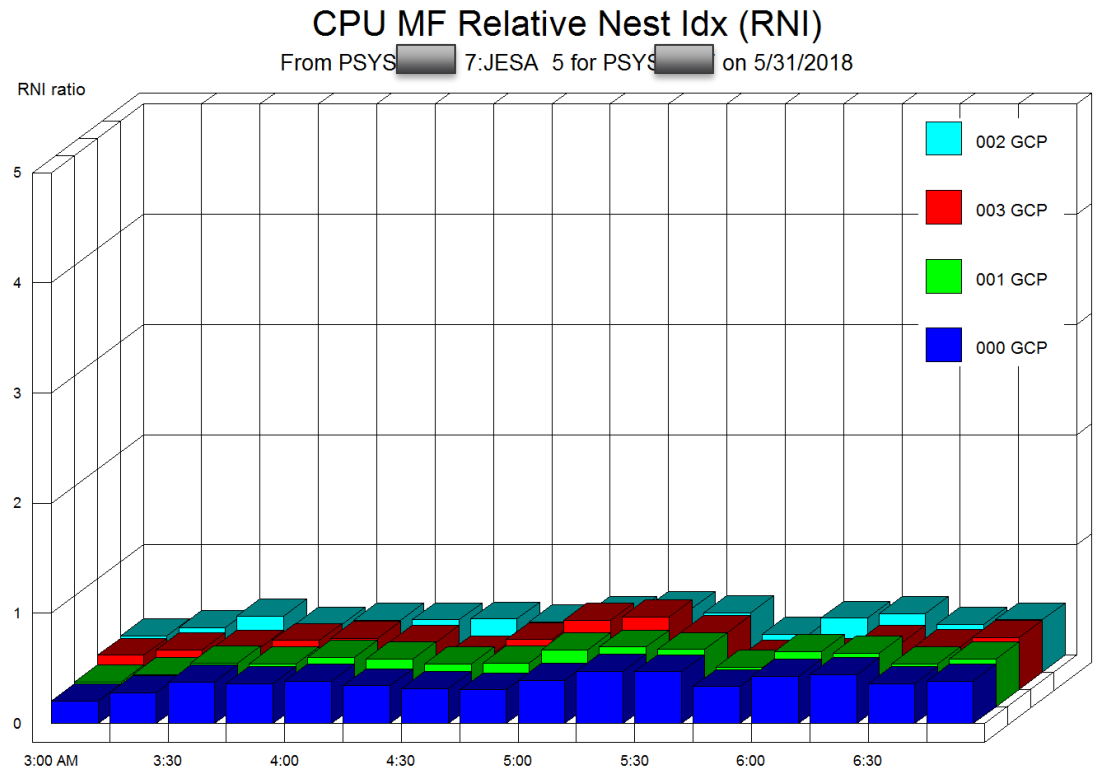
- Low Weight in batch Window

Batch LPAR –

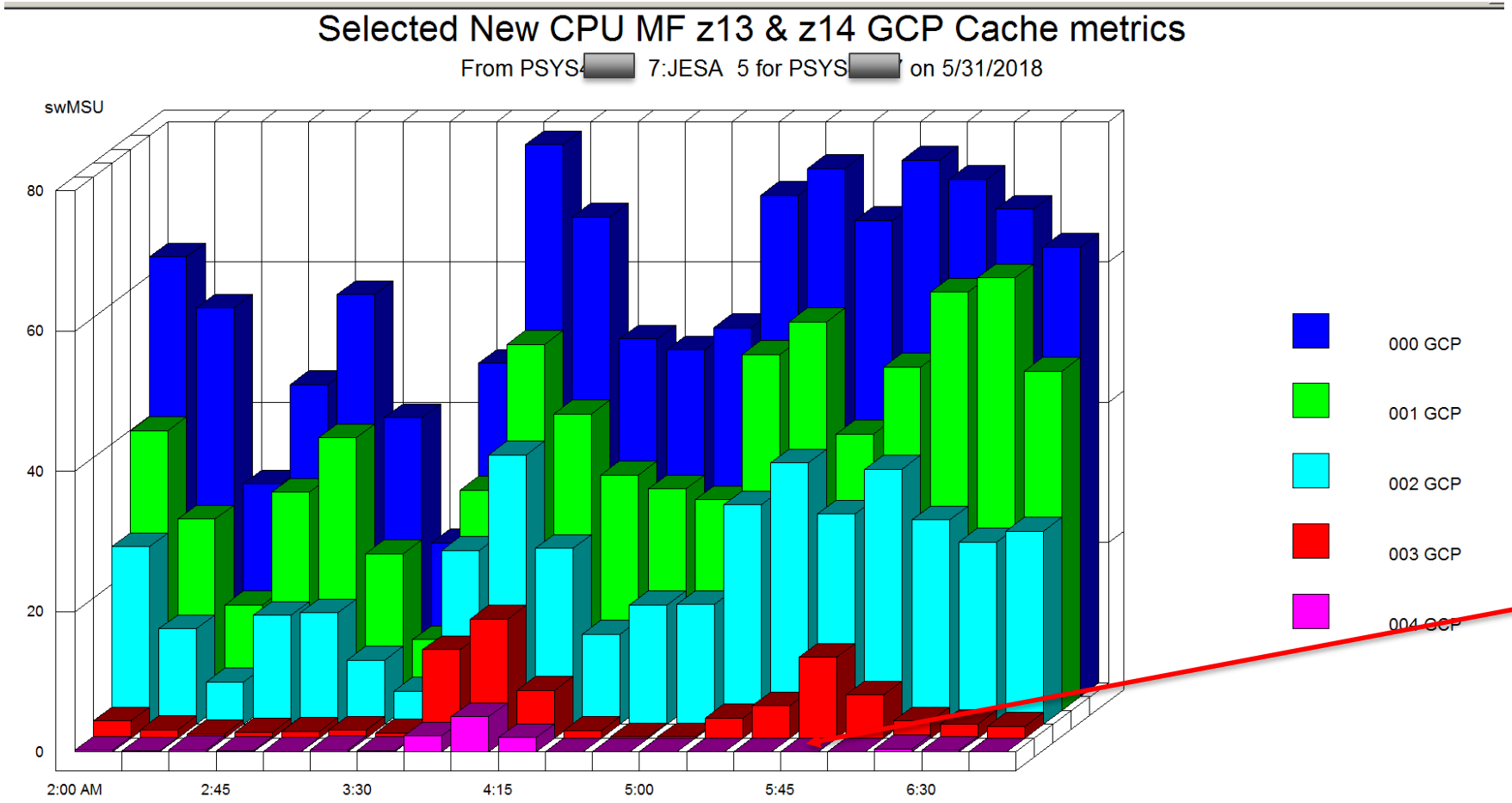
- Adequate weight to consume cycles w/o using VLs

RNI – Before vs After





VL GCP 004 – After Excluded as very lightly used



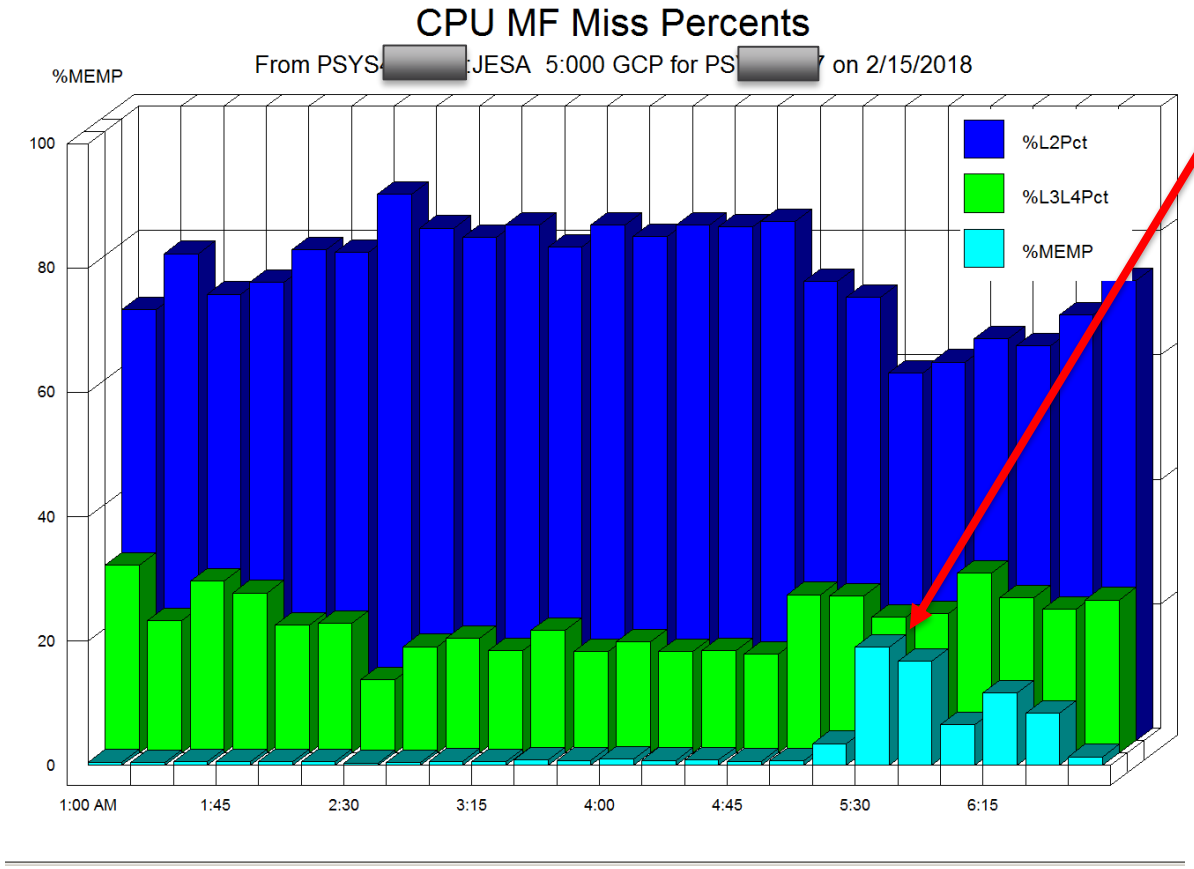
High CPI expected on very low %LP consumption

