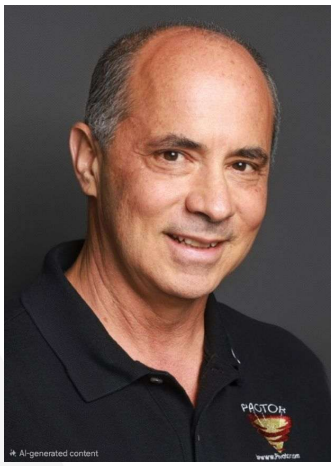
Processor MSU Consumption Analysis

Peter Enrico
Enterprise Performance Strategies
Peter.Enrico@epstrategies.com



SHARE Orlando
February 2026
Session TECH_201

Session TECH_201

Contact, Copyright, and Trademark Notices

Questions?

Send email to Peter at Peter.Enrico@EPStrategies.com, or visit our website at http://www.epstrategies.com or http://www.pivotor.com.

Copyright Notice:

Trademarks:

Enterprise Performance Strategies, Inc. presentation materials contain trademarks and registered trademarks of several companies.

The following are trademarks of Enterprise Performance Strategies, Inc.: **Health Check®, Reductions®, Pivotor®**

The following are trademarks of the International Business Machines Corporation in the United States and/or other countries: IBM®, z/OS®, zSeries® WebSphere®, CICS®, DB2®, S390®, WebSphere Application Server®, and many others.

Other trademarks and registered trademarks may exist in this presentation

Abstract

Processor Consumption Analysis

The first step to any processor analysis is to understand your processor configuration and settings. The second step is to understand what workloads, address spaces, and transactions are consuming the fixed processor resource. It is only after understanding what and how the processor is being consumed can you conduct any sort of processor tuning or optimization exercise.

During this presentation, SHARE Distinguished Speaker Peter Enrico will show you how to conduct a processor resource consumption analysis. You will be provided with a top-down approach to better understand processor measurements available to help you gain a drilldown insight into how the CPU resource is being consumed, and by what LPARs, Workloads, and transactions. Shown is what is known as a drill down approach for a processor performance analysis.

EPS: We do z/OS performance…

- **Pivotor** – z/OS performance reporting and analysis software and services
  - Not just SMF reporting, but analysis-based reporting based on expertise
  - www.pivotor.com

- **Education and instruction**
  - We teach our z/OS performance workshops all over the world
  - Want a workshop in your area? Just contact me.

- **z/OS Performance War Rooms**
  - Intense, concentrated, and highly productive on-site performance group discussions, analysis and education
  - Amazing feedback from dozens of past clients

- **MSU Reduction Exercises**
  - The goal is to reduce the MSU consumption of your applications and environment

- **Information**
  - We present around the world and participate in online forums
  - https://www.pivotor.com/content.html
    https://www.pivotor.com/webinar.html

z/OS Performance workshops available

During these workshops you will be analyzing your own data!

- Essential z/OS Performance Tuning
  - March 30 – April 3, 2026 (4 days, excl Wednesday the 1st)
- WLM Performance and Re-evaluating Goals
  - June 22 – 26, 2026 (4 days, excl Wednesday the 24th)
- Parallel Sysplex and z/OS Performance Tuning
  - May 12-13 2026
- Also… please make sure you are signed up for our free monthly z/OS educational webinars! (email contact@epstrategies.com)

Like what you see?

- The z/OS Performance Graphs you see here come from Pivotor™

- If you don't see them in your performance reporting tool, or you just want a free cursory performance review of your environment, let us know!
  - We're always happy to process a day's worth of data and show you the results
  - See also: http://pivotor.com/cursoryReview.html

- We also have a free Pivotor offering available as well
  - 1 System, SMF 70-72 only, 7 Day retention
  - That still encompasses over 100 reports!

Like what you see?

- Free z/OS Performance Educational 2026 Winter/Spring webinars!
  - The titles for our Winter/Spring 2026 webinars are as follows:
    - *New Year's Resolutions for z/OS Performance and Capacity People*
    - *How WLM Makes Decisions*
    - *What I Learned about VSAM RLS SMF Data*
    - *z/OS Performance Spotlight: Some Top Things You May Not Know*
    - *Building a Strong Foundation When You're New to z/OS Performance*
    - *Wait…Do We Need to Re-evaluate our WLM Goals?*
    - *z15 to z16 to z17 – What has changed?*
    - *Evaluating in the Mainframe Environment*
    - *Managing Workload Manager: Multiple Sysplexes and Asymmetric Sysplexes*
    - *Introduction to z Processor Measurements*
    - *(more to be announced)*

- If you want a free cursory review of your environment, let us know!
  - We're always happy to process a day's worth of data and show you the results
  - See also: http://pivotor.com/cursoryReview.html

EPS presentations this week

What	Who	When	Where
z/OS Performance Management If You Only Have 20 Minutes A Day	Scott Chapman	Mon 9:45	Salon 14
PSP: z/OS Performance Tuning - Some Top Things You May Not Know	Peter Enrico Scott Chapman	Tie 13:15	Salon 18
Planning Your Next Mainframe Processor Upgrade in 2026	Scott Chapman	Tue 15:45	Salon 15
Processor MSU Consumption Analysis	Peter Enrico	Wed 13:15	Salon 14
Can We All Share Fairly? Detection and Remediation of inter-LPAR Performance Impacts	Scott Chapman	Wed 14:30	Salon 14
Standard z/OS Measurements When Monitoring Transactions	Peter Enrico	Thu 13:15	Salon 19

Why do we care about MSUs?

The Many Facets of CPU & MSU Measurements

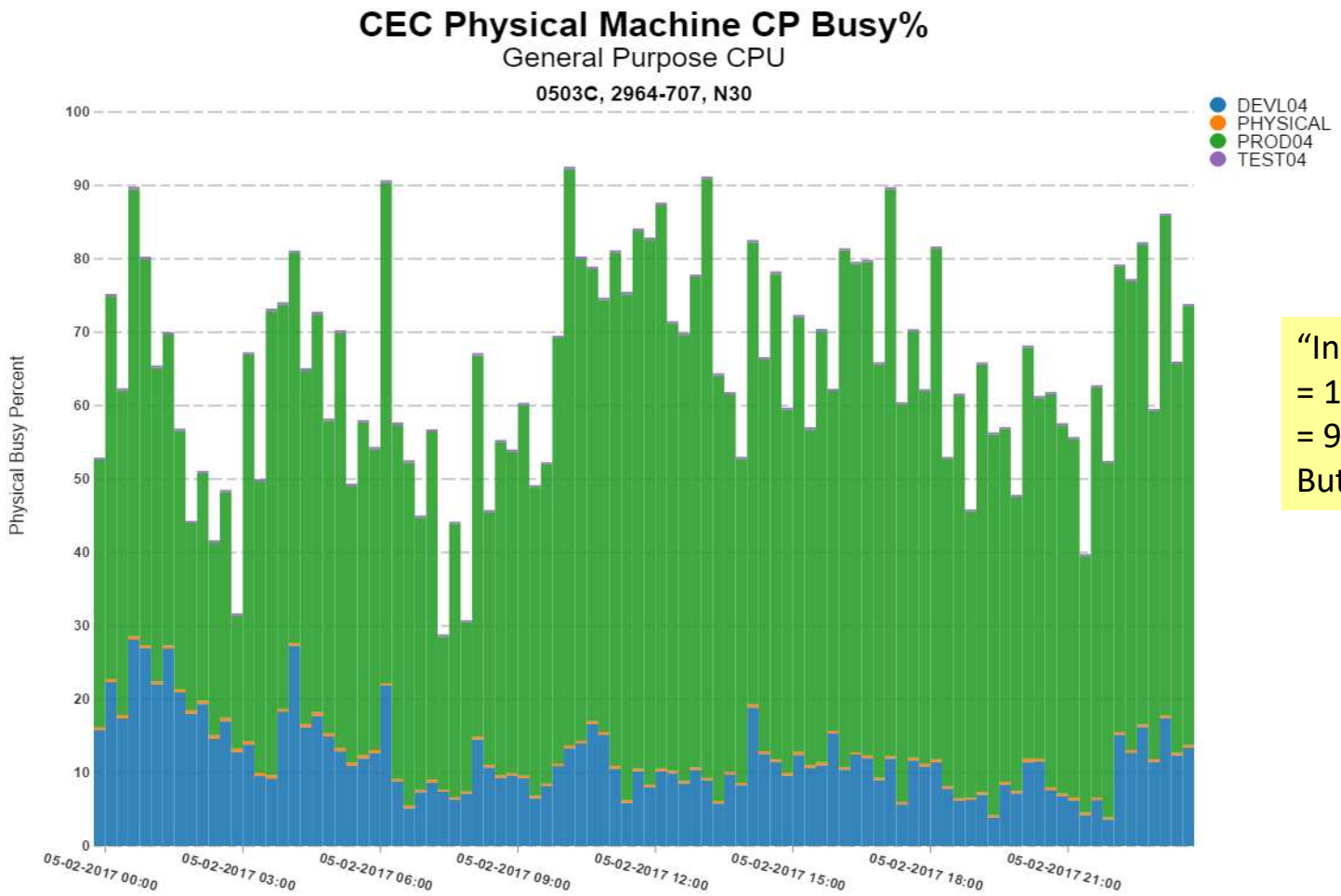
- Mainframe CPU measurements are varied and sometimes confusing
- Yet understanding CPU & MSU measurements is very important
  - Software costs are usually the largest part of the mainframe budget
  - Software costs are usually driven by CPU capacity and/or consumption
- We need to understand CPU and MSU capacity
  - How much work can our processor process?
- We need to understand CPU consumption
  - How much of our CPU capacity are we using?
- We also need to understand how our CPU and MSU capacity and consumption impacts our performance, as well as our software bills

Examples of Monitoring MSU Usage

MSUs used in Software Pricing

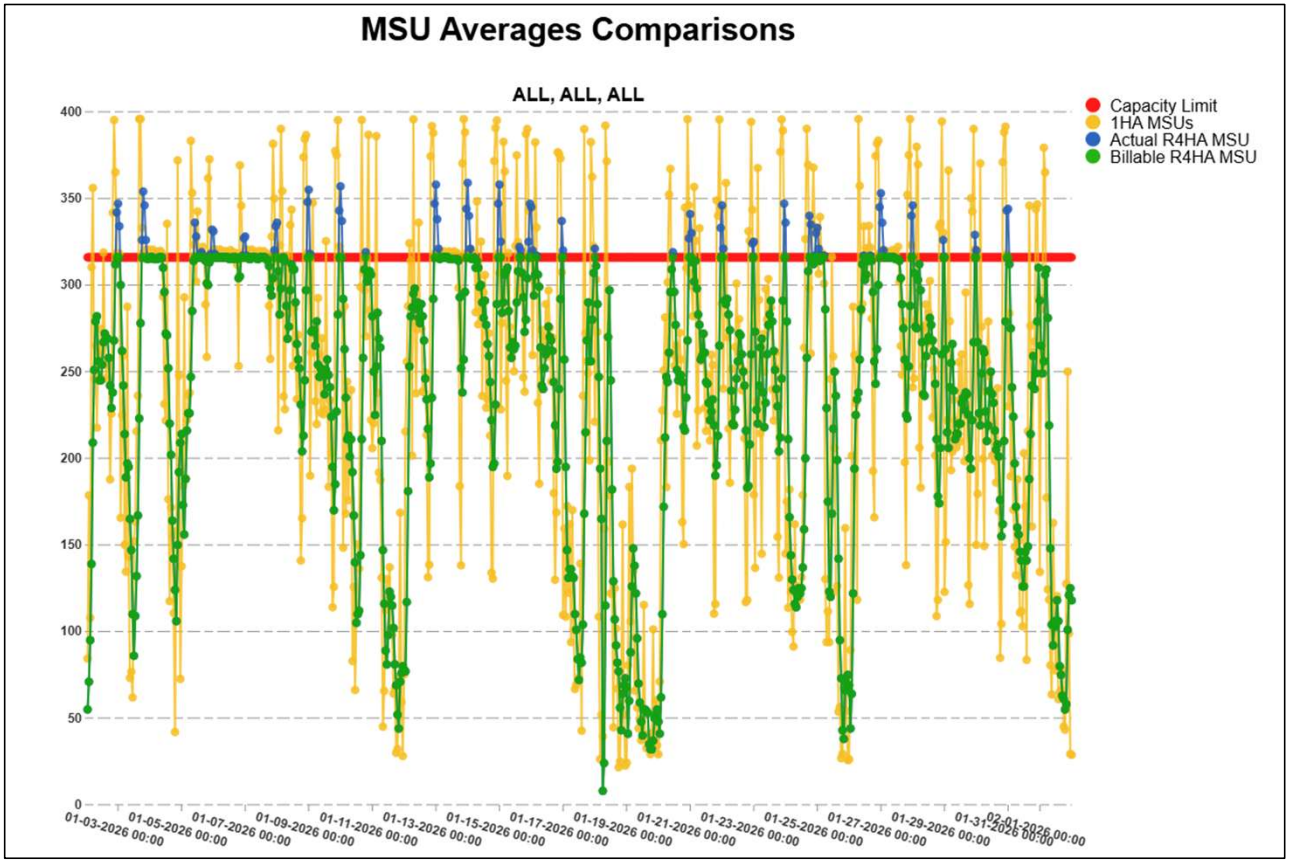
- At a super high-level MLC vs TFP is capacity usage versus consumption
  - MLC
    - MLC cost based on peak Rolling 4-Hour Average (R4HA) MSU utilization in the MLC month (2nd to 1st)
    - In a sense, this is based on the peak capacity used, so capacity limits are important
    - Other factors involved based on agreements, etc..
  - TFP
    - Individually negotiated agreements between IBM and customer based on MSU usage
    - Using MSUs to mean MSU-hours as a measure of consumption
- Super high-level analogy
  - R4HA MSU measure vs TFP MSU measure is similar in concept to looking at peak CPU utilizations versus consumption of CPU seconds

What Capacity Are You Charged For?



"Installed" Capacity
= 1212 MSUs
= 9964 PCI
But not using all of it

MSU Usage – R4HA Analysis Example 1

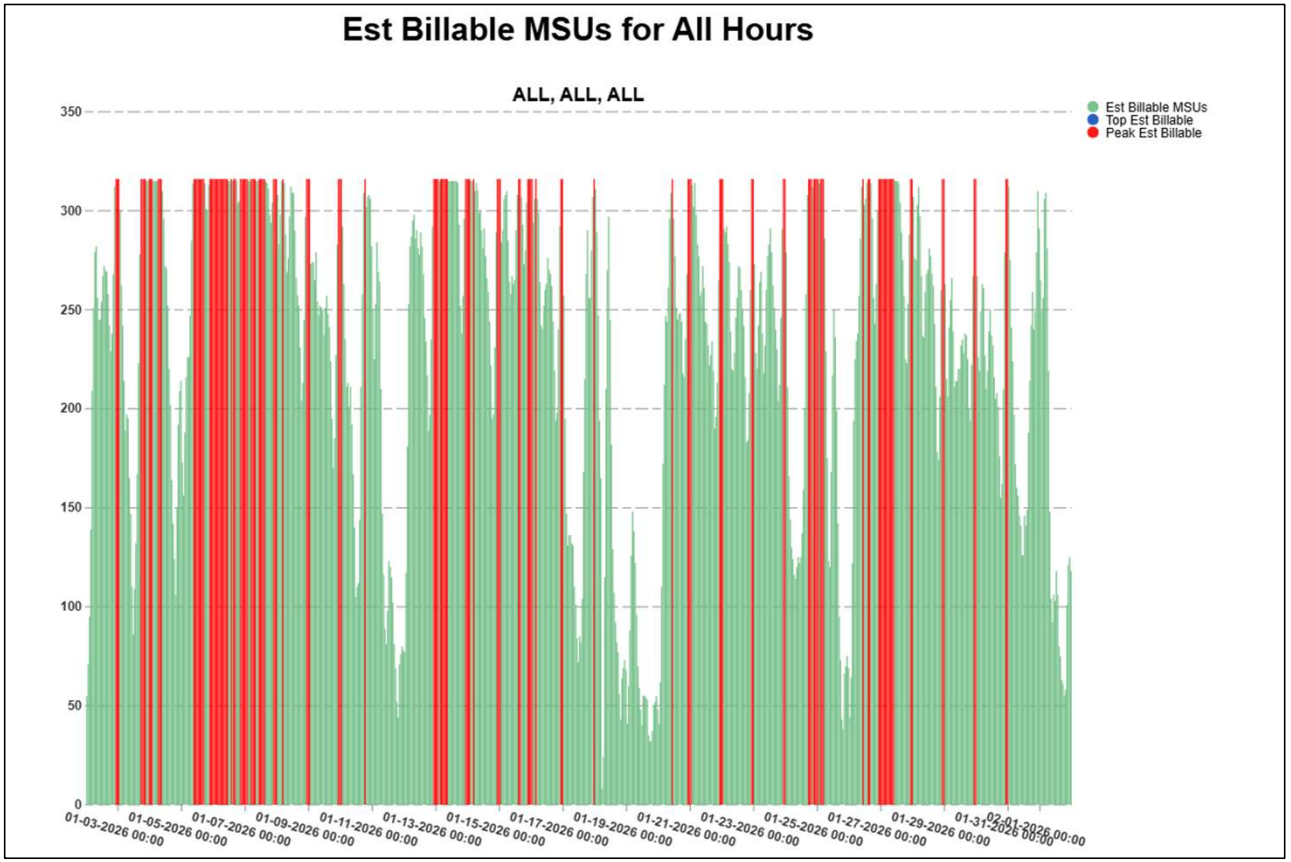


This chart reports the MSU averages for the billing month.

- Red line: Capacity limit
- Yellow line: 1HA MSUs
- Blue line: R4HA
- Green line: Billable MSUs

Because the capacity limit was reached, capping occurred at the capacity limit.

MSU Usage – R4HA Analysis Example 1

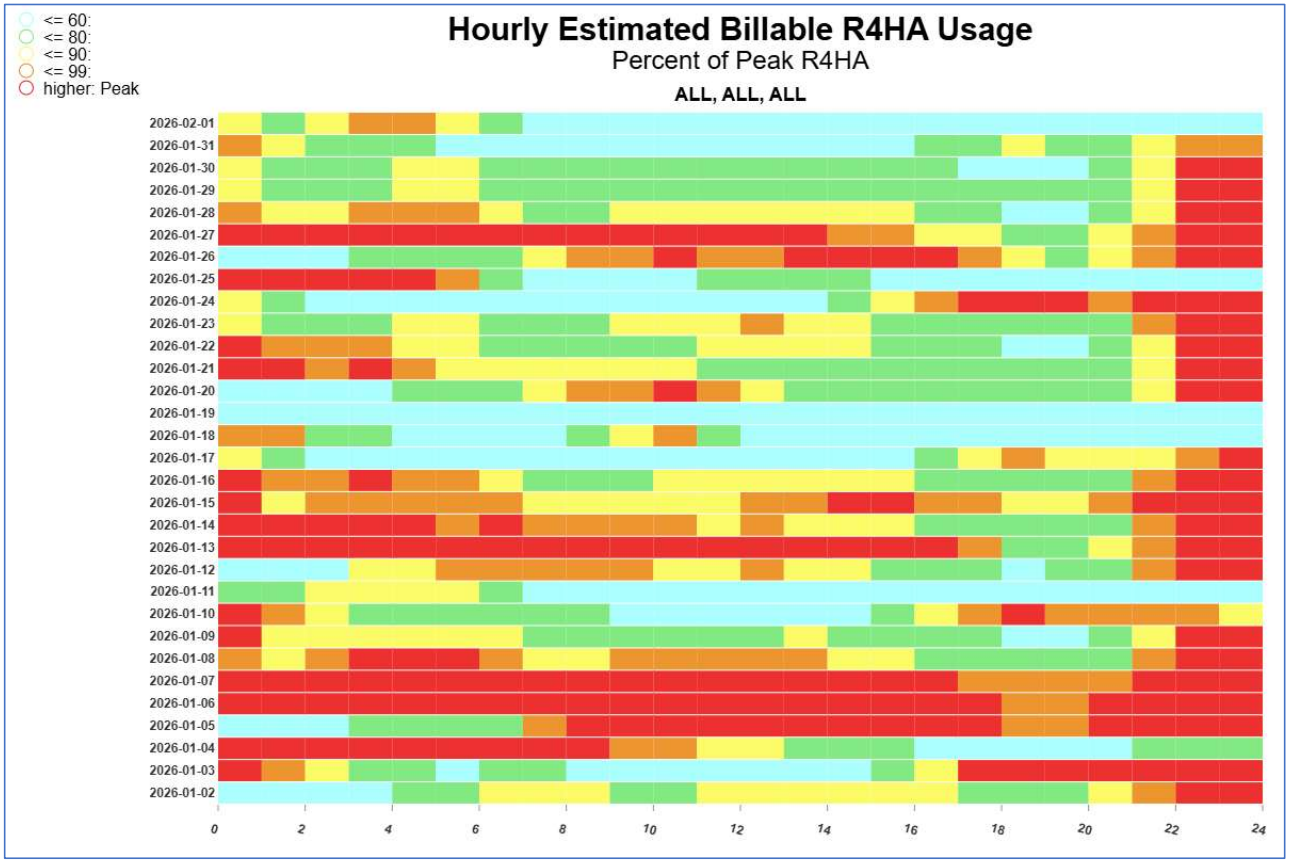


This chart reports the estimated billable MSUs for all hours of the billing month.

The red indicates intervals for the peak of the month. There are many red lines, thus the peak was hit many times.

As shown on the previous slide, the environment is regularly being capped.

MSU Usage – R4HA Analysis Example 1

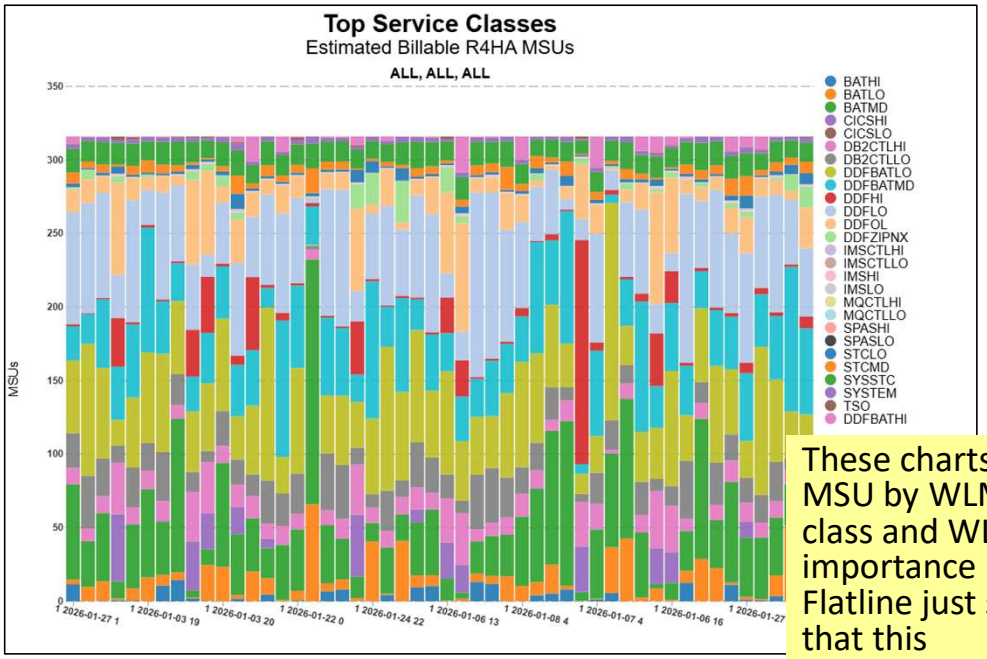


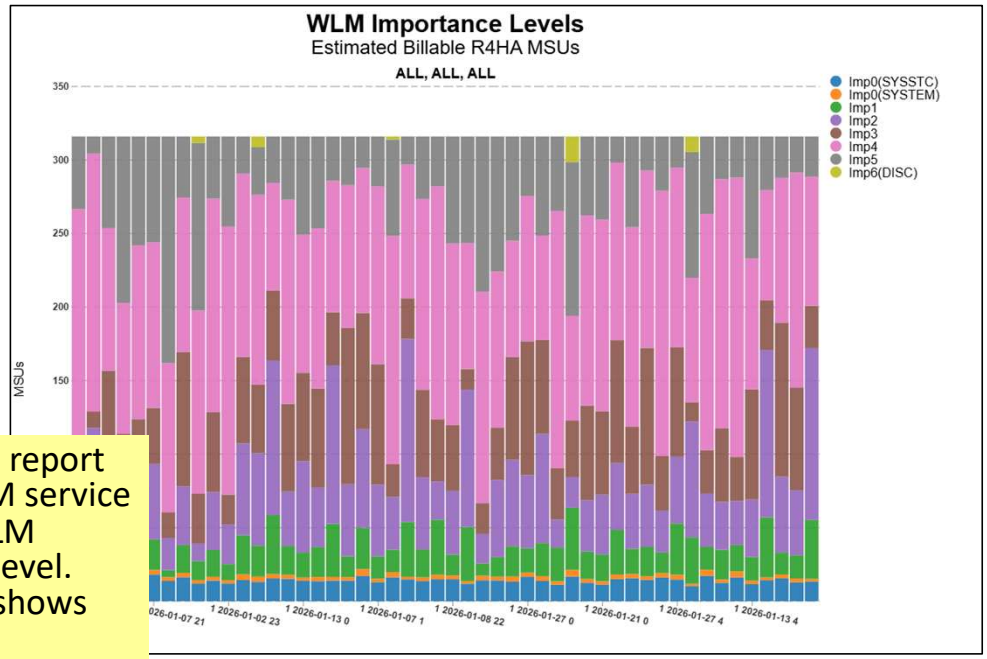
This heat chart reports the peak MSU consumption time periods.

The red indicates intervals for the peak of the month. There are many red lines, thus the peak was hit many times.

As shown on the previous slide, the environment is regularly being capped.

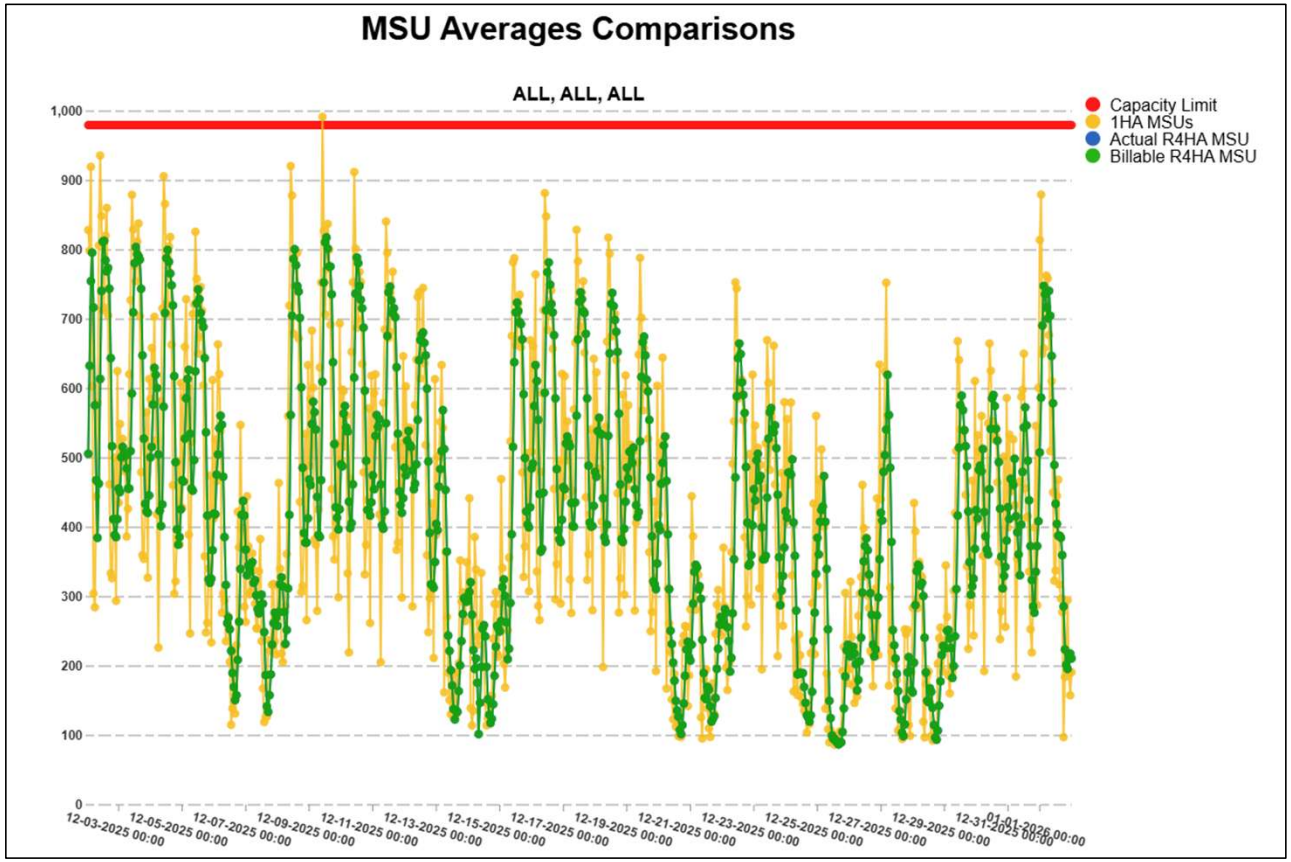
MSU Usage – R4HA Analysis Example 1





These charts report MSU by WLM service class and WLM importance level. Flatline just shows that this environment is regularly capped.

MSU Usage – R4HA Analysis Example 2



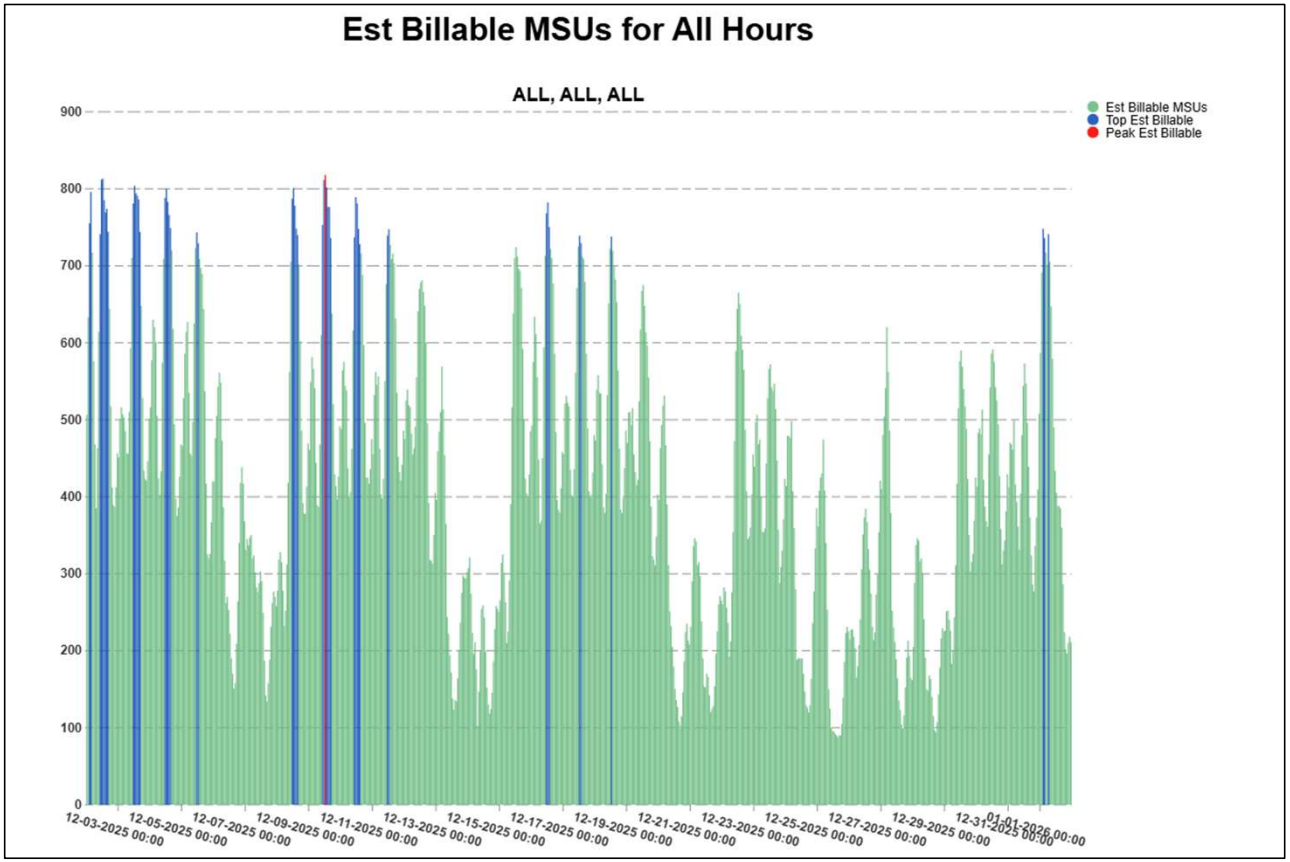
This chart reports the MSU averages for the billing month.

In this example, the R4HA never exceeded the capacity limit.

Thus, the R4HA MSUs and Billable MISs are the same.

Billed based on peak.