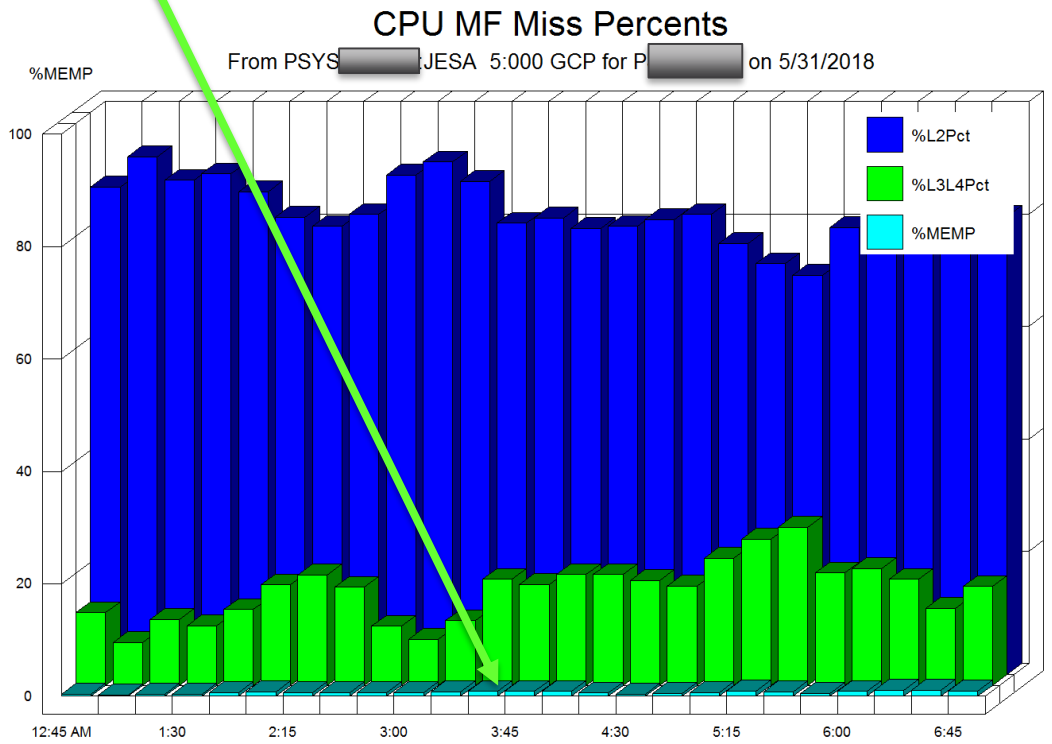
5:15 AM low from zero

RNI / Nest After – Why Improved

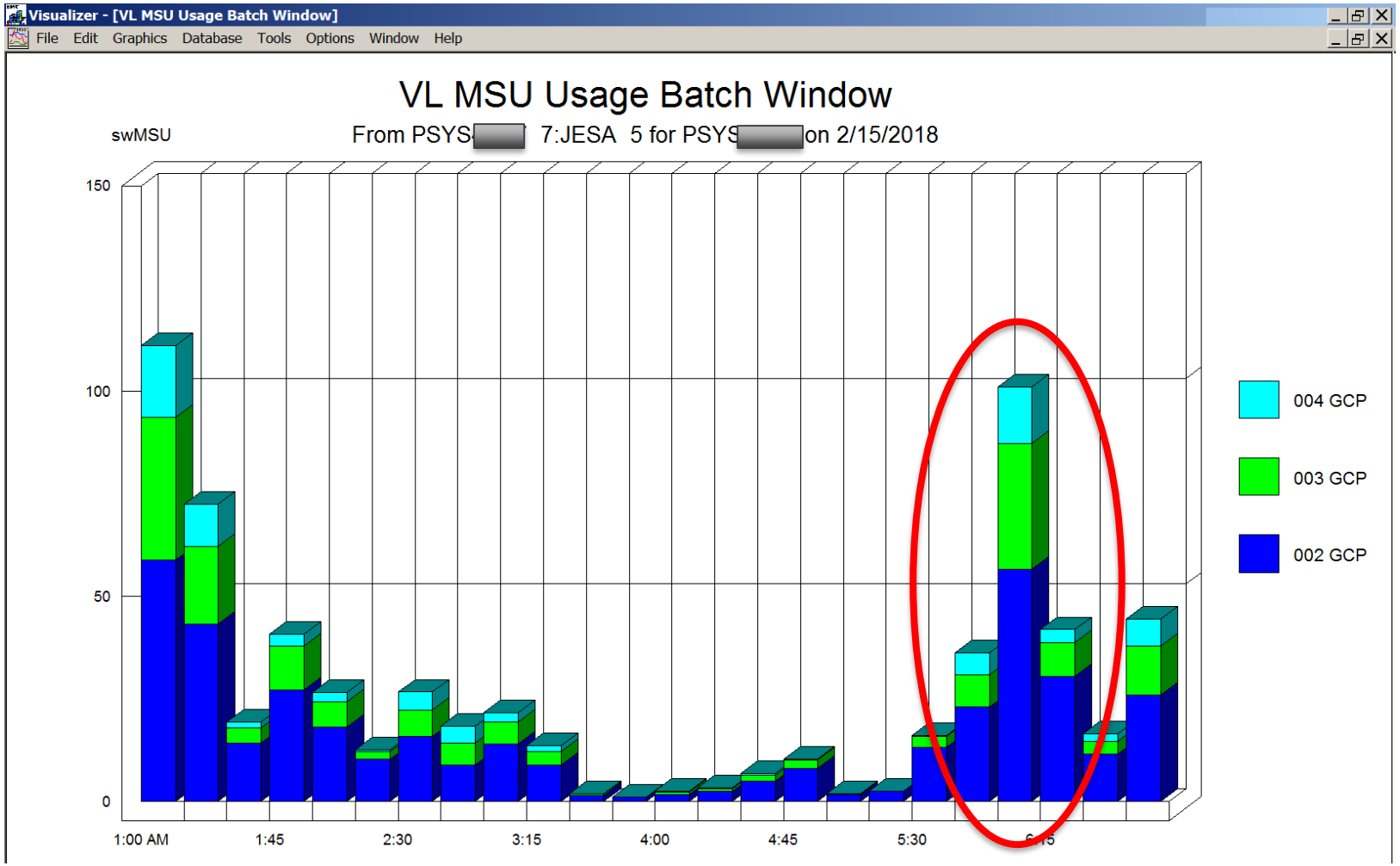
Before - Large sourced Memory

After – < 1%





Before – Batch VL High Usage



Batch Peak

- SLA completion risk

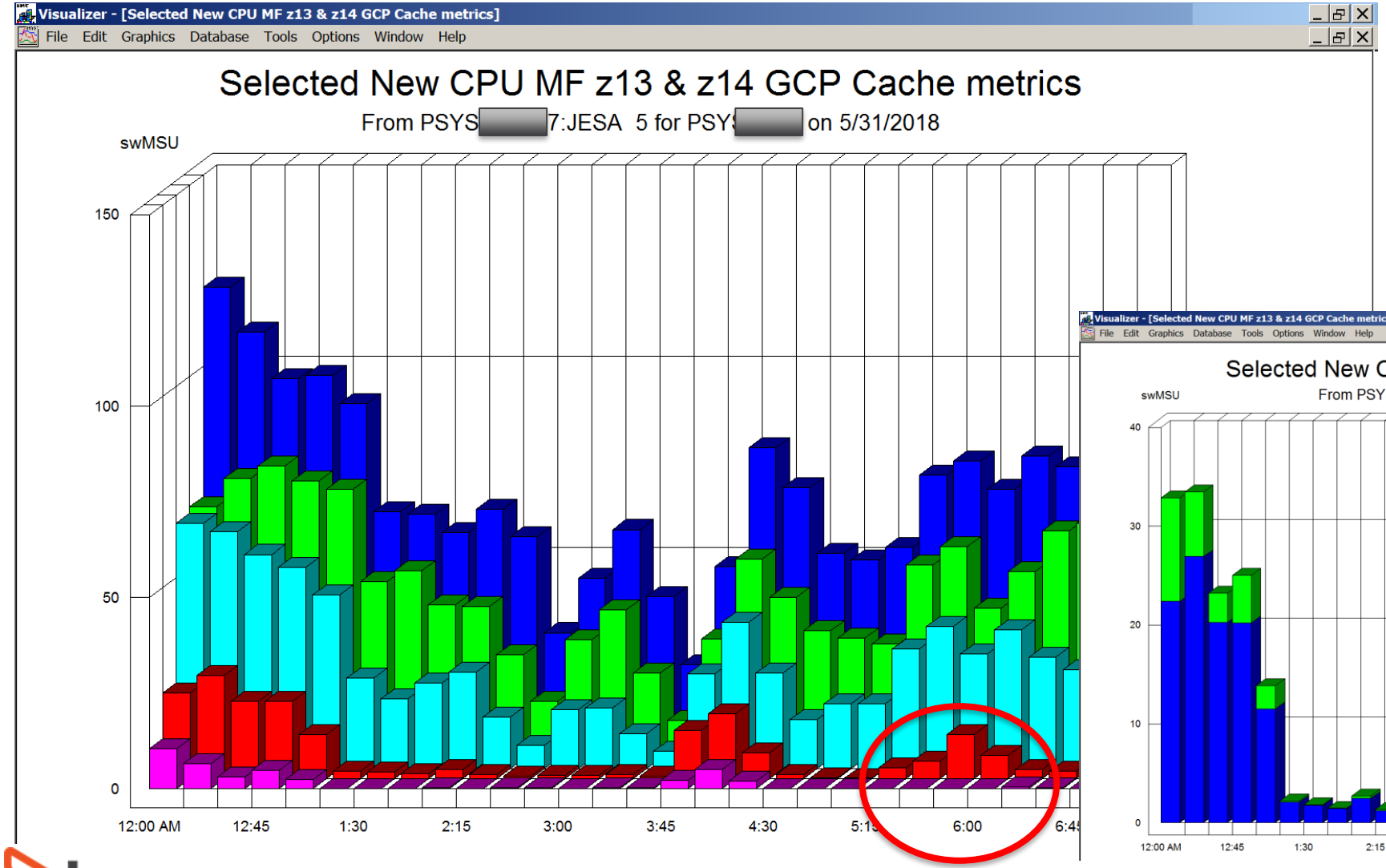
2 VMs - 0-> 1

- Use 80 – 100

3 VLs 2-> 4

- Use > 40 - 100

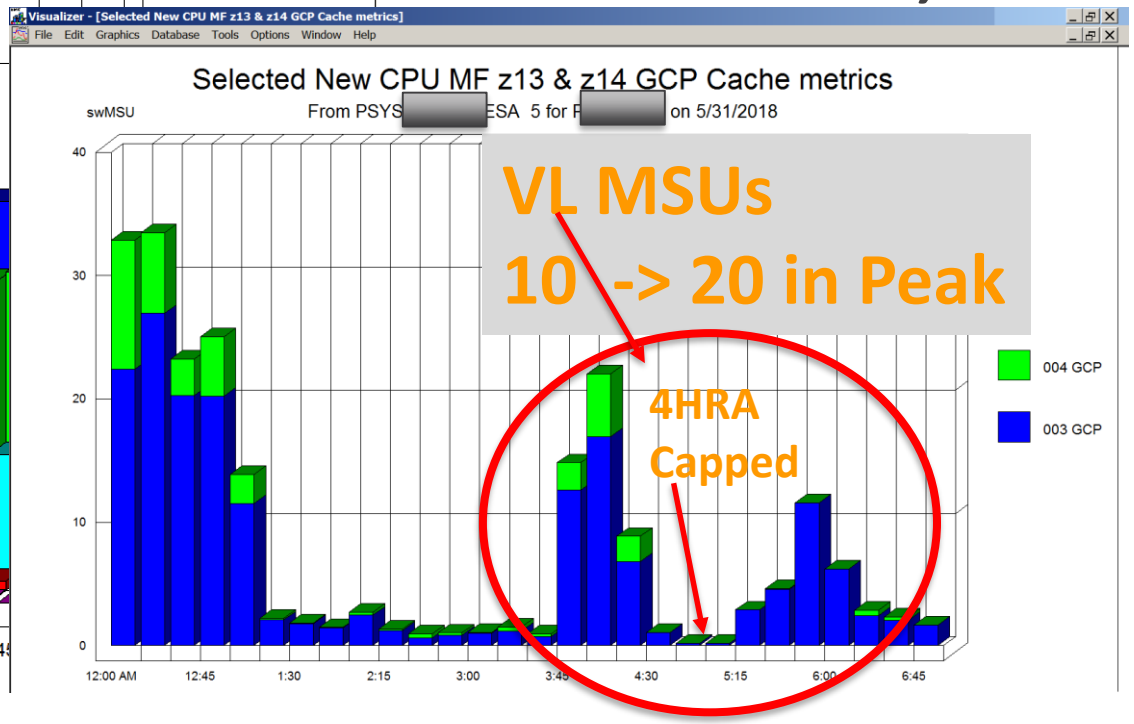
After – Batch VH vs VL Usage



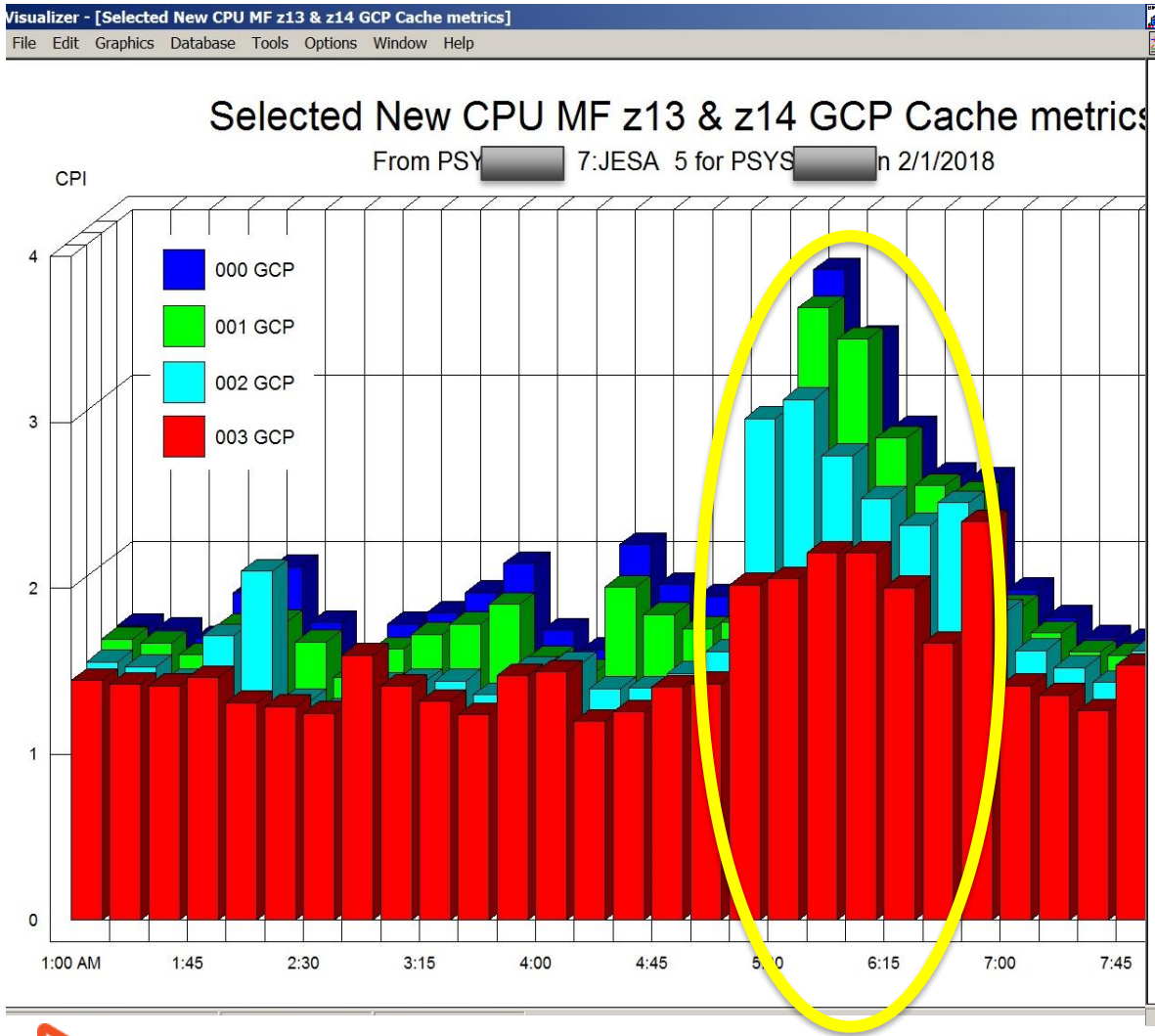
1 VH – Peak 110,

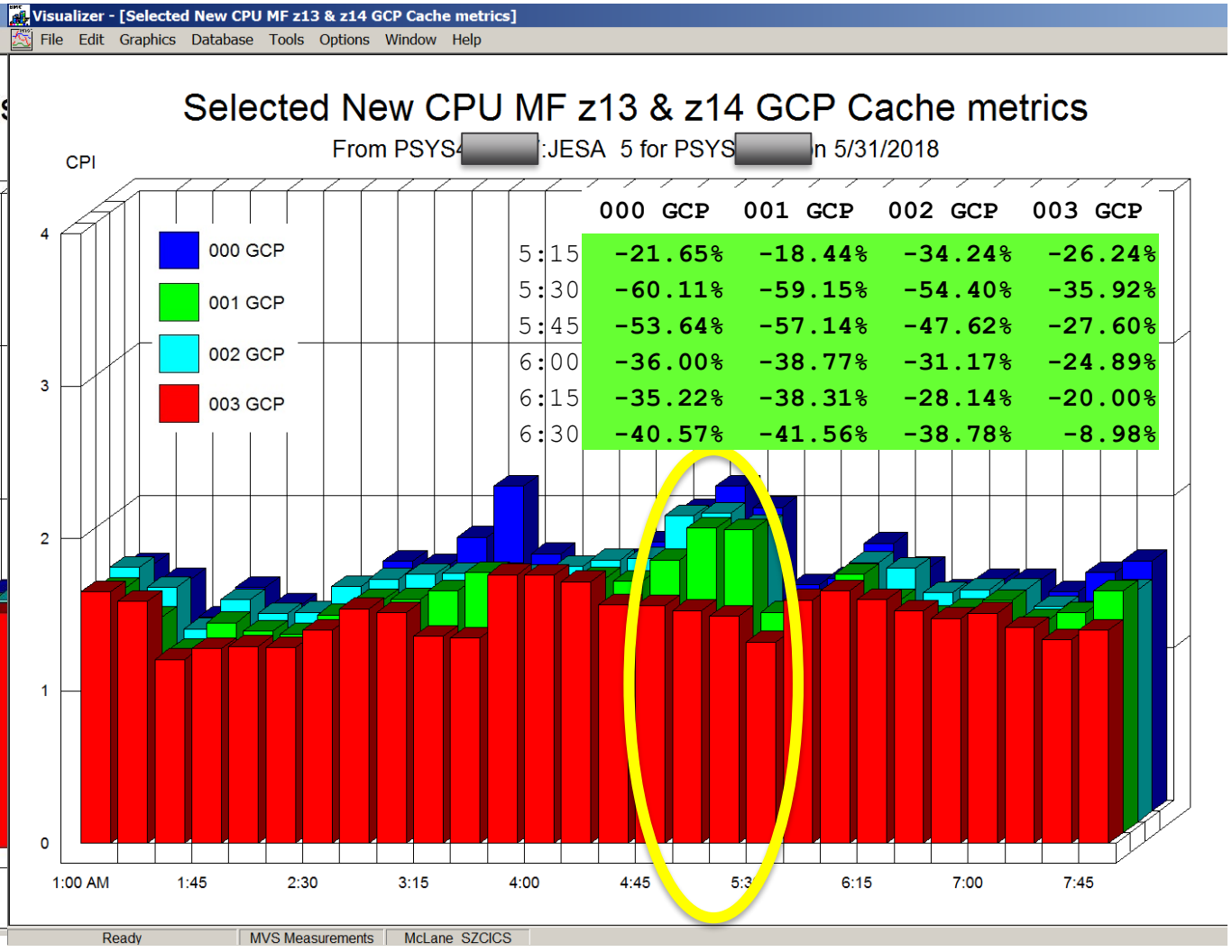
2VM – Peak 72,52

2 VL – Peak 42,32

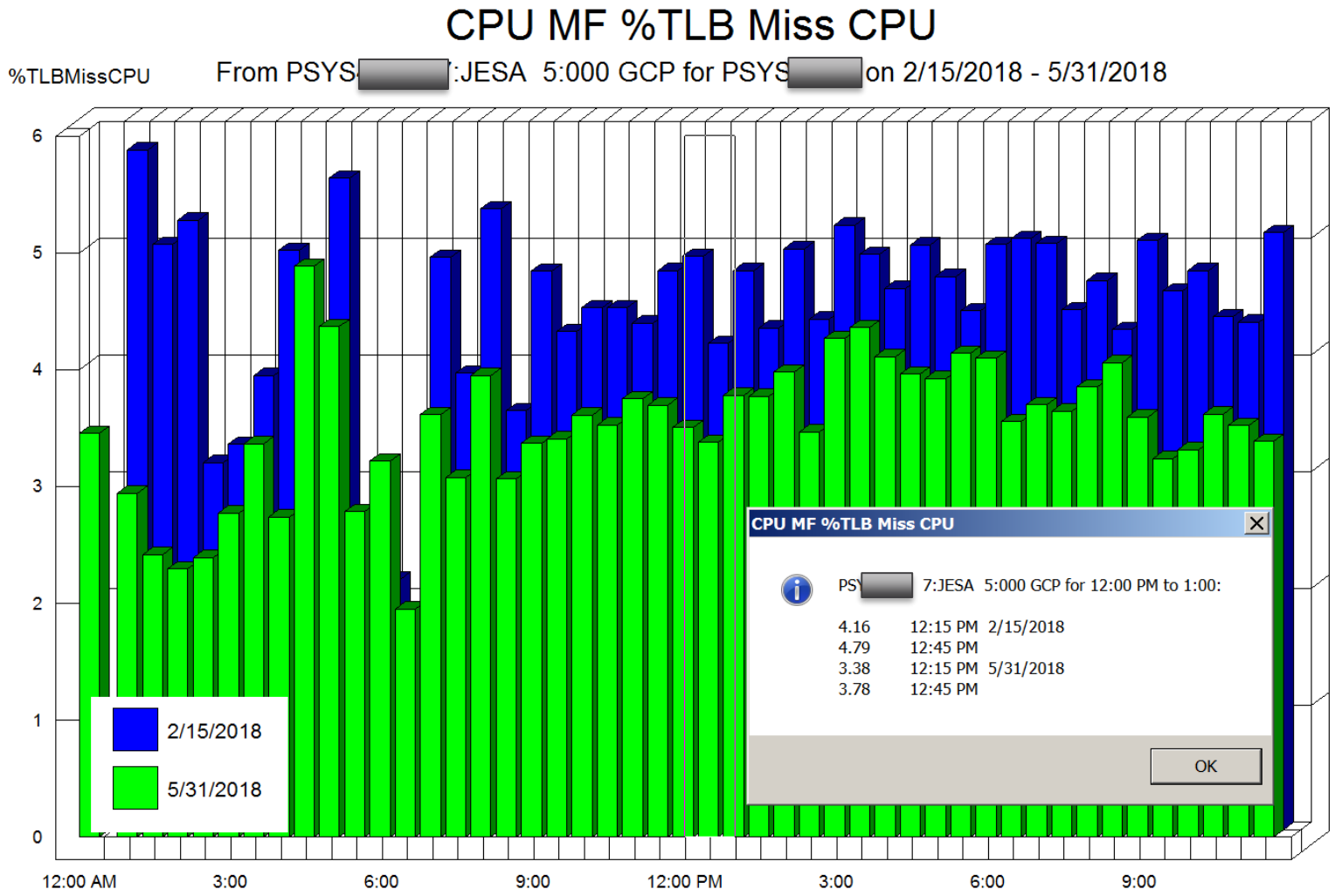


CPI Spike Significantly reduced





%TLB Miss CPU Reduced w/ VH



Before 4.79

After 3.78

Save 1%

Customer X – HiperDispatch Tuning Results