MSU Usage – R4HA Analysis Example 2



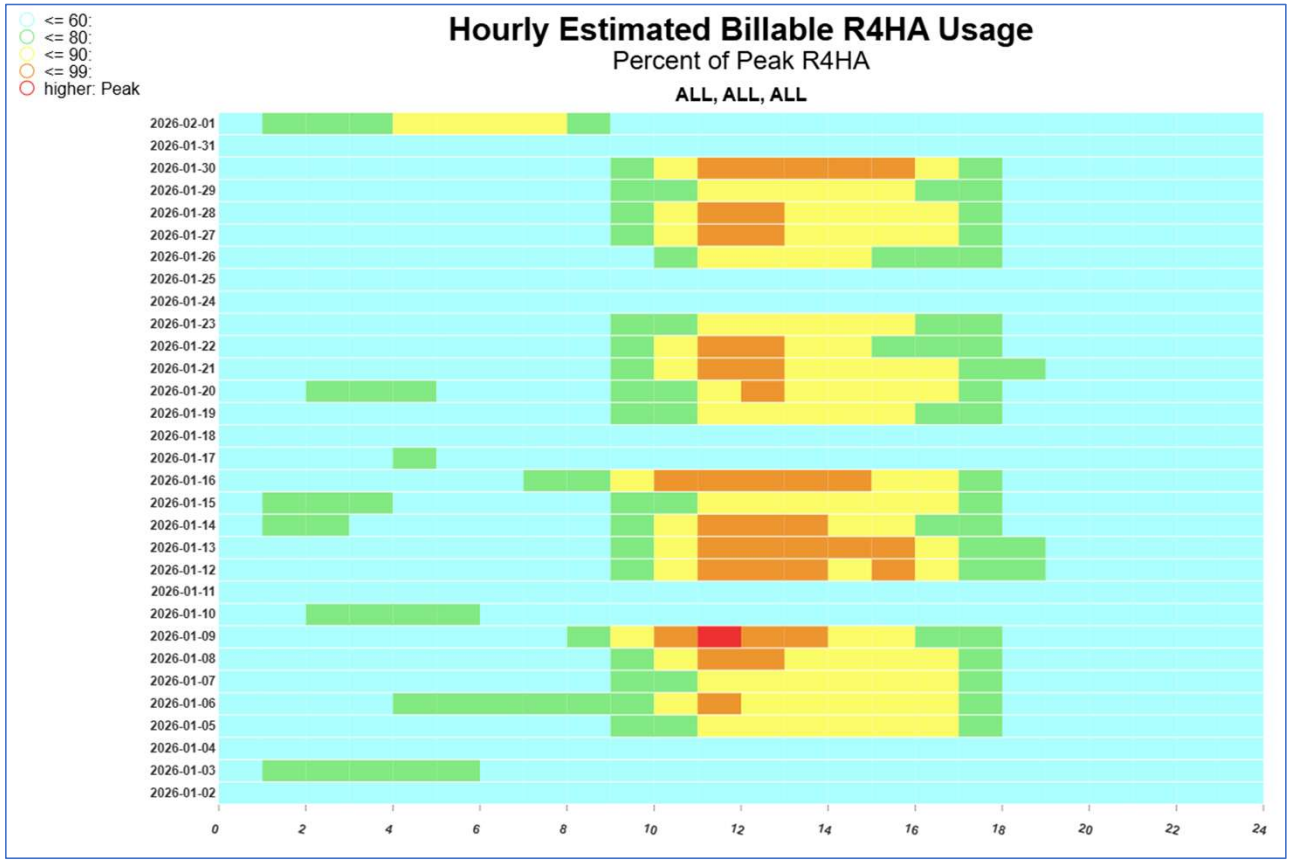
This chart reports the estimated billable MSUs for all hours of the billing month.

The red the interval for the peak of the month. There are very few peak intervals. In this case, only one single peak.

The blue indicates the next 49 peaks after the peak.

As shown on the previous slide, the environment is regularly NOT being capped.

MSU Usage – R4HA Analysis Example 2

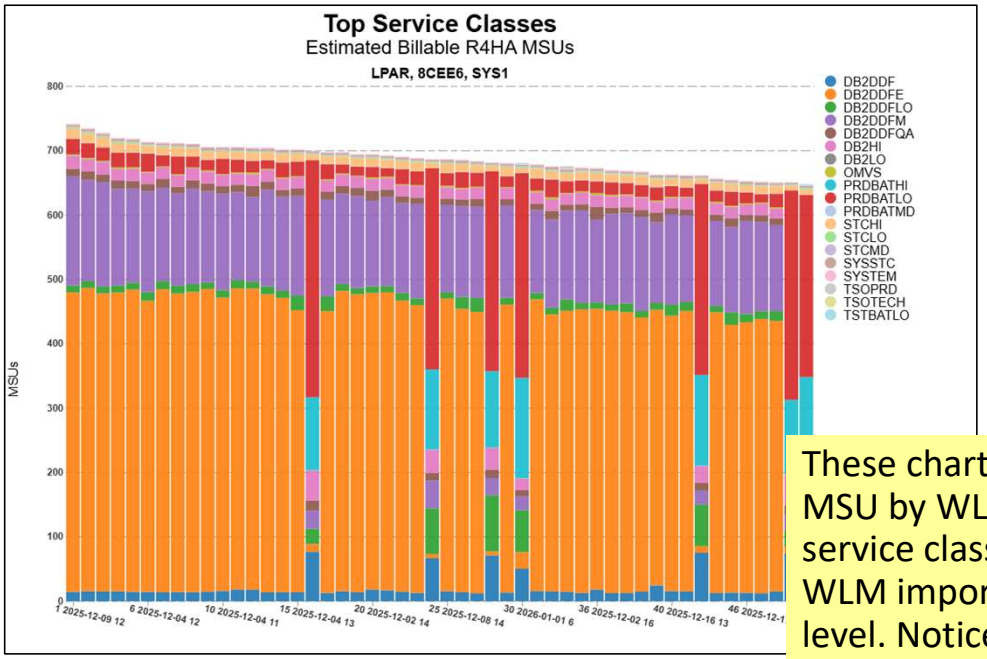


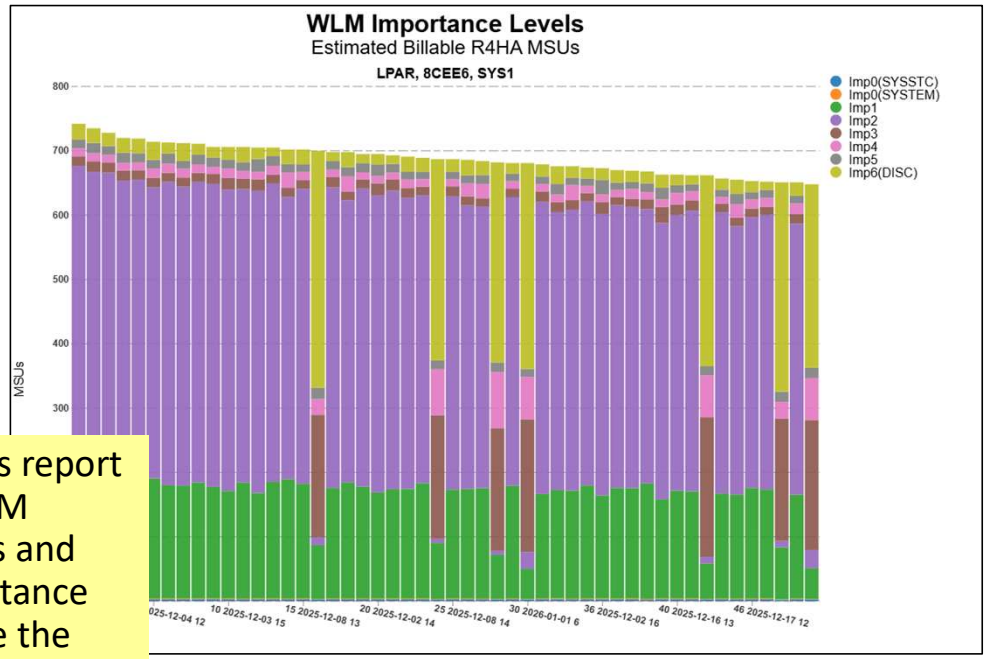
This heat chart reports the peak MSU consumption time periods.

The red indicates intervals for the peak of the month. There are many red lines, thus the peak was hit many times.

As shown on the previous slide, the environment is regularly being capped.

MSU Usage – R4HA Analysis Example 2





These charts report MSU by WLM service class and WLM importance level. Notice the workload vary for the top intervals

Effect of reducing your R4HA

- Remember:

  Reducing the R4HA by 10% will not reduce your bill by 10%

  - Savings is off from incremental (“last”) MSUs—the cheapest ones

- Determine your incremental per-MSU cost
  - Complicated by things not licensed everywhere
  - Complicated by new pricing metrics such as container-based pricing
  - But for the purposes of evaluating an R4HA reduction effort, being in the right ballpark is all that matters

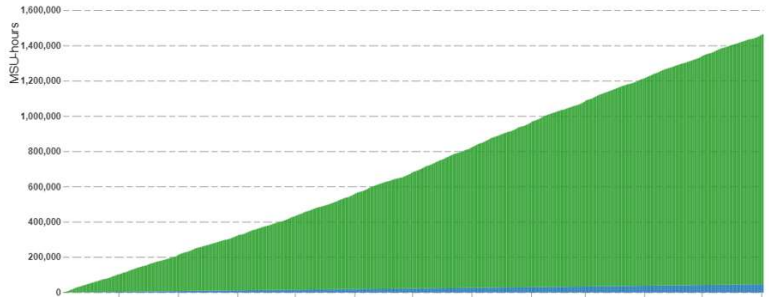
Taylor Fit Pricing (TFP)

- If you do not like R4HA, then Taylor Fit Pricing (announced May 2019)
- Requires z14+, does away with R4HA for MLC
- Individually negotiated agreements between IBM and customer
  - Not standardized pricing!
- Enterprise Capacity Solution
  - Discounted full-capacity pricing (probably only of interest if growing significantly)
- Enterprise Consumption Solution
  - Price based on total consumed MSU-hours for the year
  - Annual commitment level + true-up at end of year
    - Unused MSU-hours can roll over for 1 year
    - Grow above commitment at minimum 50% discount of per MSU-hour cost of baseline
  - Tailored

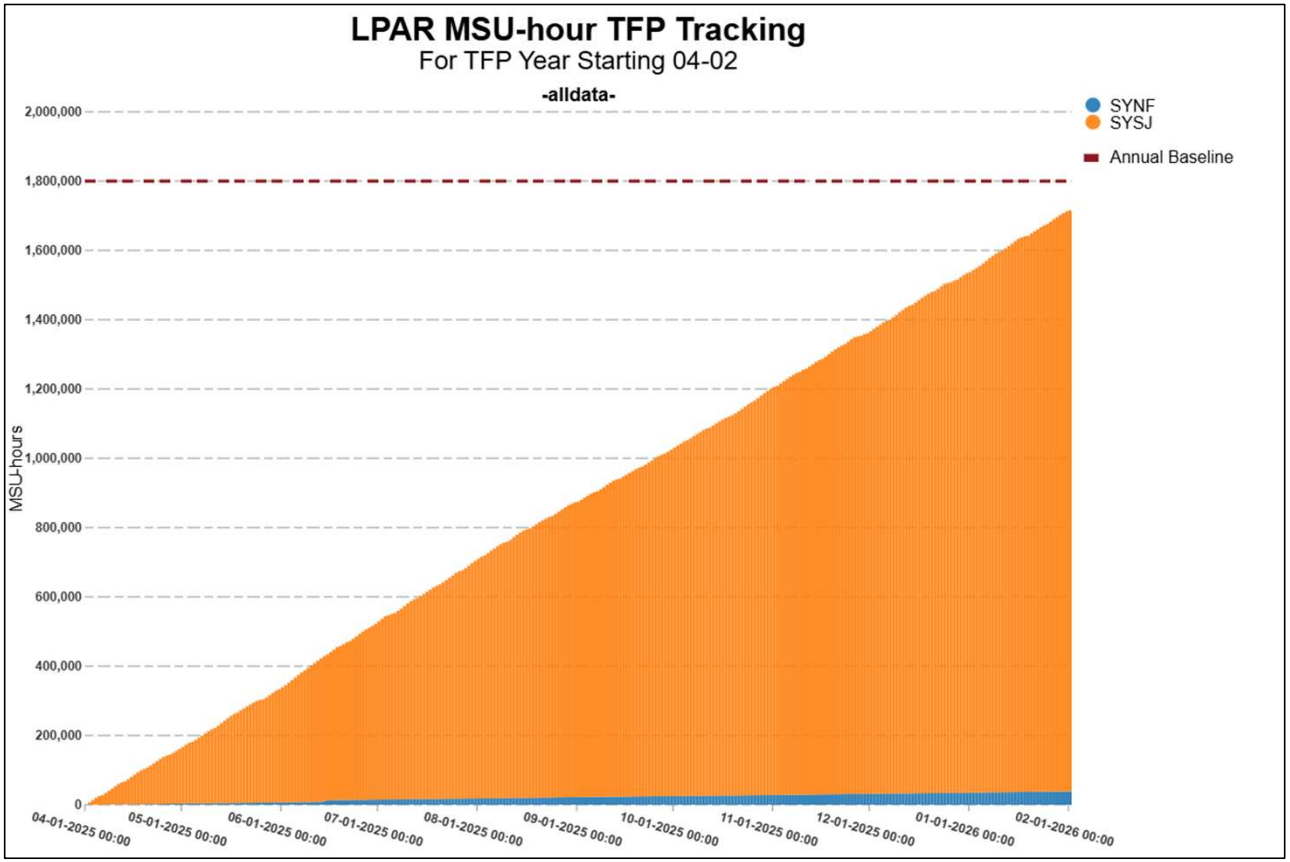
https://www-01.ibm.com/common/ssi/cgi-bin/ssialias?infotype=an&letternum=ENUS219-014&subtype=ca&supplier=897

TFP Terminology

- TFP uses “MSUs” as a measure of consumption
  - MSUs also a measure of total capacity of the machine, e.g. like MIPS or SU/sec
  - TFP is really using MSU-hours (or so I call it)
- Consider 500 MSU machine
  - 15 minute RMF interval show utilization of 80% -- you used 400 MSUs of capacity
  - But that was for 15 minutes, so that’s only 100 MSU-hours
- Really this moves from a “monthly peak rolling 4-hour average MSU consumption” to “annual total hourly average MSU consumption”



MSU Usage – TFP Analysis Example 1

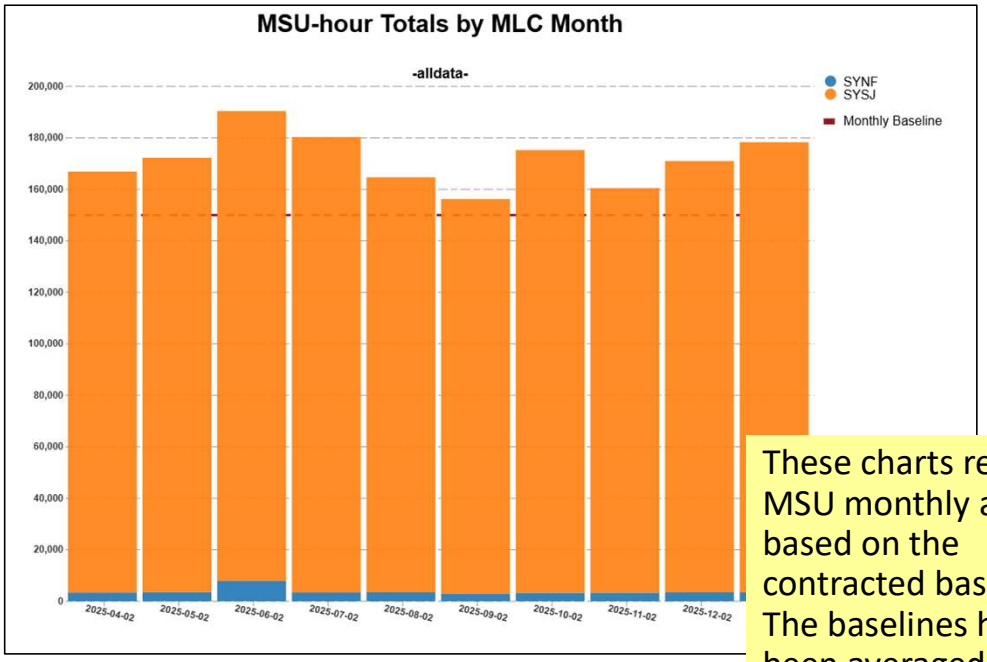


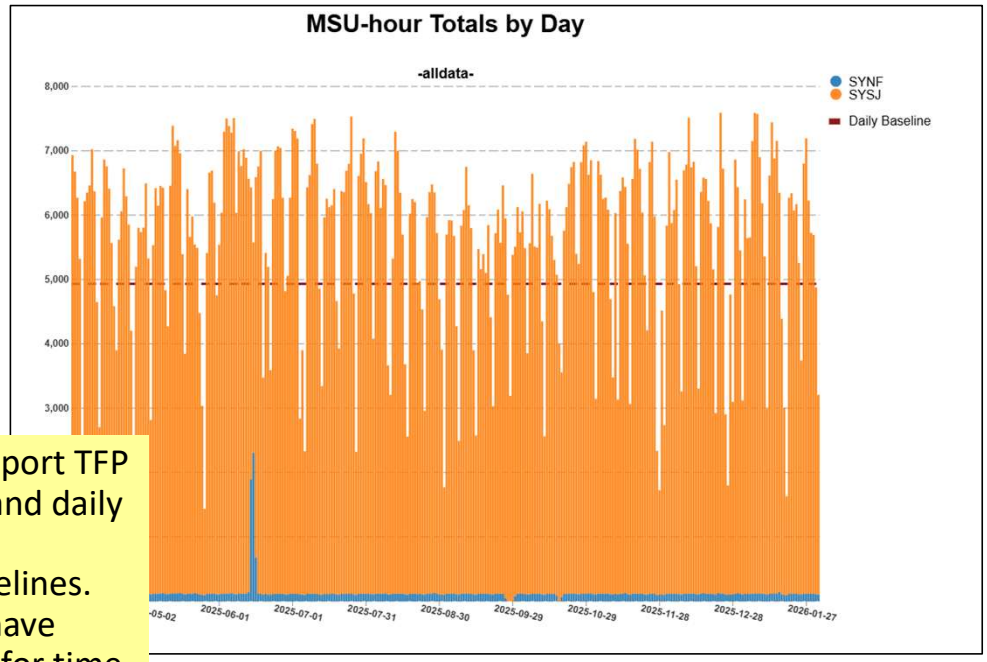
This chart reports the cumulative MSU consumption since anniversary date relative to the annual baseline agreed upon with IBM.