Customer X - Environment

Before

4 GDPS LPAR –

- High weight 24hrs a day
- Only needed < 1% of time

2 LARGE Prod LPAR –

- Used > Entitlement as plenty of GDPS used < 1% of entitlement

After

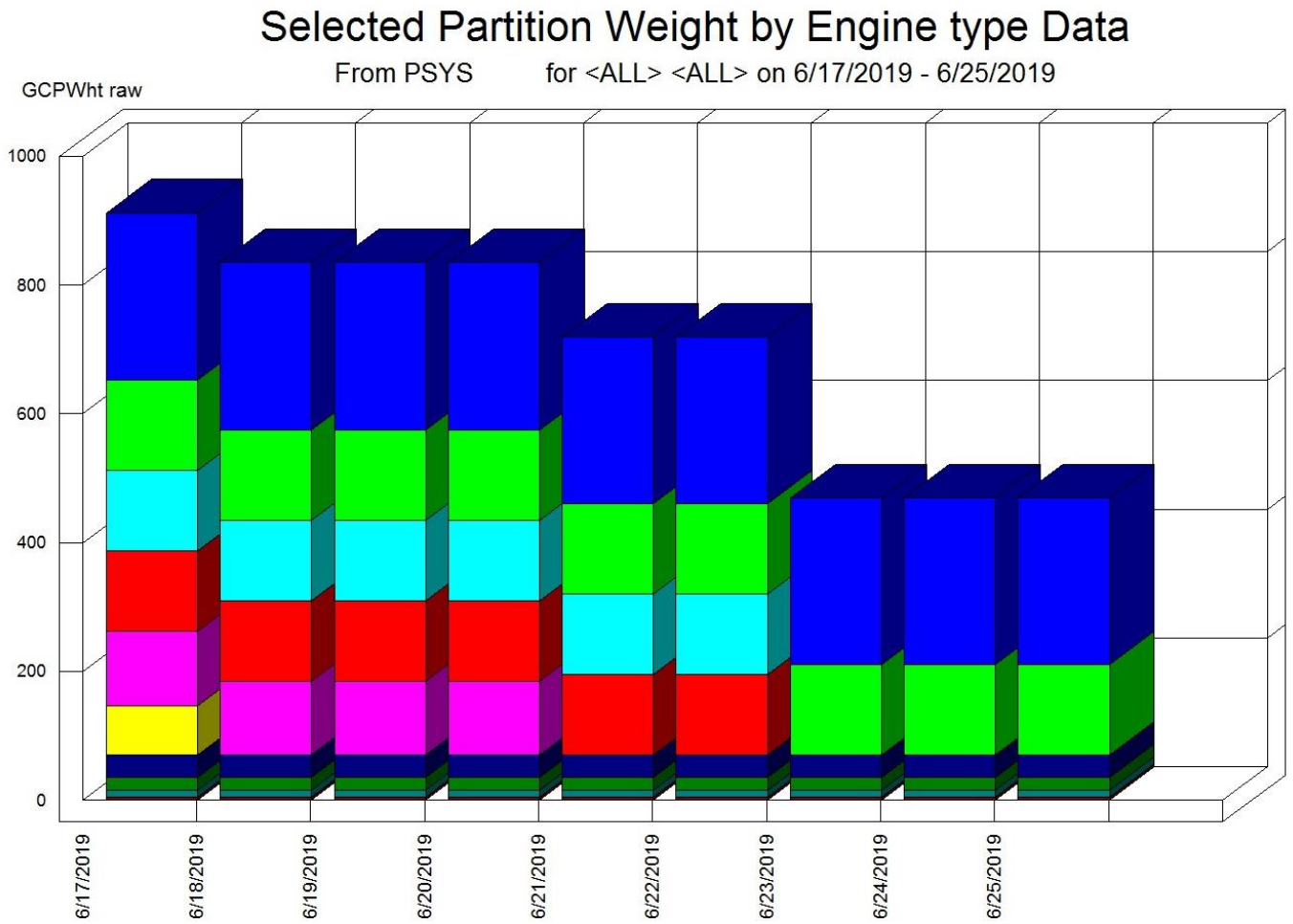
Fewer GDPS LPARs –

- Lower Weight on remaining

Prod LPARs –

- Adequate weight to consume cycles w/o using VLs