With TFP, all utilization contributes to the software cost.

Note that although there is no separate Dev/Test environment here, the Dev/Test solution should usually be part of a TFP contract, so annual total will be really driven by production totals.

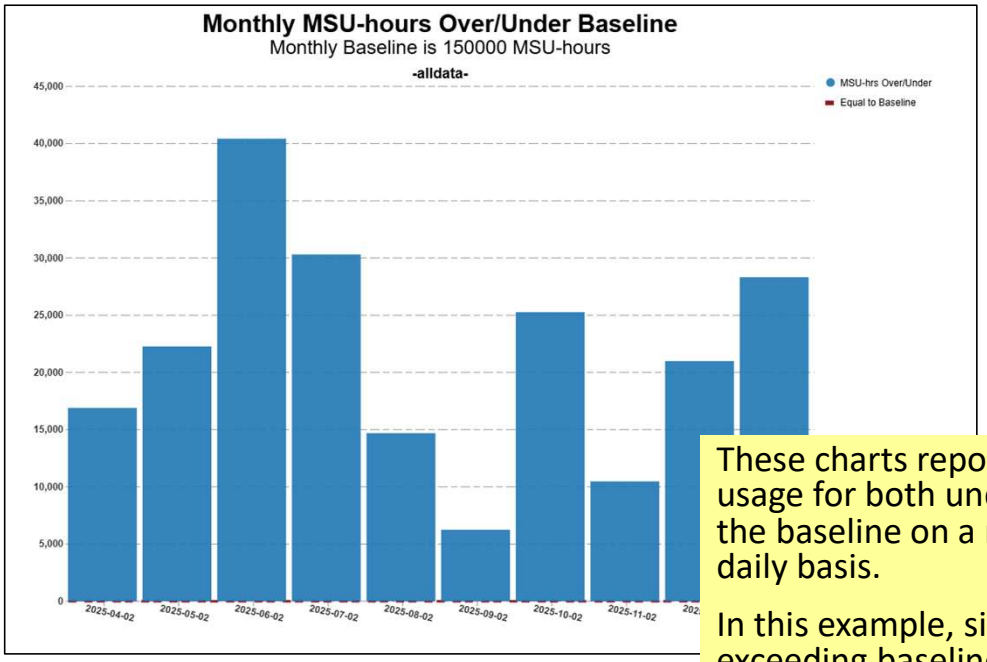
MSU Usage – TFP Analysis Example 1

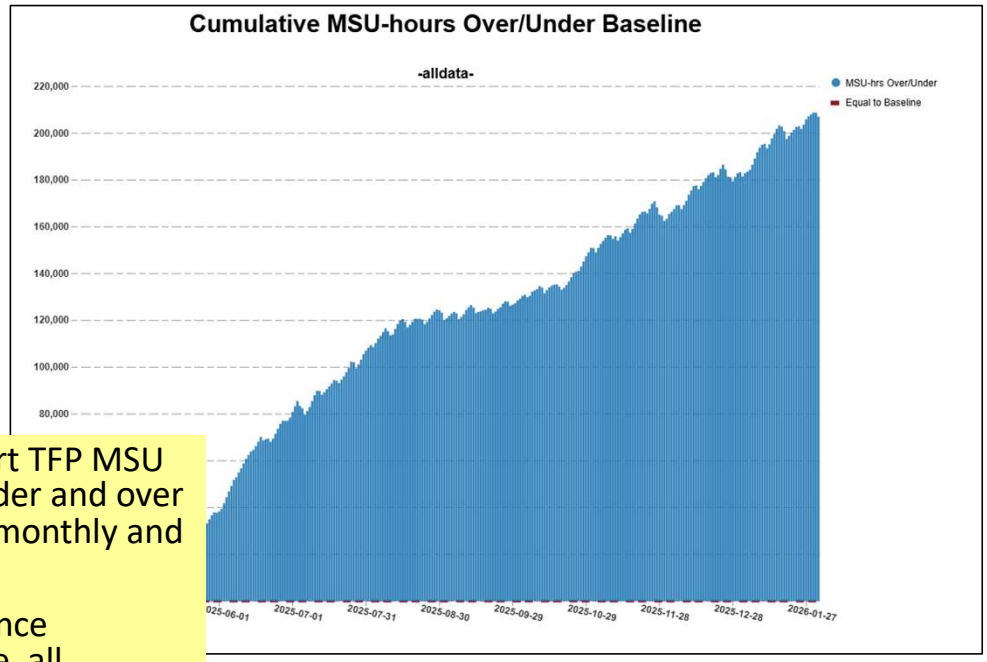




These charts report TFP MSU monthly and daily based on the contracted baselines. The baselines have been averaged for time period being reported.

MSU Usage – TFP Analysis Example 1

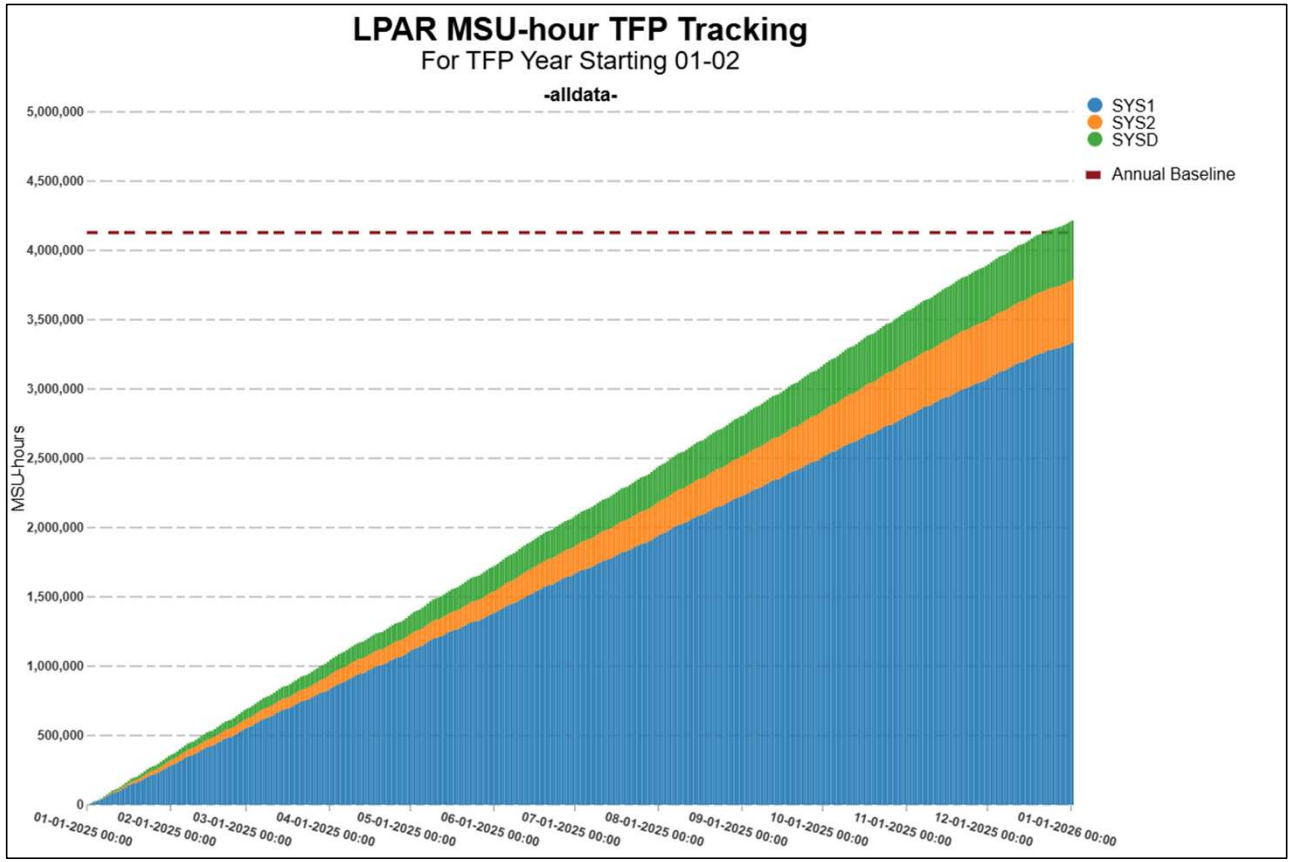




These charts report TFP MSU usage for both under and over the baseline on a monthly and daily basis.

In this example, since exceeding baseline, all numbers are 'over'

MSU Usage – TFP Analysis Example 2

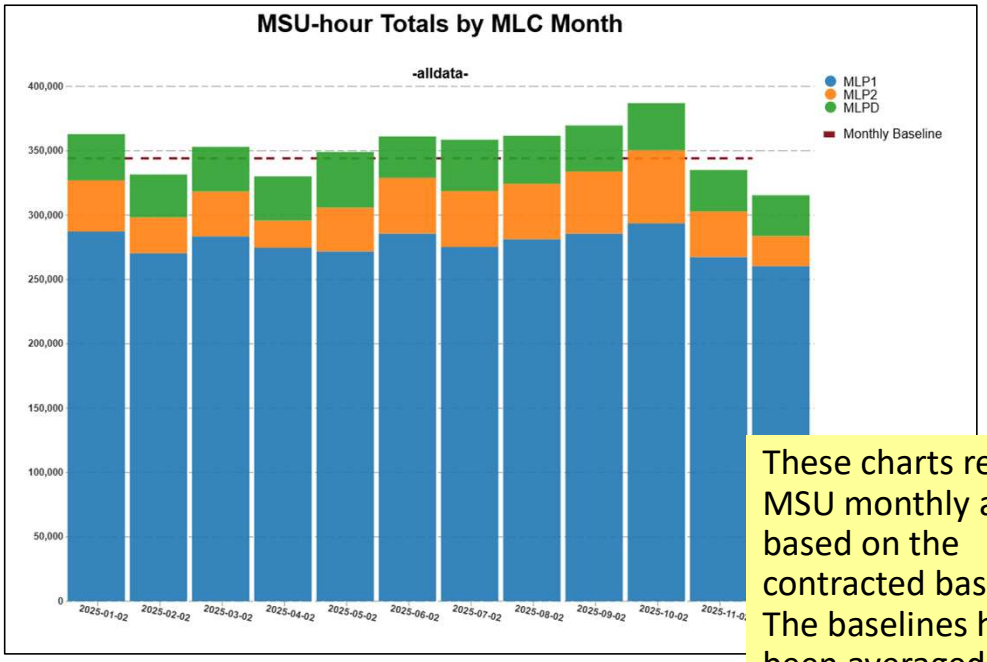


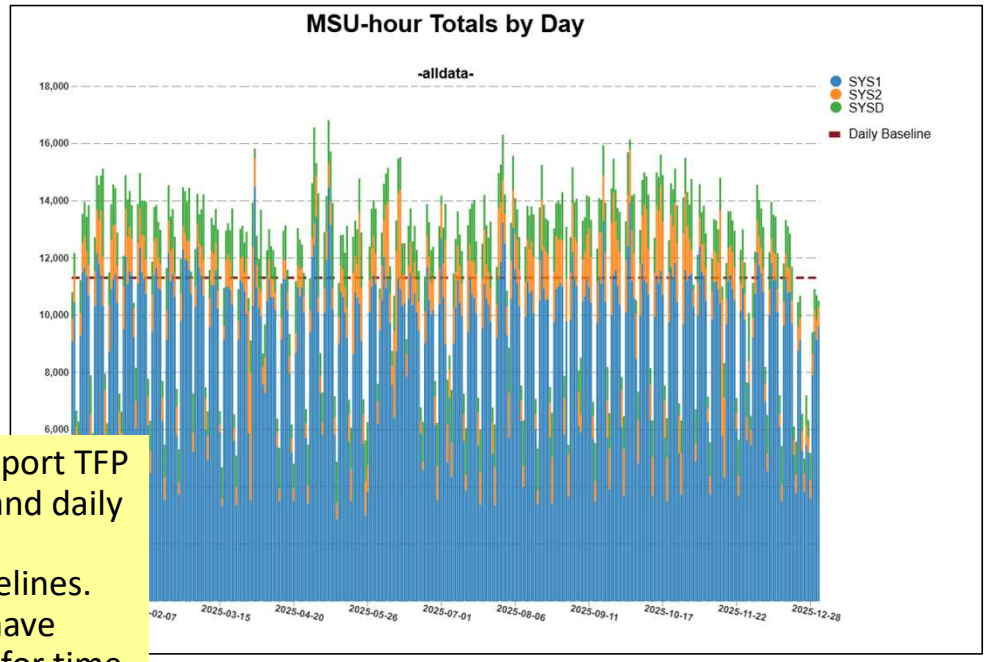
This chart reports the cumulative MSU consumption since anniversary date relative to the annual baseline agreed upon with IBM.

With TFP, all utilization contributes to the software cost.

Let's say SYSD is Dev/Test, Dev/Test solution should usually then be part of a TFP contract, so annual total will be really driven by production totals.

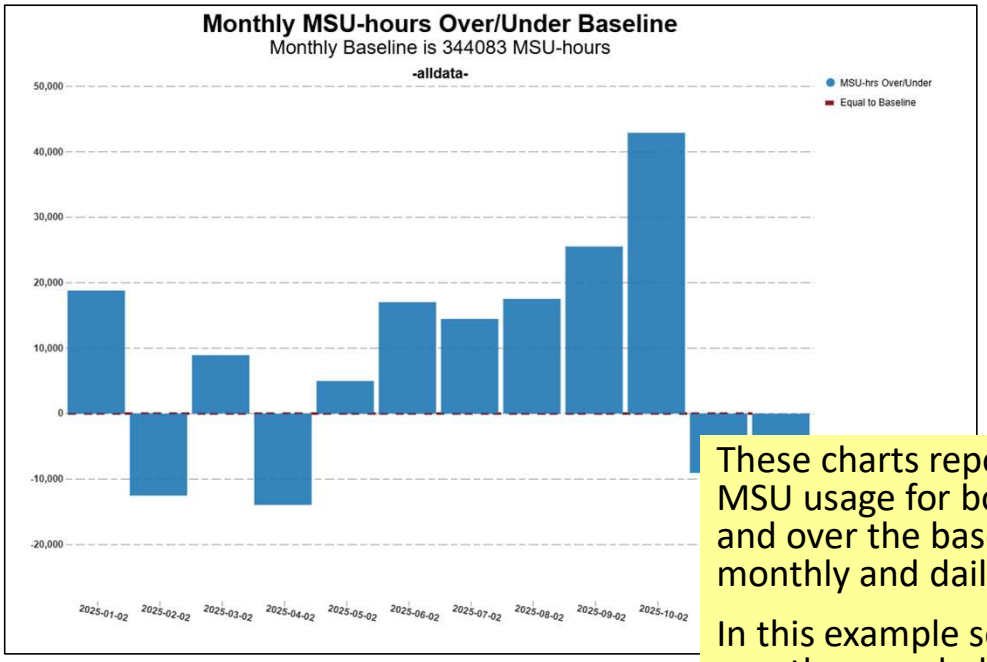
MSU Usage – TFP Analysis Example 2

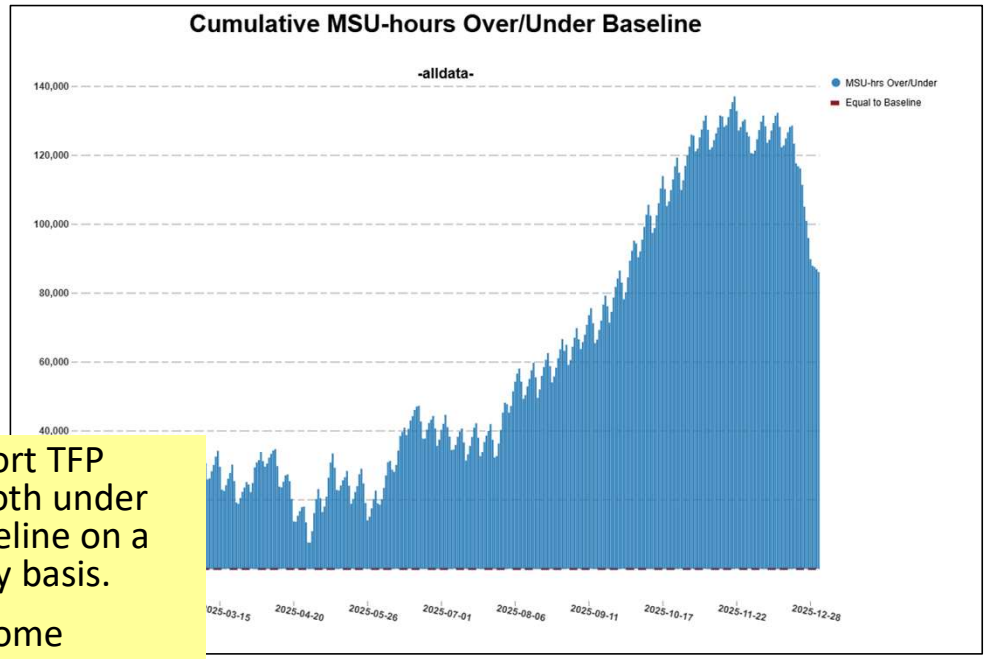




These charts report TFP MSU monthly and daily based on the contracted baselines. The baselines have been averaged for time period being reported.

MSU Usage – TFP Analysis Example 2





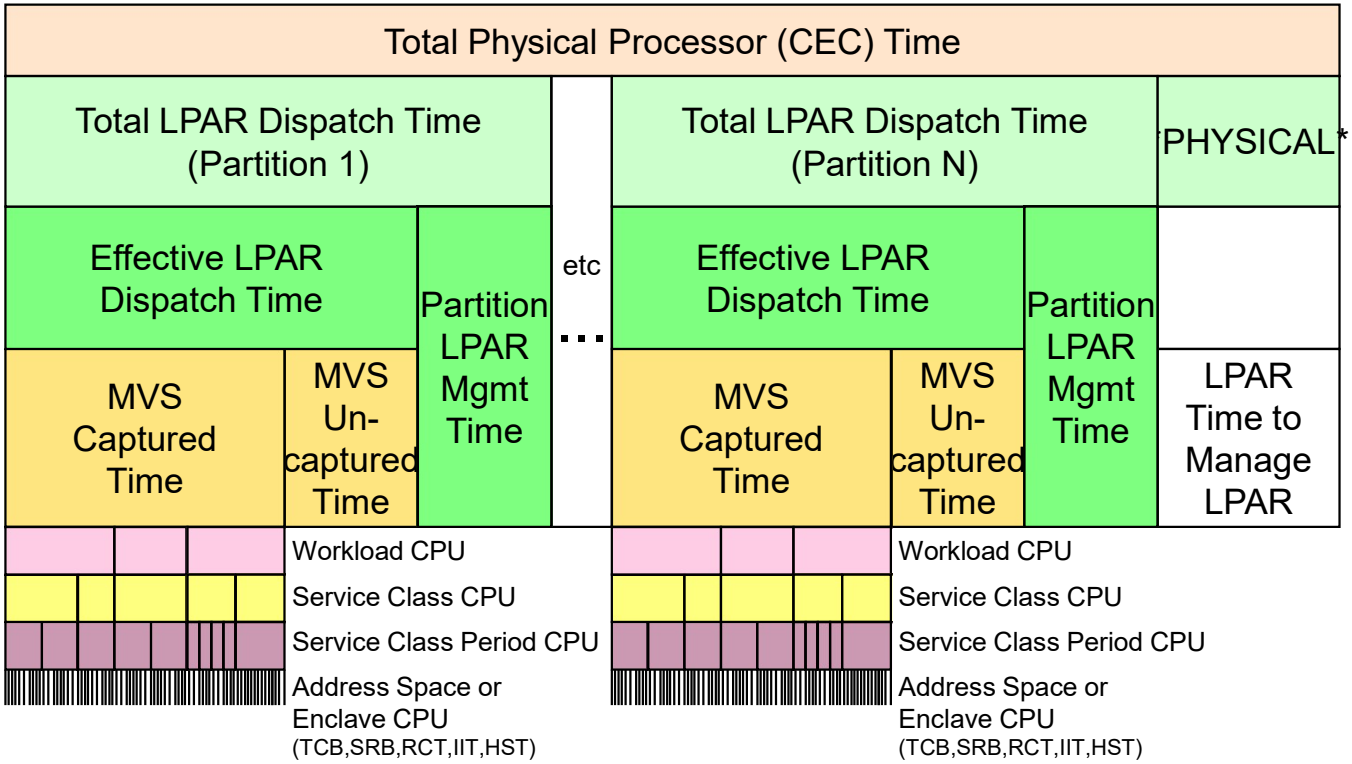
These charts report TFP MSU usage for both under and over the baseline on a monthly and daily basis.

In this example some months were below baseline.

Digging into MSU Usage

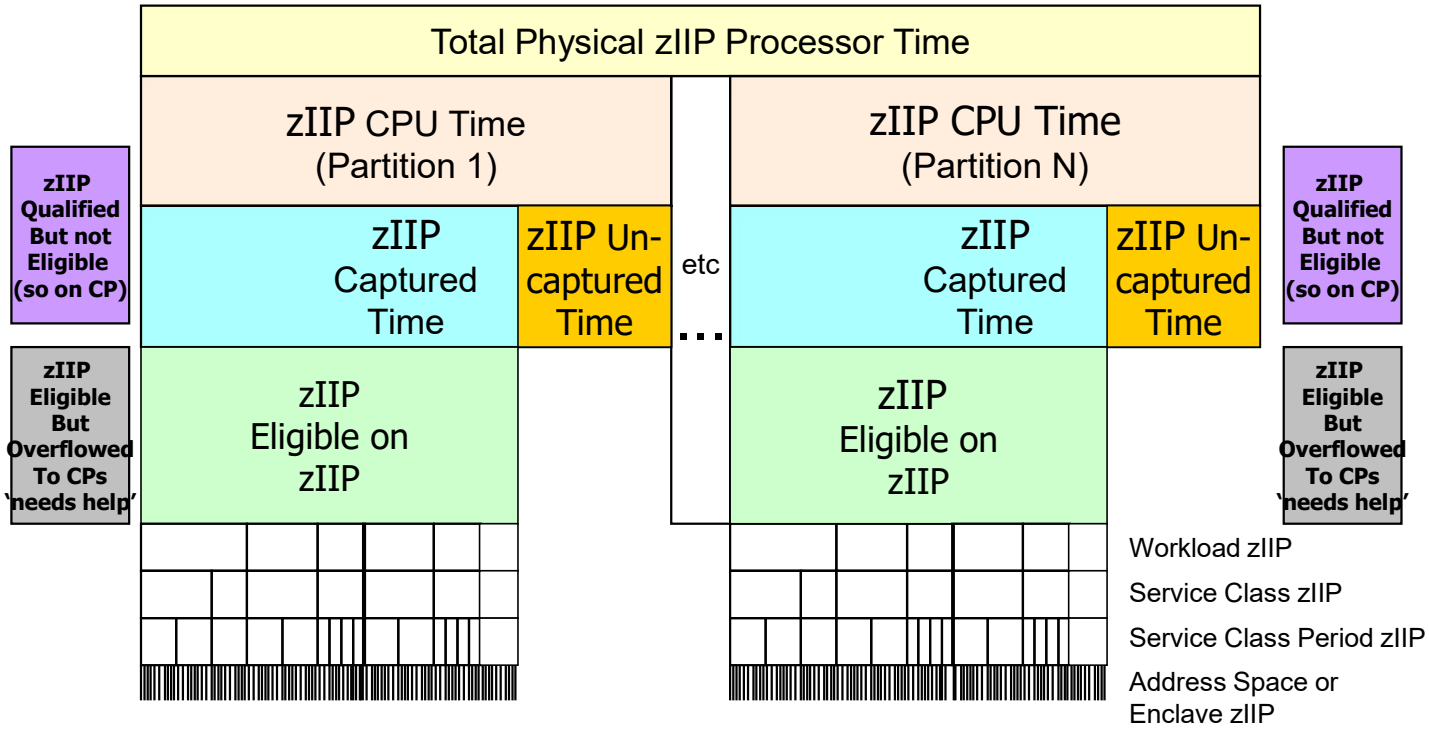
Breakdown of General-Purpose Processor

- We always needed to understand the break down of CP CPU consumption

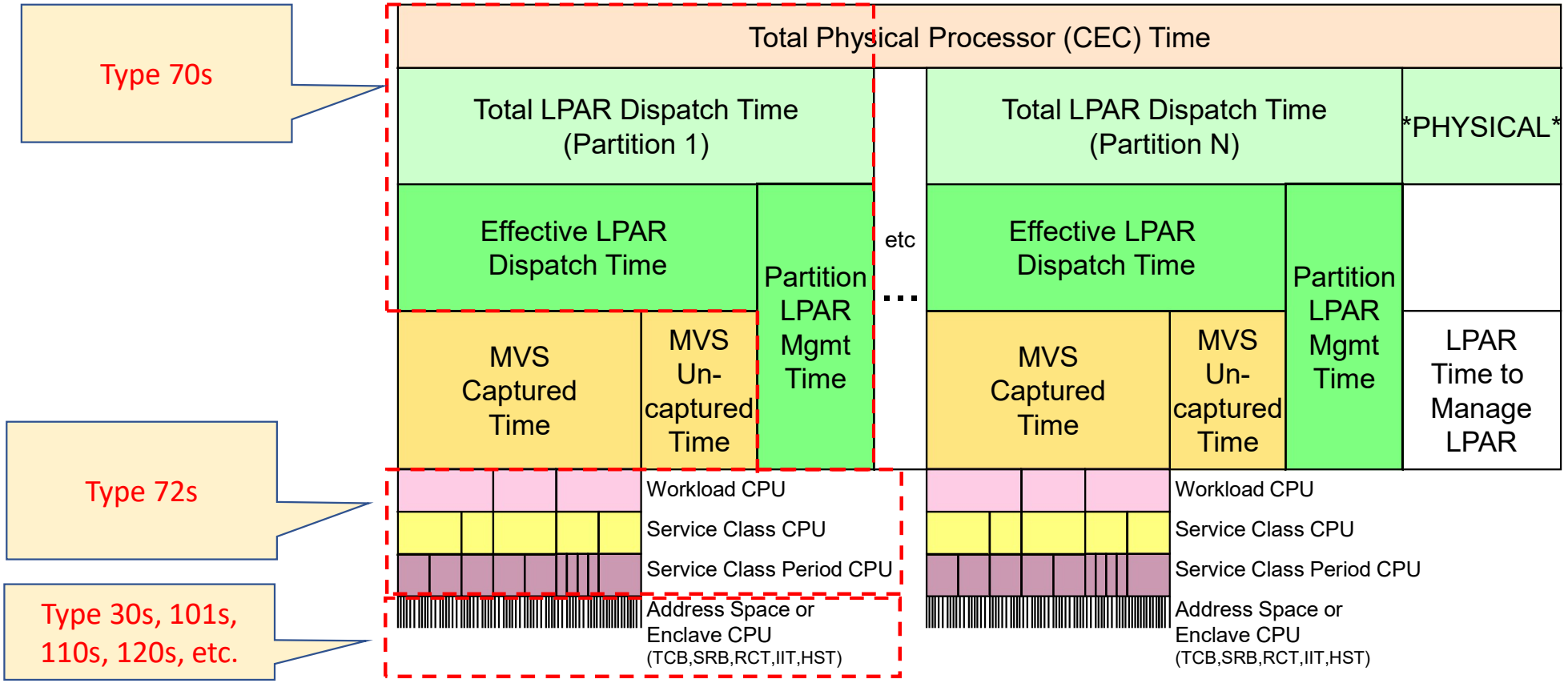


Breakdown of zIIP Engine Time

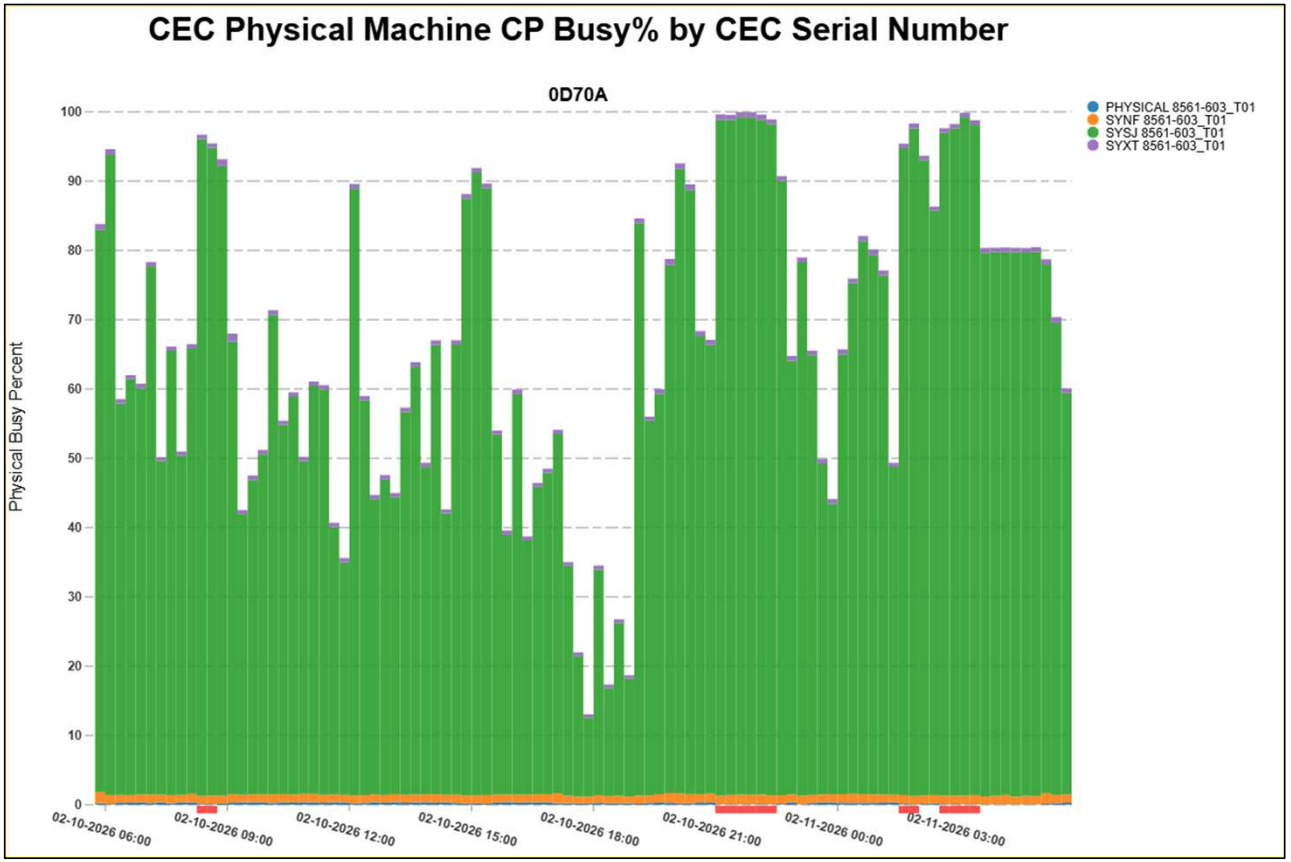
- Although not needed for an MSU analysis, we should also understand how PR/SM allocates the zIIP processor resource



Breaking Down CPU Consumption



Physical Process CPU Utilization



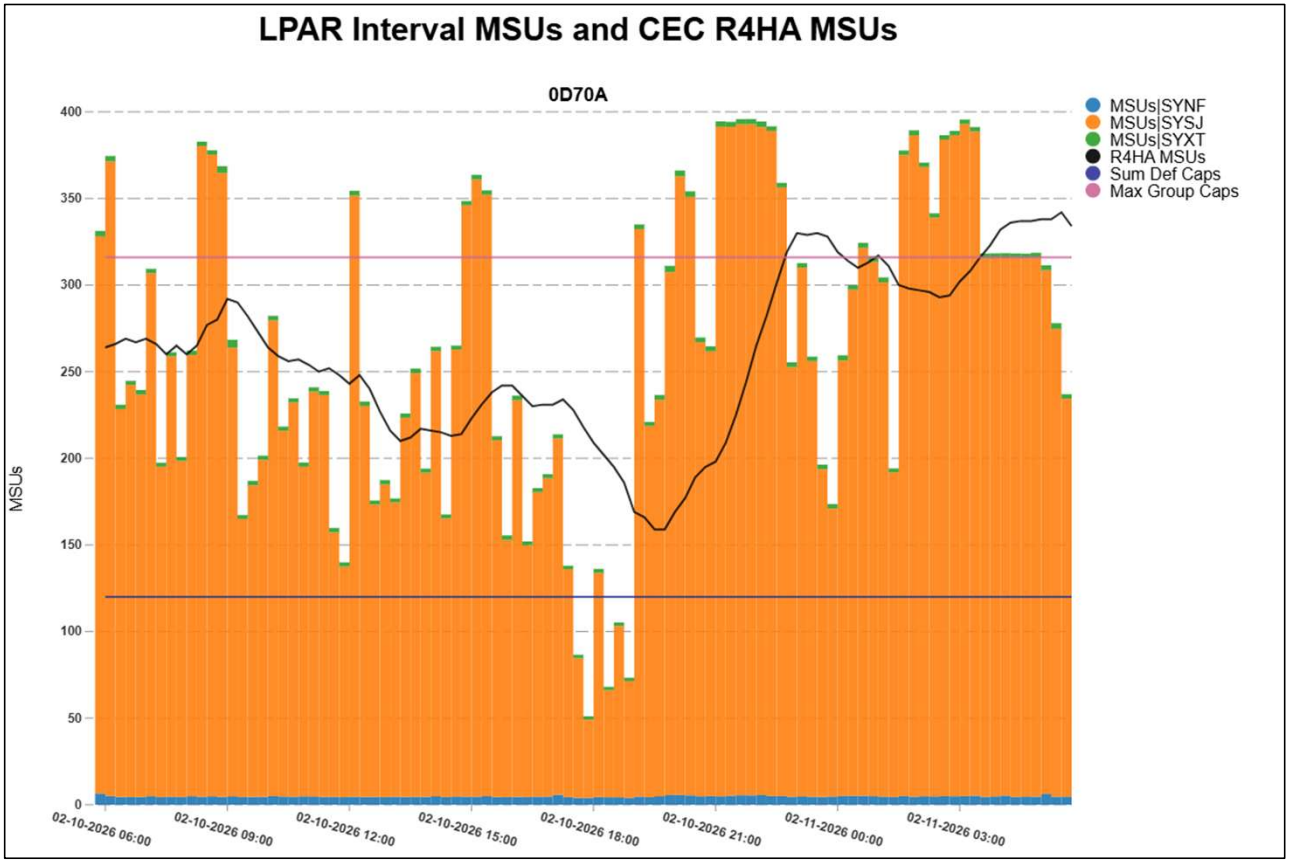
This chart just shows the utilization of a CEC for a single day. Although not many LPARs, it is still an interesting example.