GDPS Eliminate



PRD1
QA01
GDPS1
GDPS2
GDPS3
GDPS4
TEST
QA02
SAND
SAND

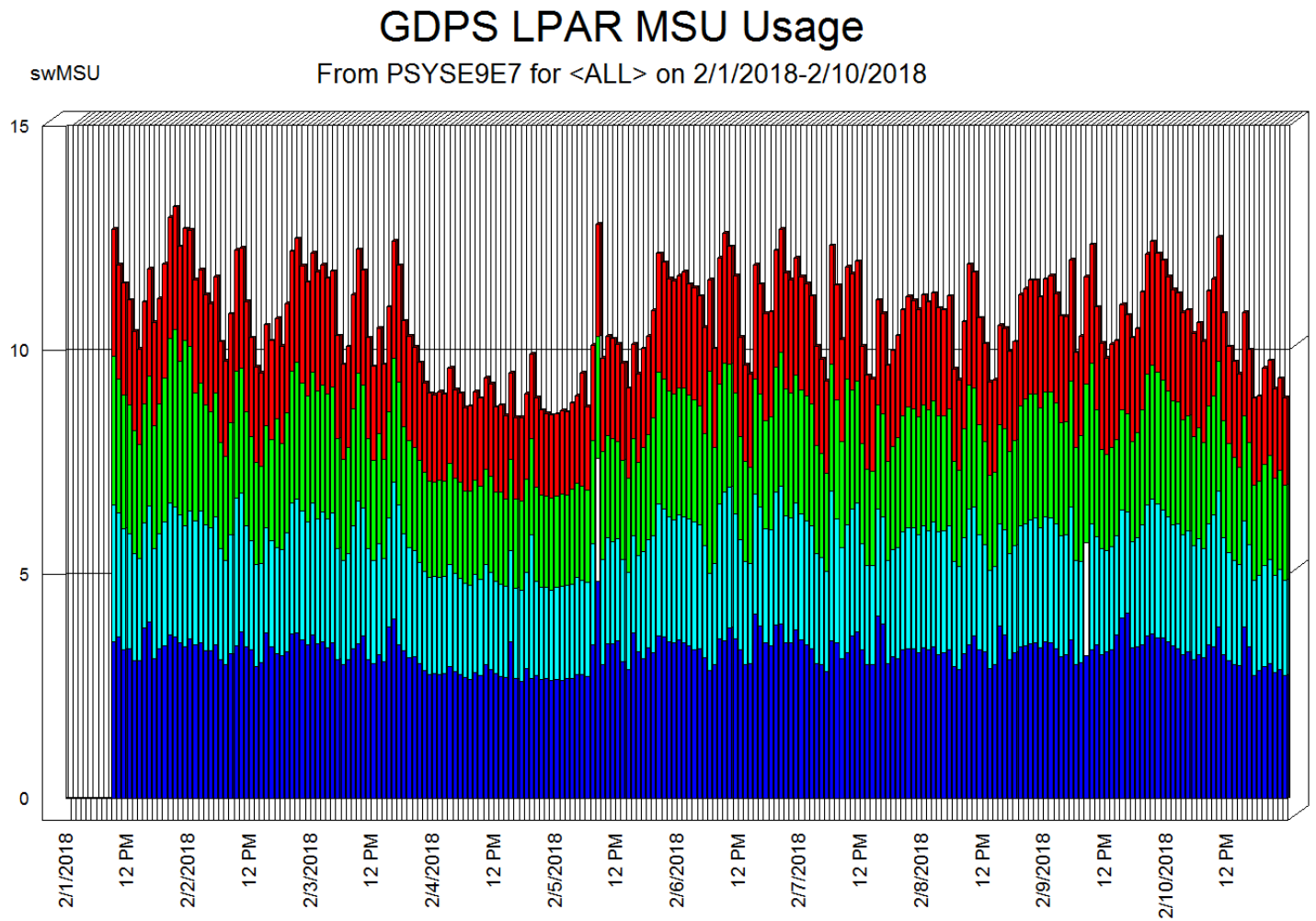
Eliminated 2nd

Eliminated 1st

GDPS Entitlement
43% of CPC

IBM GDPS Recommendation
GDPS > LPAR supporting

GDPS MSU Usage



GDPS

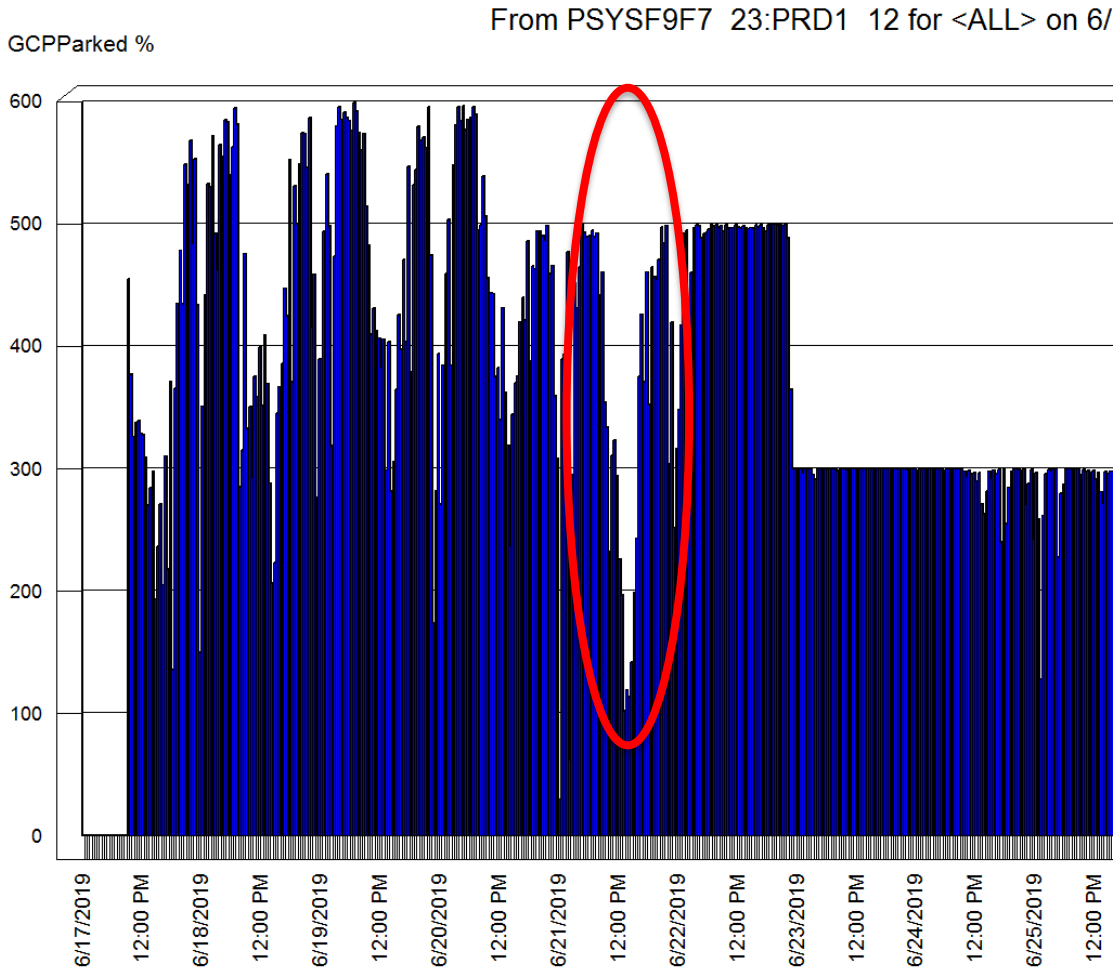
Used

- **1.6%**
- **< 14 MSUs**

Entitlement

- **47% CPC**
- **413 of 880 MSUs**