Always take note of the CEC utilization pattern.

In this example, notice the flat lining of the CEC utilization. This usually indicates some sort of capping was in effect.

Also notice that this CEC regularly hits 100% CEC utilization.

Could measure CPU Consumption in MSUs



Another way of looking at CEC utilization is in terms of MSUs.

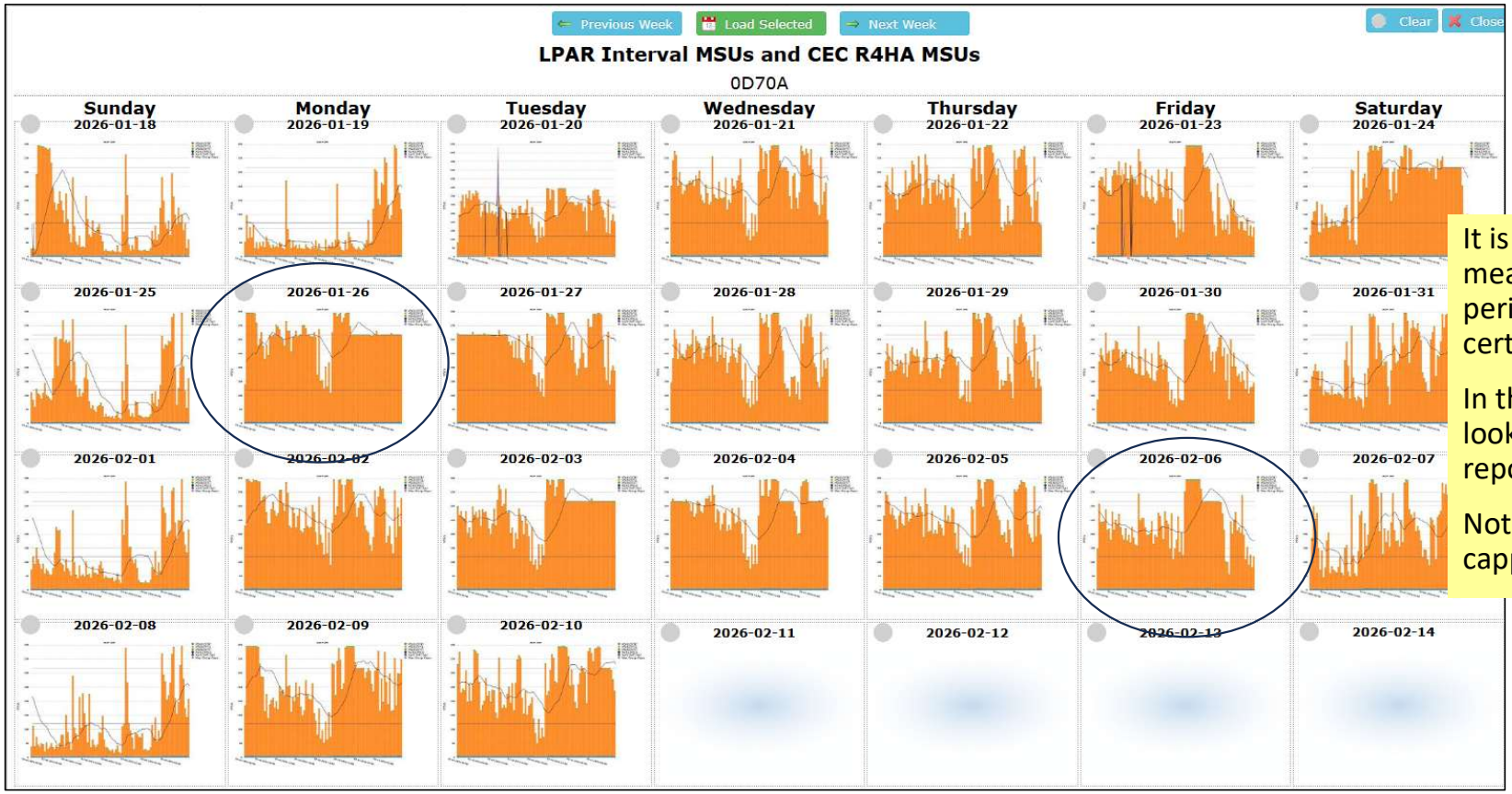
Best to look at MSUs consumed by each LPAR, but then also relate these values to MSU limits that may exist that would affect the MSU consumption.

In this example we see we have a group cap.

Also note the rolling 4-hour average pattern relative to the group cap value.

As a side note, we are not a fans of reporting things in terms of 'MIPS', but if you have a MIPS to MSU value, it can be applied here.

Could measure CPU Consumption in MSUs

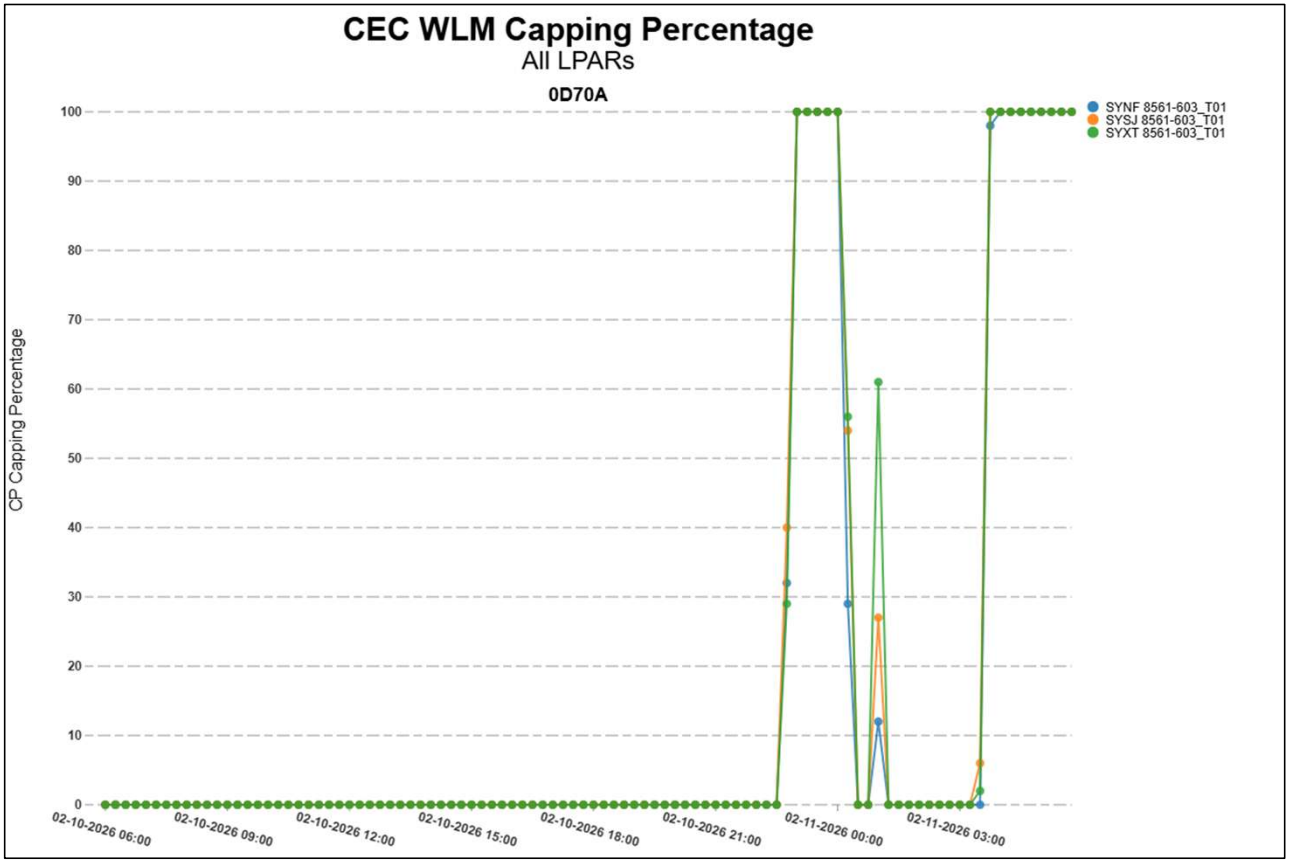


It is also nice to look at these measurements over a longer period to see if there are certain patterns.

In this example, we are looking at this MSU CEC report by calendar date.

Note the areas of supposed capping.

Understand if LPARs are being capped



Part of processor consumption analysis is also understanding if processor is being limited to an LPAR due to capping.

Look at capping patterns.

In this example, we see all LPARs being capped (due to the group caps we saw earlier).

Decomposing CPU Consumption
-
LPAR Level Analysis

- CPU Consumed by an LPAR
  - The LPAR utilization trinity (LPAR Busy%, Workload Busy%, and MVS Busy %
  - Capture ratios
  - Work Unit distribution to gain insights to latent demand
  - Host Effect CPU Consumption

z/OS CPU Times

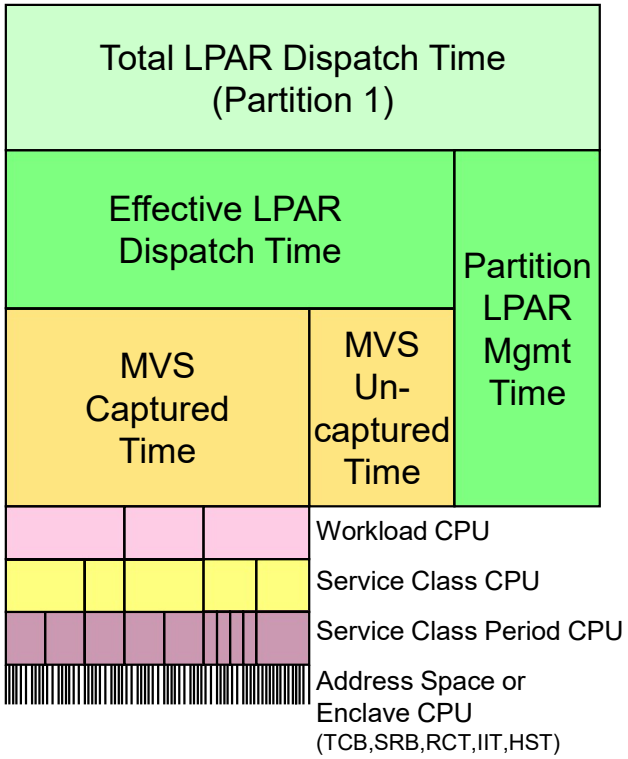
- Capture Ratios used to understand the stability and cost of system overhead

- Effective Dispatch Time
  - Time that the z/OS and the workloads were executing on the CPU

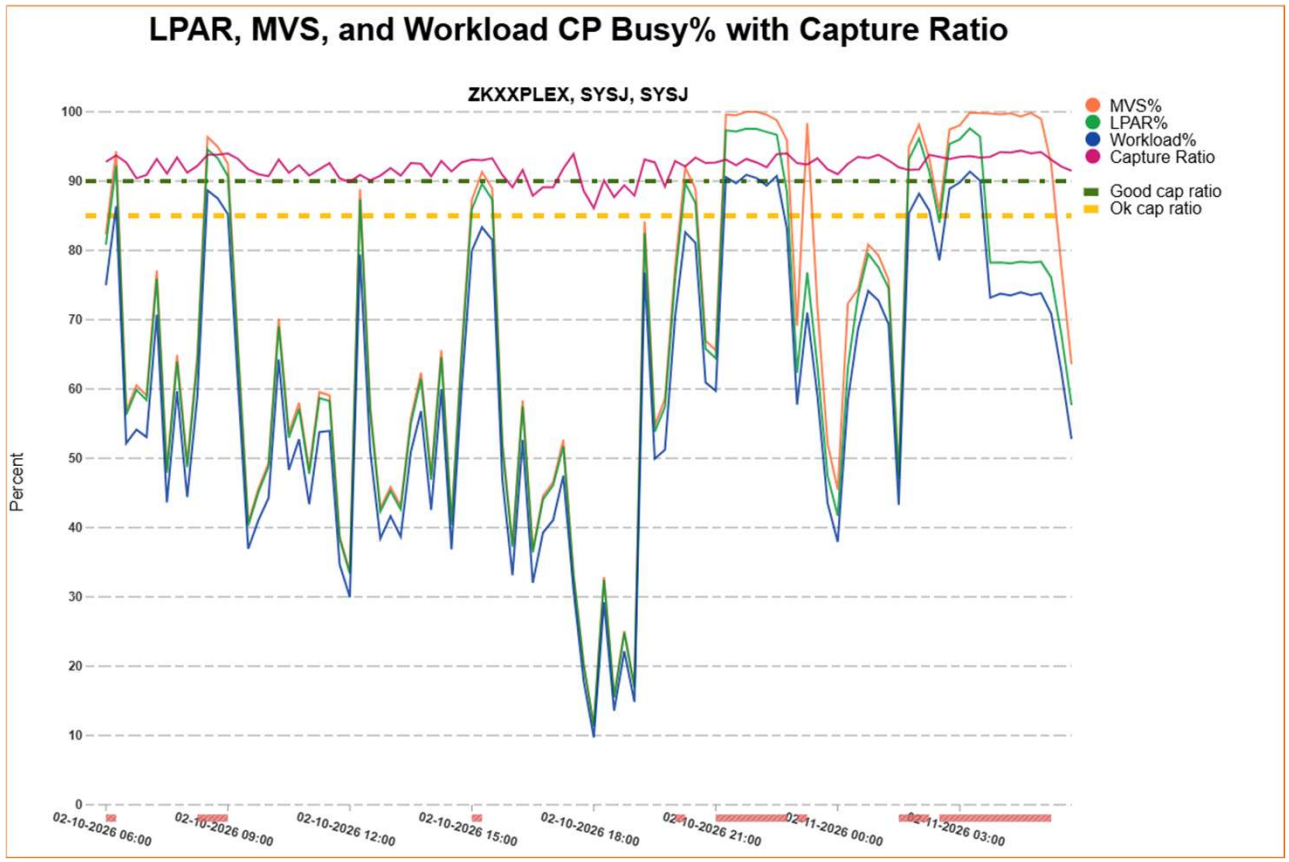
- MVS Capture Time
  - Time that can be accounted for towards specific workloads

- MVS Un-captured Time
  - System overhead
  - These MSUs are included in pricing

- Capture Ratio
  - Ratio of MVS Capture Time to Effective Dispatch Time



LPAR Busy%, Workload%, MVS%, and Capture Ratio



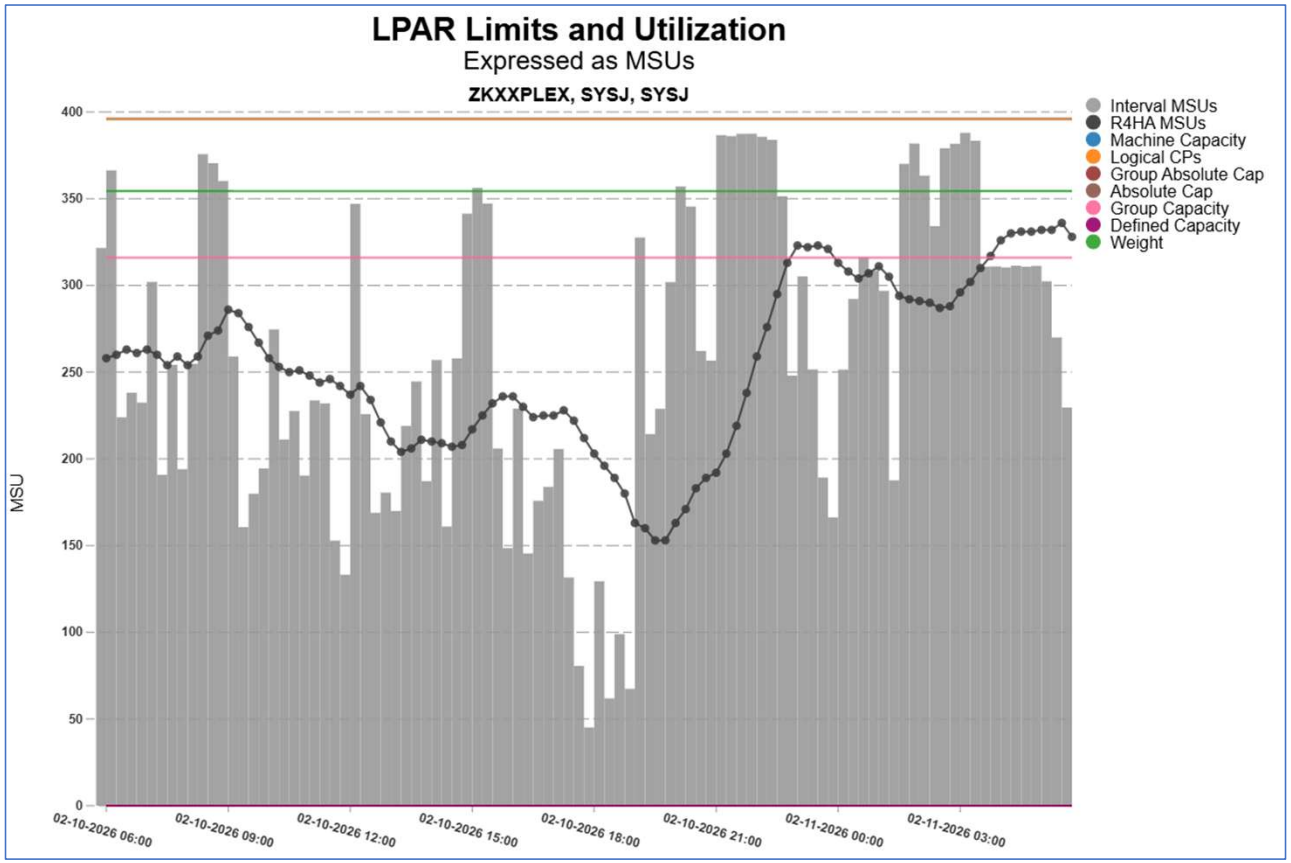
Understand the following key values for each of your LPARs:

- LPAR Busy % - How busy this LPAR is keeping its logical processors
- Workload Busy % - CPU utilization based on CPU we can account for towards particular workloads
- Delta between LPAR% and Workload% is the Uncaptured%
- Capture Ratio – What percentage Workload% is of LPAR%
- MVS Busy % is what utilization relative to MVS Waits.
  Basically, what MVS wanted.
  Delta between LPAR Busy %
  and MVS Busy % is usually due to capping or weight enforcement

Uncaptured Time consumes CPU which translates to MSUs

- Many causes for uncaptured time. Common causes are as follows:
  - High page fault rates
  - First reference page faults
  - Full preemption
  - Suspense lock contention
  - Spin lock contention
  - Getmain/Freemain activity (recommend cell pools)
  - SRM time-slice processing
  - Interrupts
  - SLIP processing
  - Long queues being processed in uncaptured processing
  - Affinity processing (such as need for a specific CPU or crypto facility)

LPAR Limits to control R4HA costs



Controlling MSUs also means you need to have insights to all the MSU limits an LPAR and it's workloads must adhere to.