PRD1 Using Significant VL



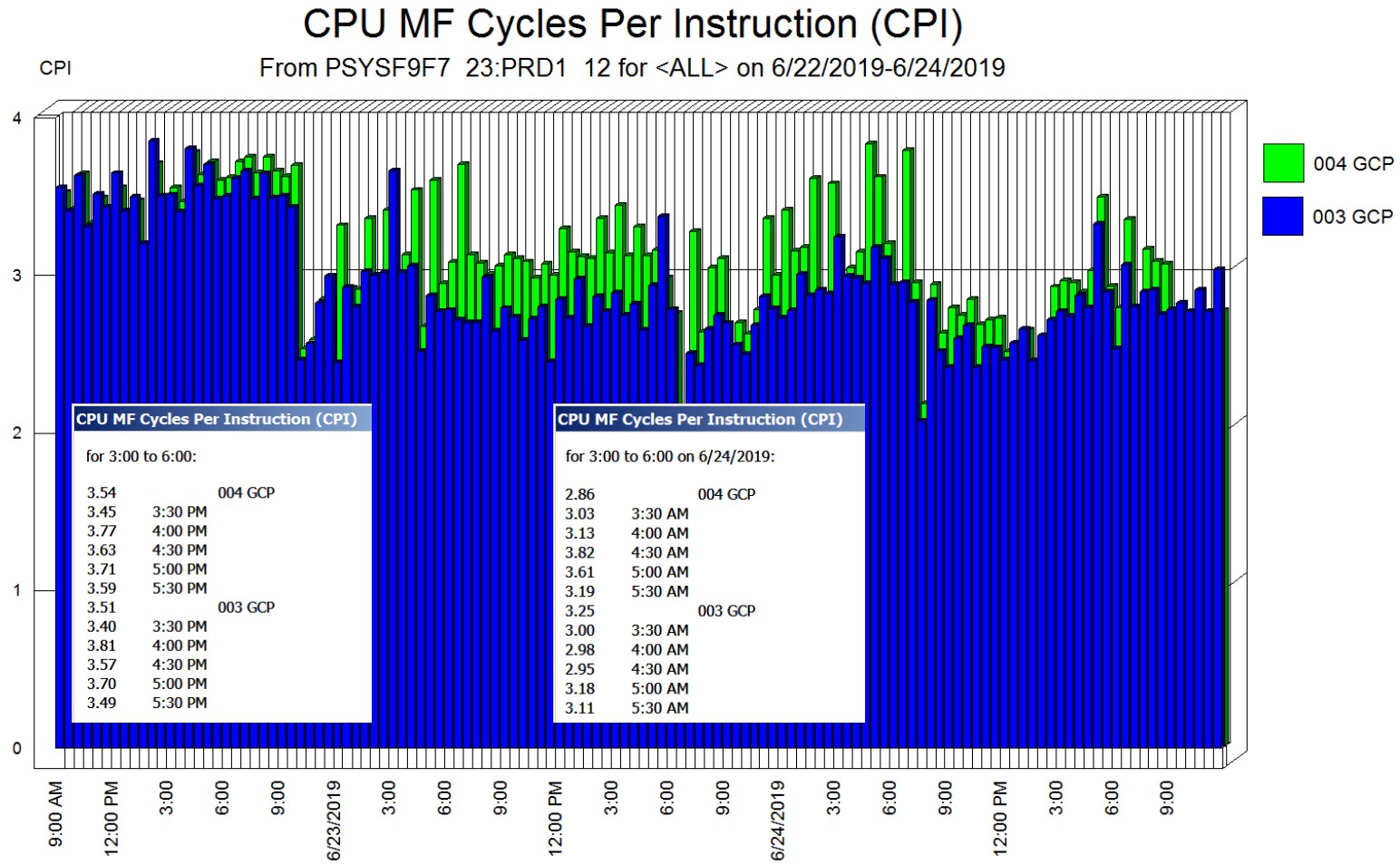
Before

- 2 VH, 2 VM
- 6 VLs
  - 2-4 un-parked

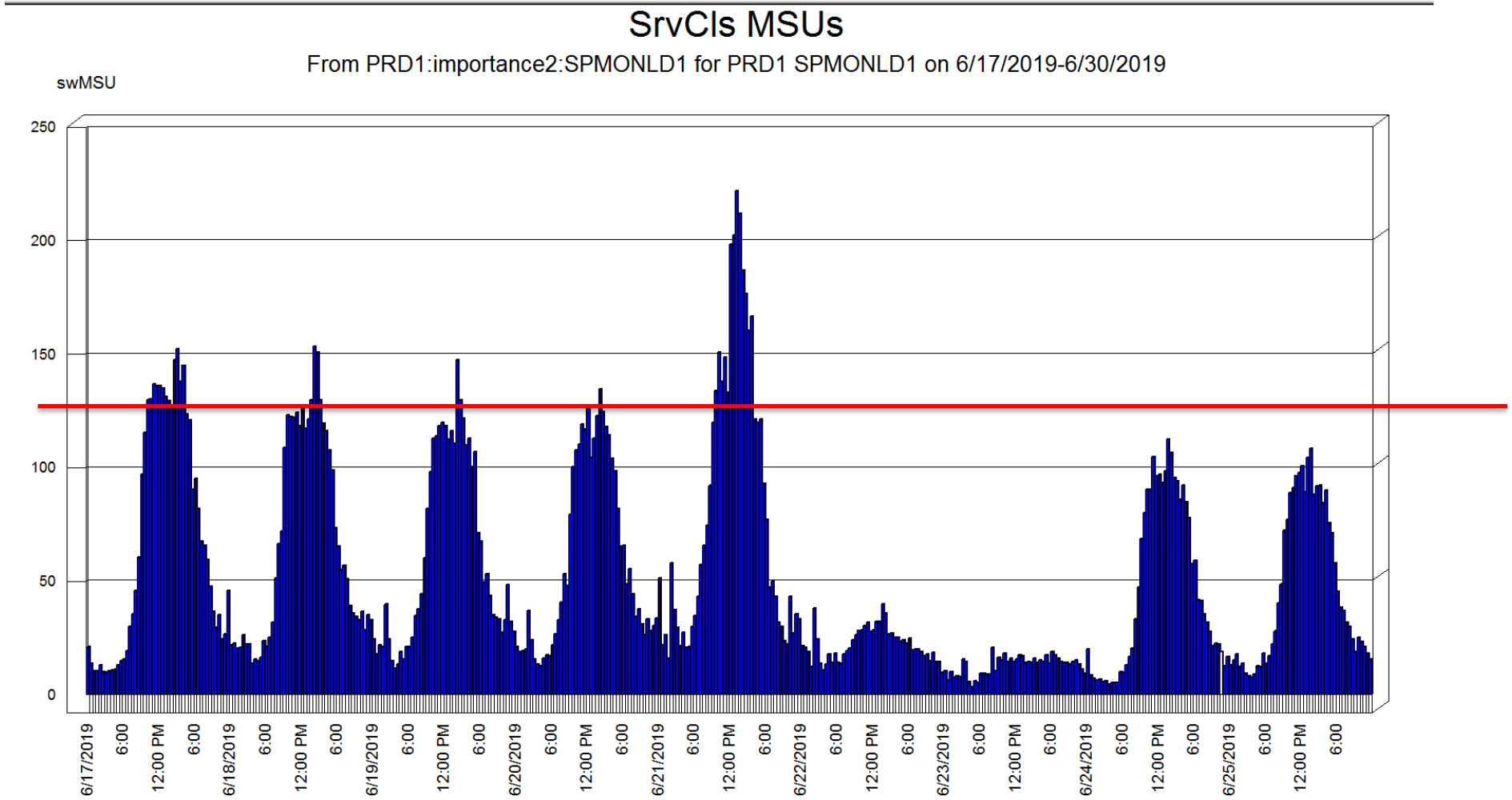
After

- 5 VH, 1 VM
- 3 VLs
  - rarely un-parked

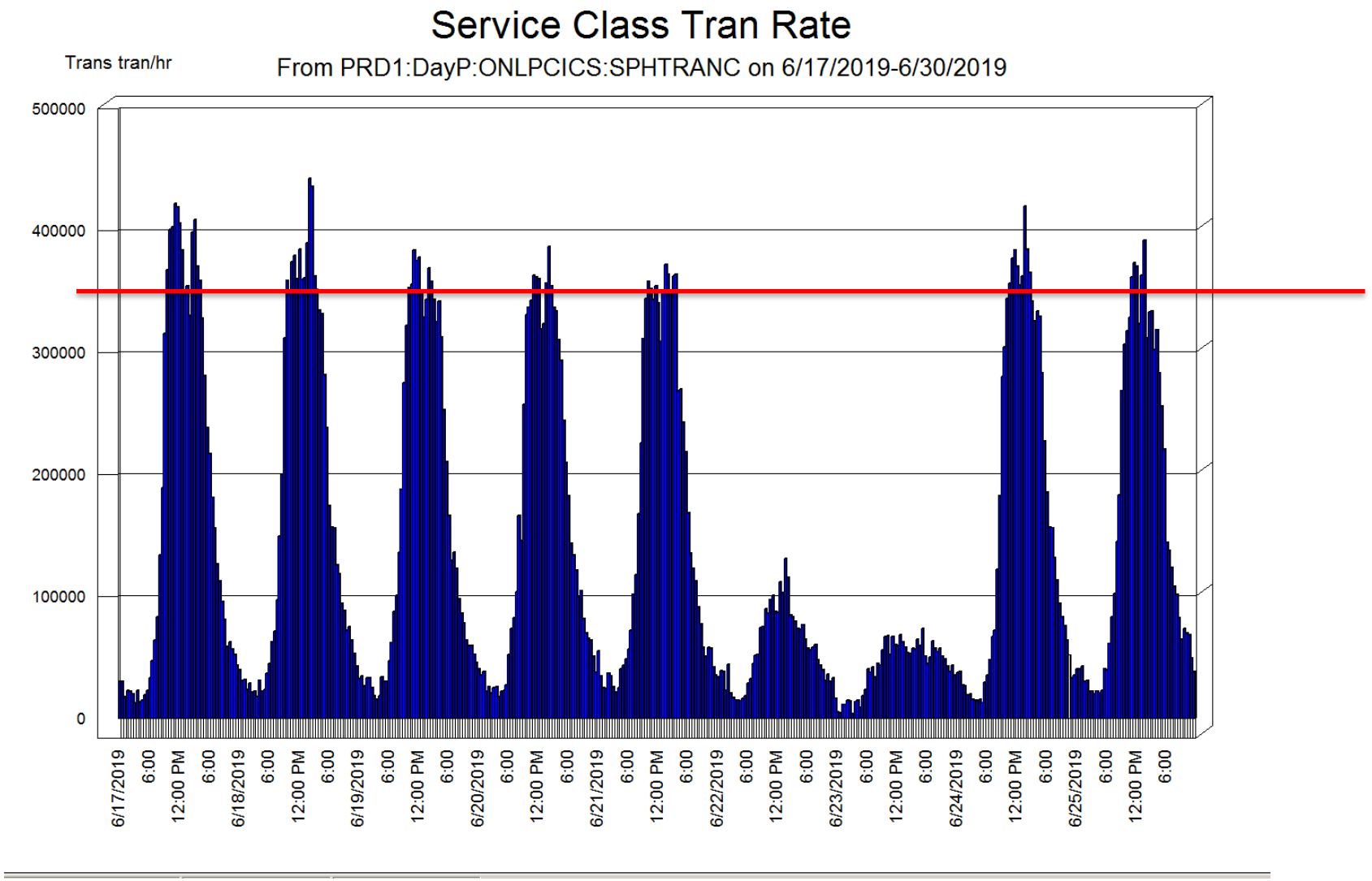
Before / After Last 2 eliminated



Online SrvCls MSU Drop



Lower MSUs on Same #Tran



Customer Y – Cache Tuning Results