In this example, we see that SYSJ has limits that bound its MSU consumption.

- MSU machine capacity
- MSU Logical CP capacity
- MSU of LPAR weight
- MSU Group Capacity

Note: Defined Capacity (DC) Limits and Group Capacity (GC) Limits are two methods to control your R4HA costs

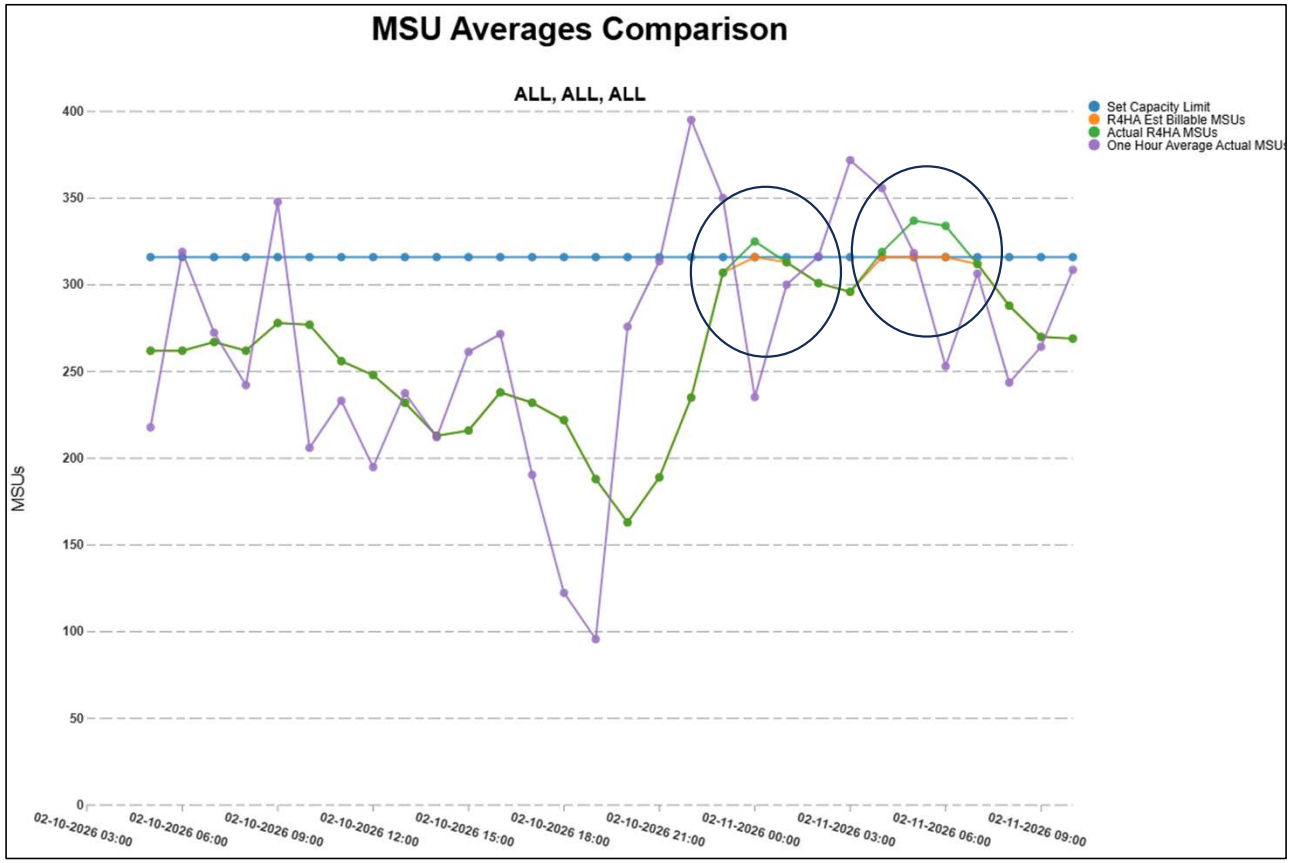
- AKA "Soft capping"
- Provided by IBM, something you can do for no additional charge
- Separate ISV software exists to manage these dynamically

DC is for individual LPARs

GC is for groups of LPARs on the same CEC

- Do not have to be in the same sysplex
- LPARs in a capacity group can also have DC limits

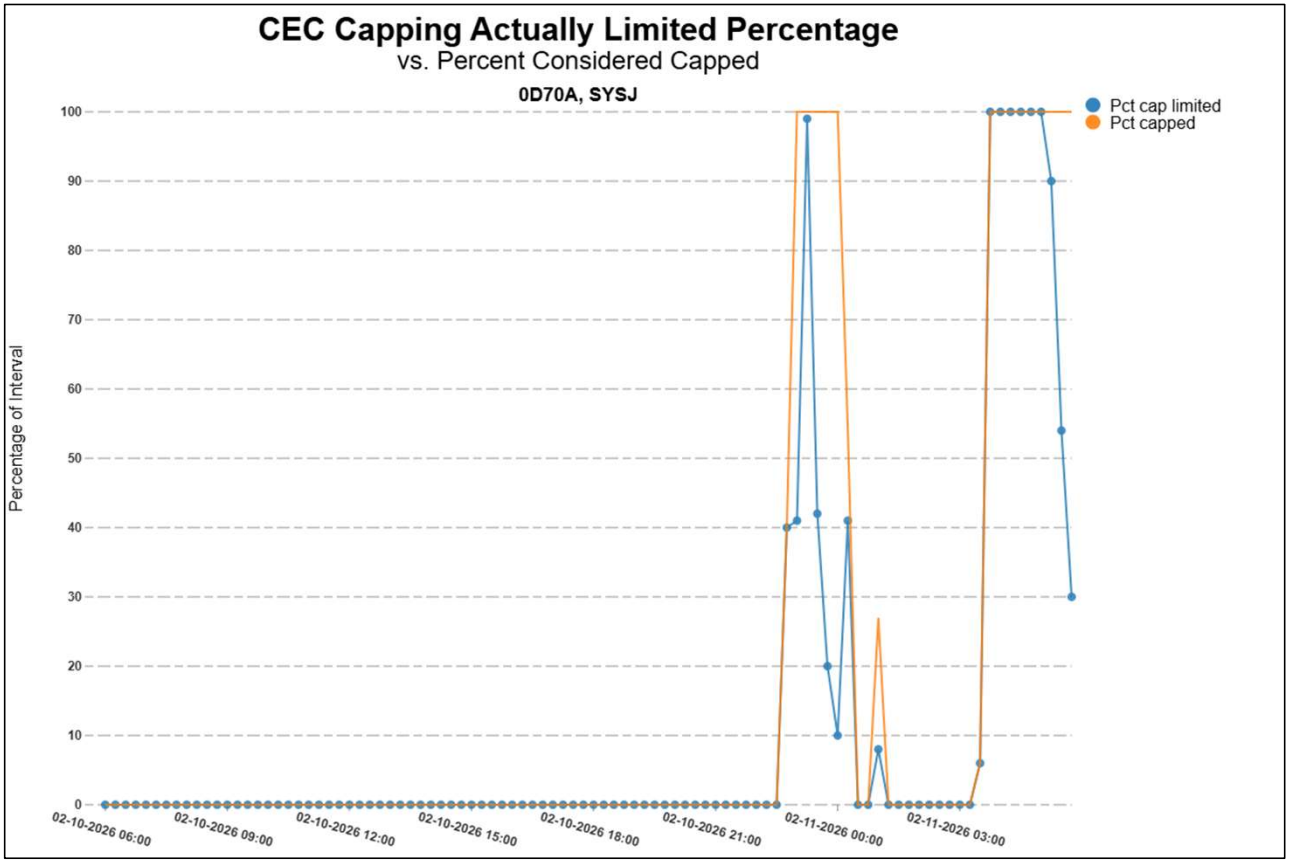
Rolling 4-hour average



The Rolling 4-Hour Average(R4HA) of an LPAR could exceed the capacity limit, but the billable MSUs cannot.

When a workload wants to consume above the capacity limit when the R4HA is exceeded, this is when capping occurs.

Understanding if capping is limiting an LPAR



An LPAR could be capped, but if the LPAR does not have demand for CPU during the capping period, we probably care less about the capping.

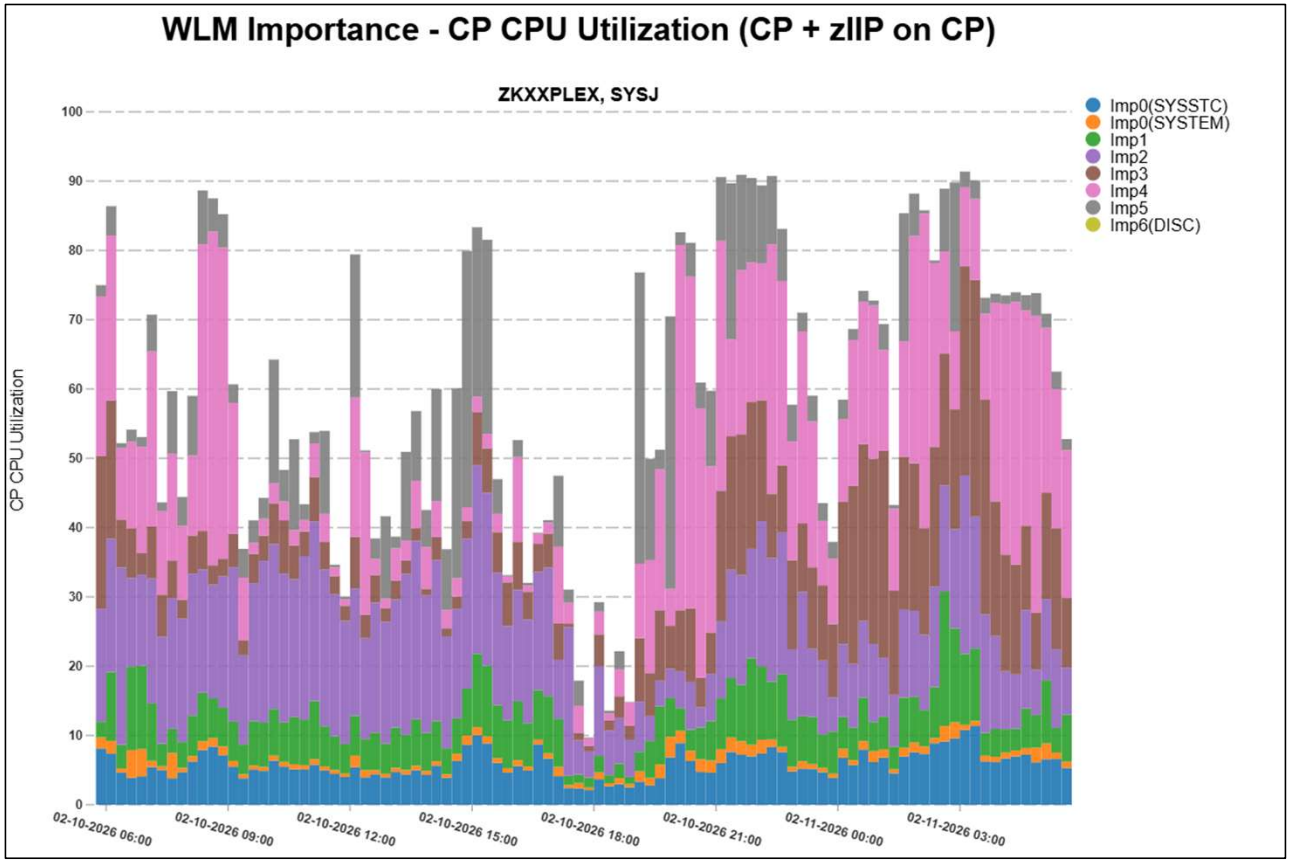
So, when looking at capping, make sure you also determine if the LPAR is being limited by the Cap.

In this example, we see the LPAR is being limited by the capping most of the capping periods of time.

Decomposing CPU Consumption
-
WLM Workload Level Analysis

- CPU Consumption at the importance level
- CPU Consumption at the WLM Service Class and Service Class Period Level
- Commentary about Report Classes
- Other CPU consumption measurements
  - –CPU consumed at promotion
  - –Did lower importance work not consume CPU due to lack of demand or due to lack of CPU?

Workload Utilization by Importance Level

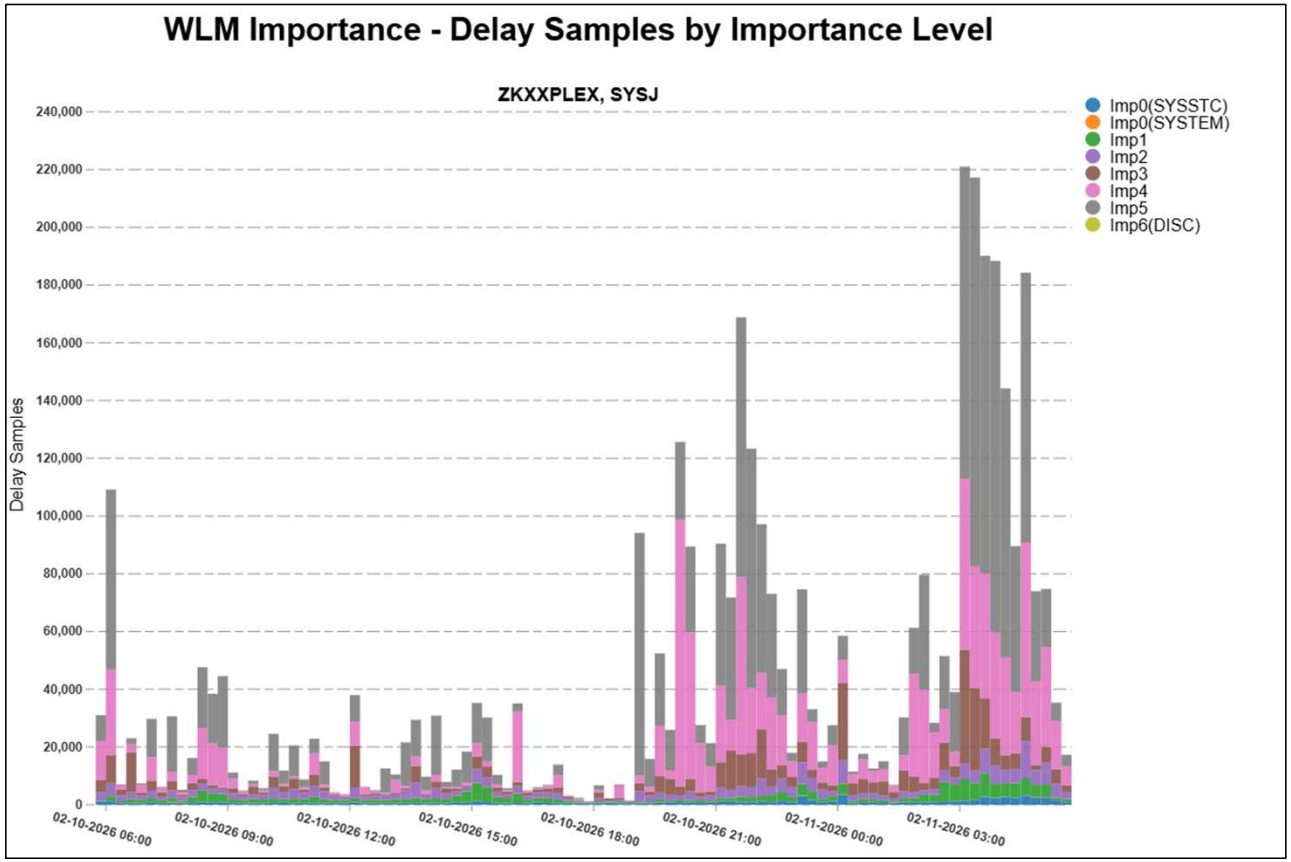


When examining CPU consumption, it is interesting to gain insights into how much CPU is being consumed at each WLM importance level. Your WLM importance levels should be designed so that higher importance work has a lower importance pool of CPU to 'steal from'.