Large Real Storage to Map

Fewer Translation Look-aside Buffers – a single TLB entry fulfills many more address translations than ordinary base or large page.

Better TLB coverage - improves performance:

– TLB Misses - Decreased

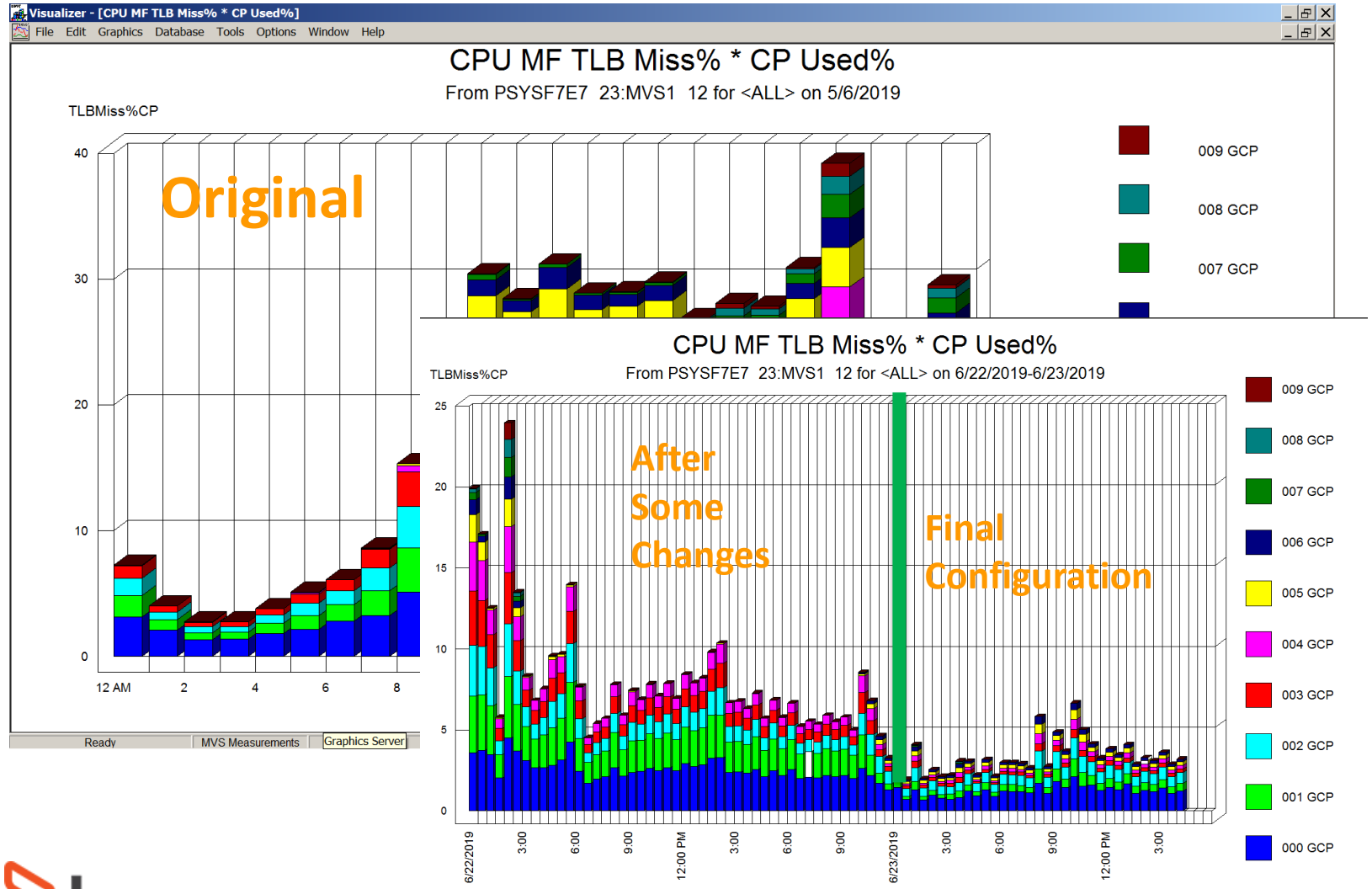
– DAT Faster - less time converting virtual addresses into physical addresses

– Chip Cache More efficient - Uses less real storage to maintain DAT structure

2 GB memory page (EC12+) is

- **2048 times larger than a 1 MB large page**
- **524,288 times larger than an ordinary 4 KB base page.**

TLB Miss % - Impact



Was 30-40% of 1 CP

Now Typically

2-5%

Save > 25% of 1 CP of 10 way or 2.5%

bmc Software - GSE UK Conference 2019

GUIDE SHARE EUROPE UK REGION

Tuesday 5th November

Start Time	End Time	Stream	Room	Title	Speaker
16:45	17:45	zCMPA	Woodcote	Hiperdispatch – SLA improvements & MSU reductions	Donald Zeunert
16:45	17:45	Db2	Nurburgring	MLC – I'm paying HOW MUCH for Db2?	Phil Grainger

Wednesday 6th November

Start Time	End Time	Stream	Room	Title	Speaker
11:45	12:45	IMS	Wellington B	Modernizing IMS Change Management	David Schipper
13:45	14:45	IMS	Wellington B	IMS10: Using Real-Time IMS Data for Security Analysis	Nick Griffin
16:30	19:30	IMS	Wellington B	Innovative Customer Solutions to IMS Challenges	David Schipper

Thursday 7th November

Start Time	End Time	Stream	Room	Title	Speaker
09:00	10:00	Db2	Nurburgring	Putting the capital A in 'Agile on the mainframe'	Tony Poole
11:45	12:45	Db2	Nurburgring	Express Yourself	Marcus Davage

bmc

Please submit your session feedback!

Do it online at http://conferences.gse.org.uk/2019/feedback/LF

This session is LF

1. What is your conference registration number?

This is the three digit number on the bottom of your delegate badge

2. Was the length of this presention correct?

1 to 4 = "Too Short" 5 = "OK" 6-9 = "Too Long"

1 2 3 4 5 6 7 8 9

3. Did this presention meet your requirements?

1 to 4 = "No" 5 = "OK" 6-9 = "Yes"

1 2 3 4 5 6 7 8 9

4. Was the session content what you expected?

1 to 4 = "No" 5 = "OK" 6-9 = "Yes"

1 2 3 4 5 6 7 8 9

Dank u

Danke

謝謝

Thank You

Dzięki

Grazie

Bring IT to Life.™

Gracias

どうも ありがとう

רב תודות

Спасибо!

Merci

多謝

धन्यवाद

Tak

Obrigado

감사합니다

مرسي

Supplemental

More info

Best Practice document for defining logical CPs and zIIPsto an LPAR

- URL: www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/TD106388

WLM LPAR Design Tool – for HiperDispatch

URL: https://ibm.biz/BdZTVw

Logical CP Management – Old Changes

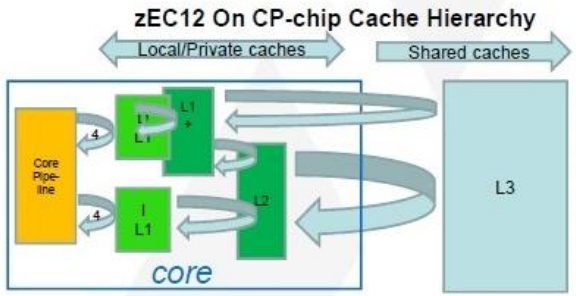
z13 MCL Fix N98779 D22H Bundle S14 - changed how PR/SM assigned VH and VM processors for “small” LPARs with weight within {1.5 to 2.0} Physical CP share

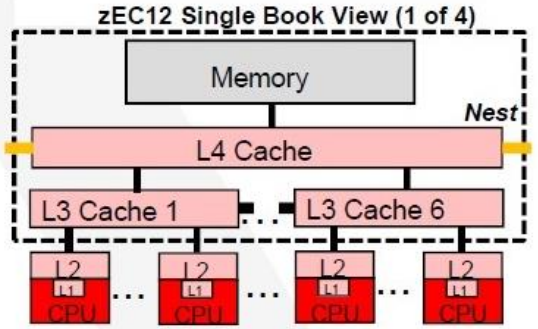
- Before were assigns 1 VH and 1 VM
- Now assigned 2 VM processors
- **Benefit due** to PR/SM optimizing placement of VH processors but not necessarily the placement of VM processors
    - Causes the 2 VMs to most often be placed close together on the same or adjacent chips

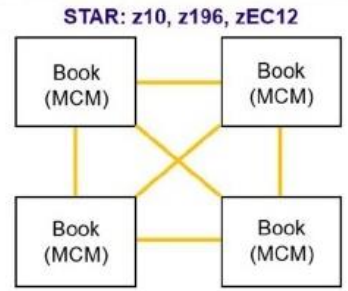
OA47968 (WLM Hiperdispatch enhancements) - Changes is the park, un-park sequence of VL processors, ensures VLs with logically high processor numbers are parked first because those are very often not closely located to VH or VM processors

zProcessor Cache Changes

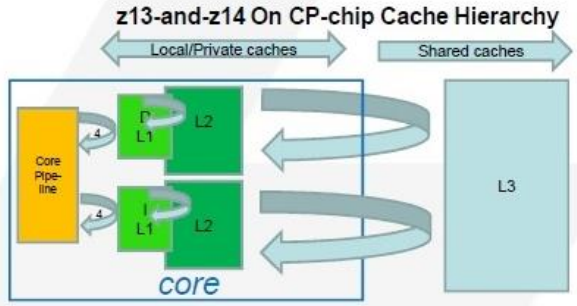
zEC12
L1 private 64k i + 96k d
L2 private 1 MB i + "L1+" 1 MB d
L3 shared 48 MB per CP chip
L4 shared 384 MB per **book**
6 cores + 1 L3 / CP chip
6 CP chips + 1 L4 / **book**
4 **books** (STAR) / CEC

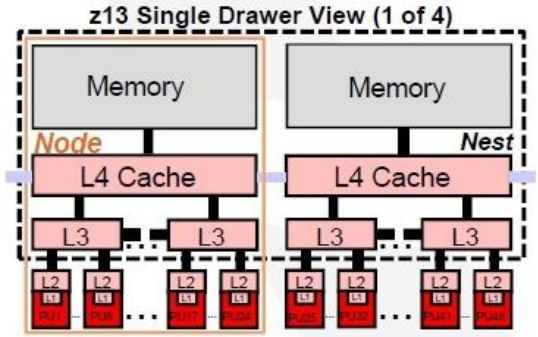


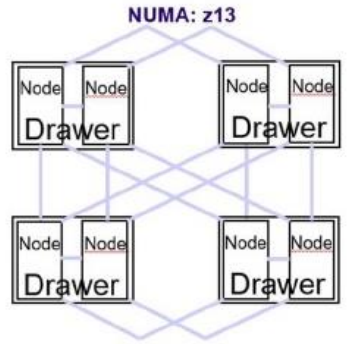




z13
L1 private 96k i, 128k d
L2 private 2 MB i + 2 MB d
L3 shared 64 MB / chip
L4 shared 480 MB / **node**
8 cores + 1 L3 / CP chip
3 CP chips + 1 L4 / **node**
2 **nodes** / **drawer**
4 **drawers** (NUMA) / CEC







z14
L1 private 128k i, 128k d
L2 private 2 MB i, 4 MB d
L3 shared 128 MB / chip
L4 shared 672 MB / **drawer**
10 cores + 1 L3 / CP chip
3 CP chips / **cluster**
2 **clusters** + 1 L4 / **drawer**
4 **drawers** (numa) / CEC