In this example, notice that little CPU is being used by low importance work.

- Workload utilization is calculated as (sum Service Class CPU Time) / (#Logical CPs * Interval Time)
- Is it because there is little to no low importance work?
- Or is it because low importance work cannot run due to lack of capacity?

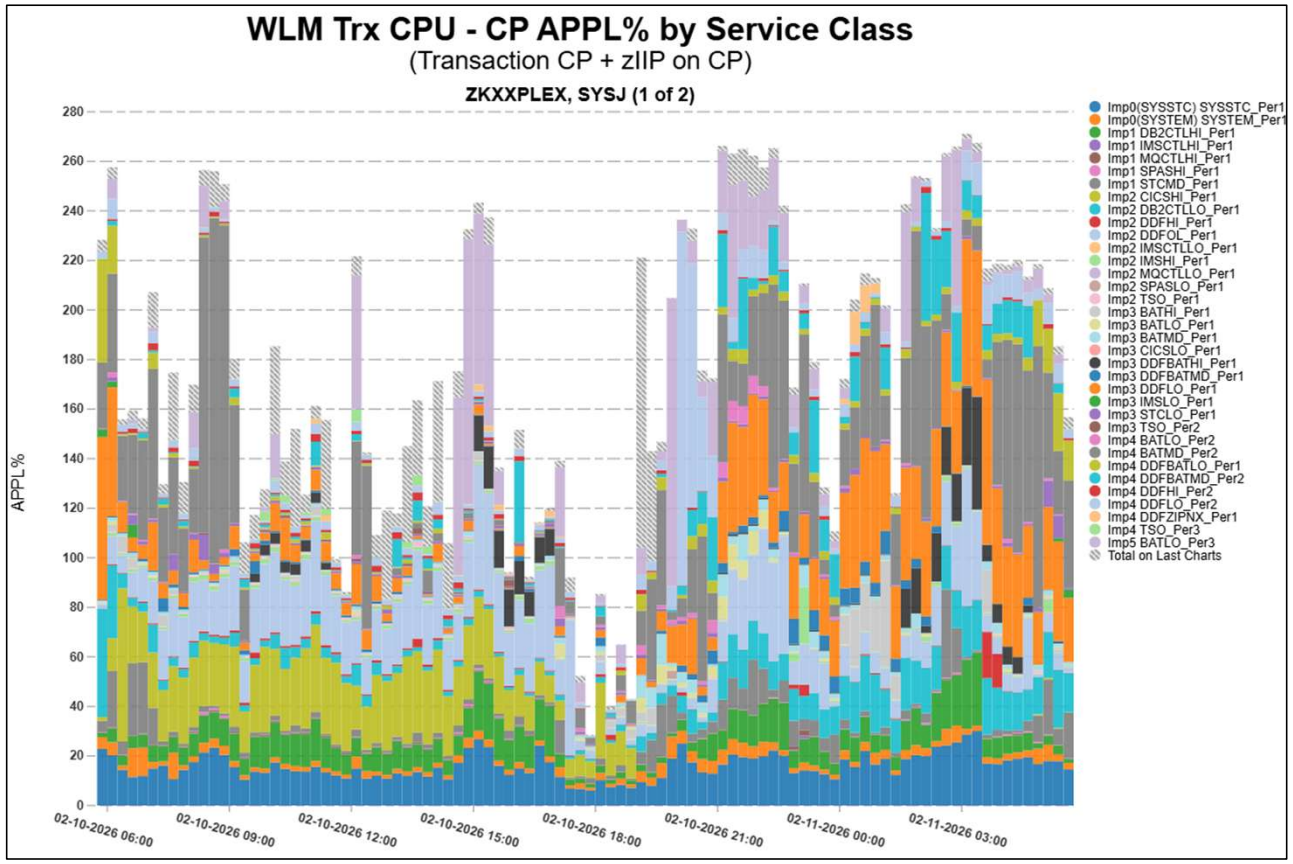
Delay Samples by Importance Level



Examine delay samples by importance level. This will help us to determine if lower importance work consuming little CPU is due to lack of demand or because higher importance work does not leave enough for the lower importance work to run.

In this and in the pervious example we saw that discretionary work used little CPU and have very few delay samples. This leads us to believe there is not much demand for CPU by discretionary work.

Workload APPL% by Service Class Period

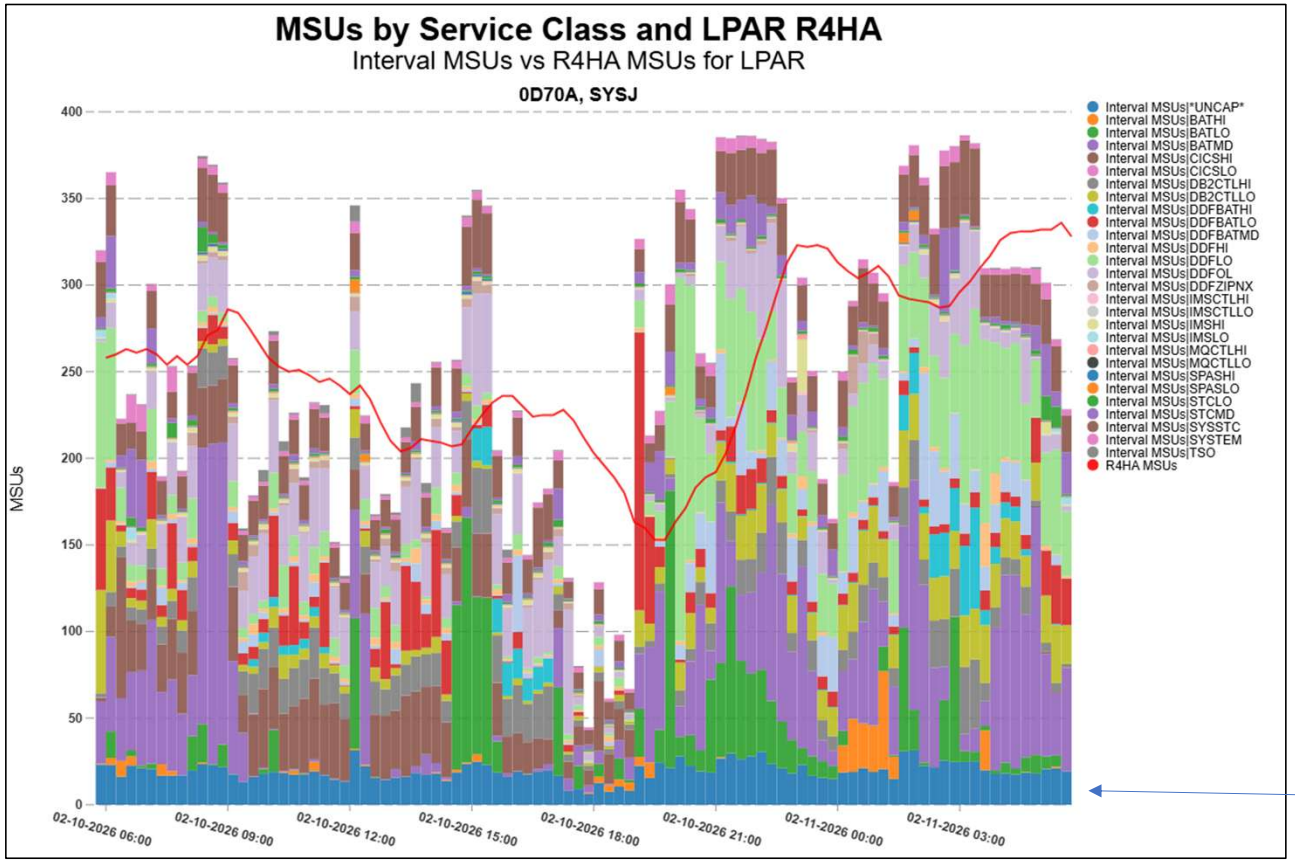


Continue your decomposition of processor consumption by determine which workloads are using the CPU. We can do this by looking at CPU consumed by WLM service class periods.

Who are the consumers and at which importance levels.

This charge shows processor consumption as a percentage of a single CPU. Thus, 100% means one CP of capacity, and 200% means 2 CPs of capacity, etc.

Workload Utilization by Service Class Period



Continue your decomposition of processor consumption by determine which workloads are using the MSUs.

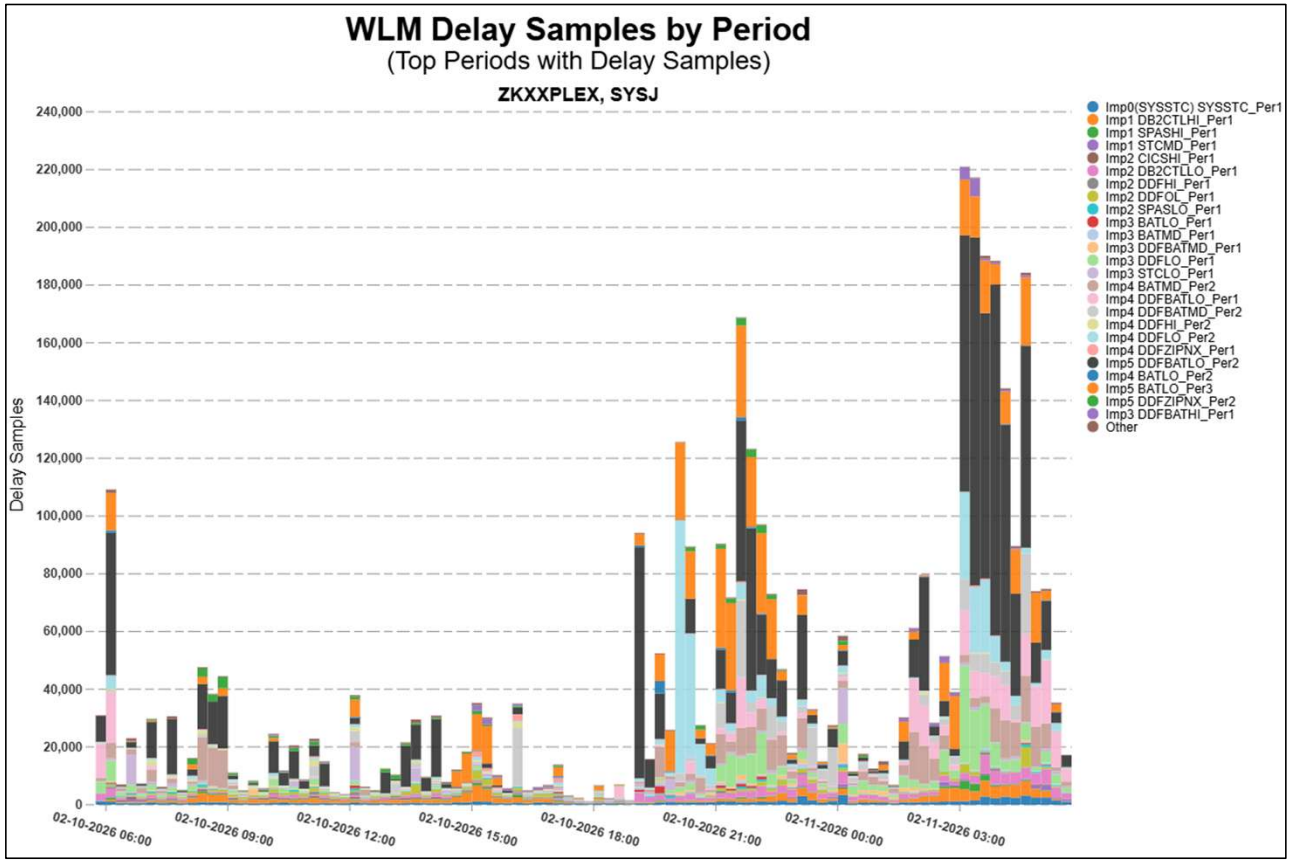
Who are the consumers and at which importance levels.

This chart shows processor consumption in terms of MSUs

Note un-captured MSUs at the bottom

Note MSUs consumed due to Uncaptured

Delay Samples by Service Class Period

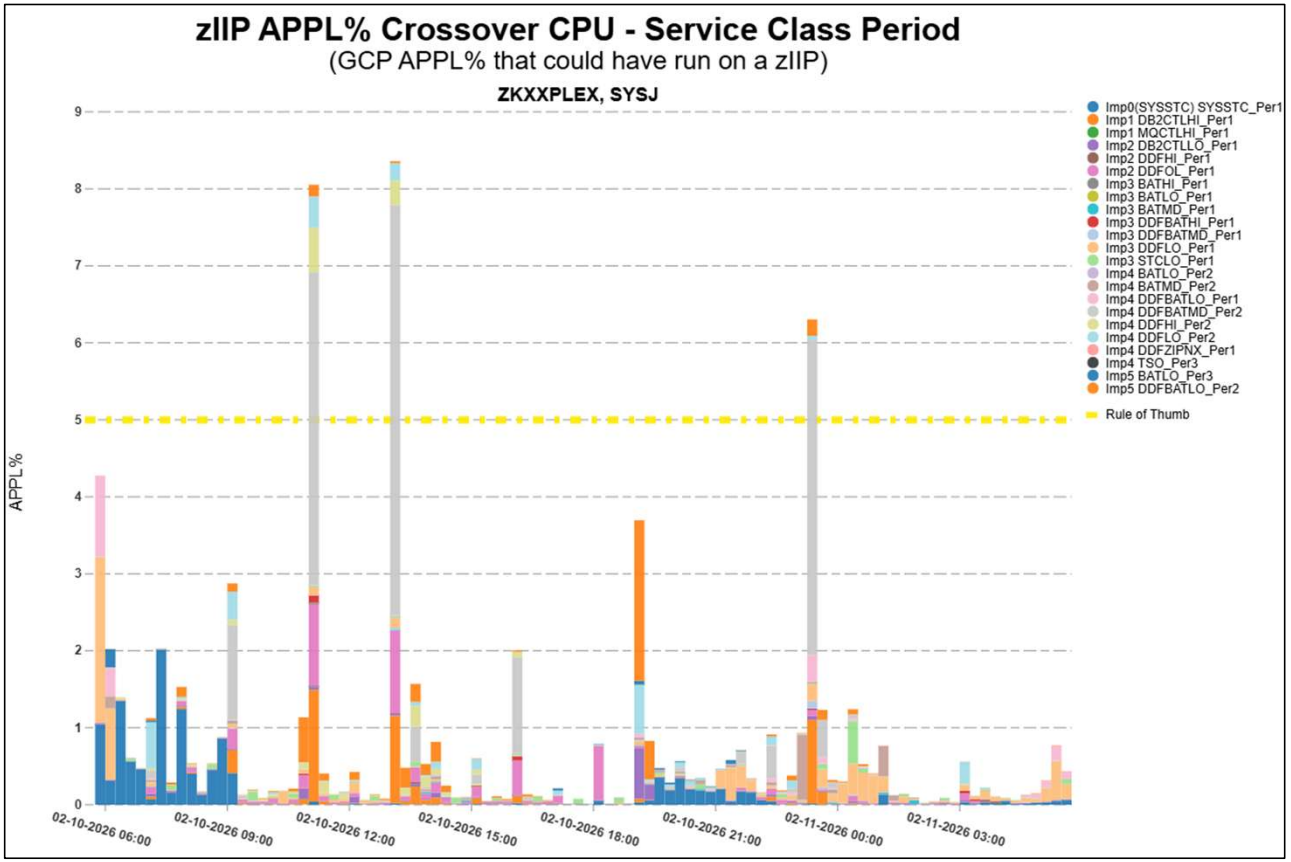


As we did with CPU analysis by importance level, we want to look at WLM delay samples by service class period. This way we gain insights into which workloads are suffering due to lack of CPU.

Related to MSUs

- If not capping, then delays are due to some other constraint
- If not capping, then maybe lowering a capacity constraint will lower something like the R4HA. So make sure lower importance workloads will suffer
- If capping, then examine delays to make sure it is the lower importance workloads suffering the most.

zIIP Cross Over Appl% - zIIP work on CP engines



Also make sure you understand the CPU cross over from zIIP engines to CP engines.

- Are you crossing over during peak periods of the month and possibly affecting your monthly software bill?

Why is crossover so important??

- MLC costs are based on GCP utilization (either peak R4HA or total)
- If crossover is significant during your peak R4HA (or you're on TFP), your MLC costs are higher than they would be if you had more zIIP capacity
  - All usual MLC savings caveats apply:
    - If using R4HA, how many peaks do you have?
    - If using R4HA, how large is the peak vs. next highest interval?
    - If using TFP, are you tracking to above your baseline?
    - Where are you on the MLC price curve (10% R4HA reduction means <10% savings)
- If your GCPs are slower than your zIIPS, zIIP eligible work is not performing as well as if it was on the faster zIIPs
  - Although possibly better than if waited for the overloaded zIIP to become available

Avoid significant crossover

Addressing Crossover

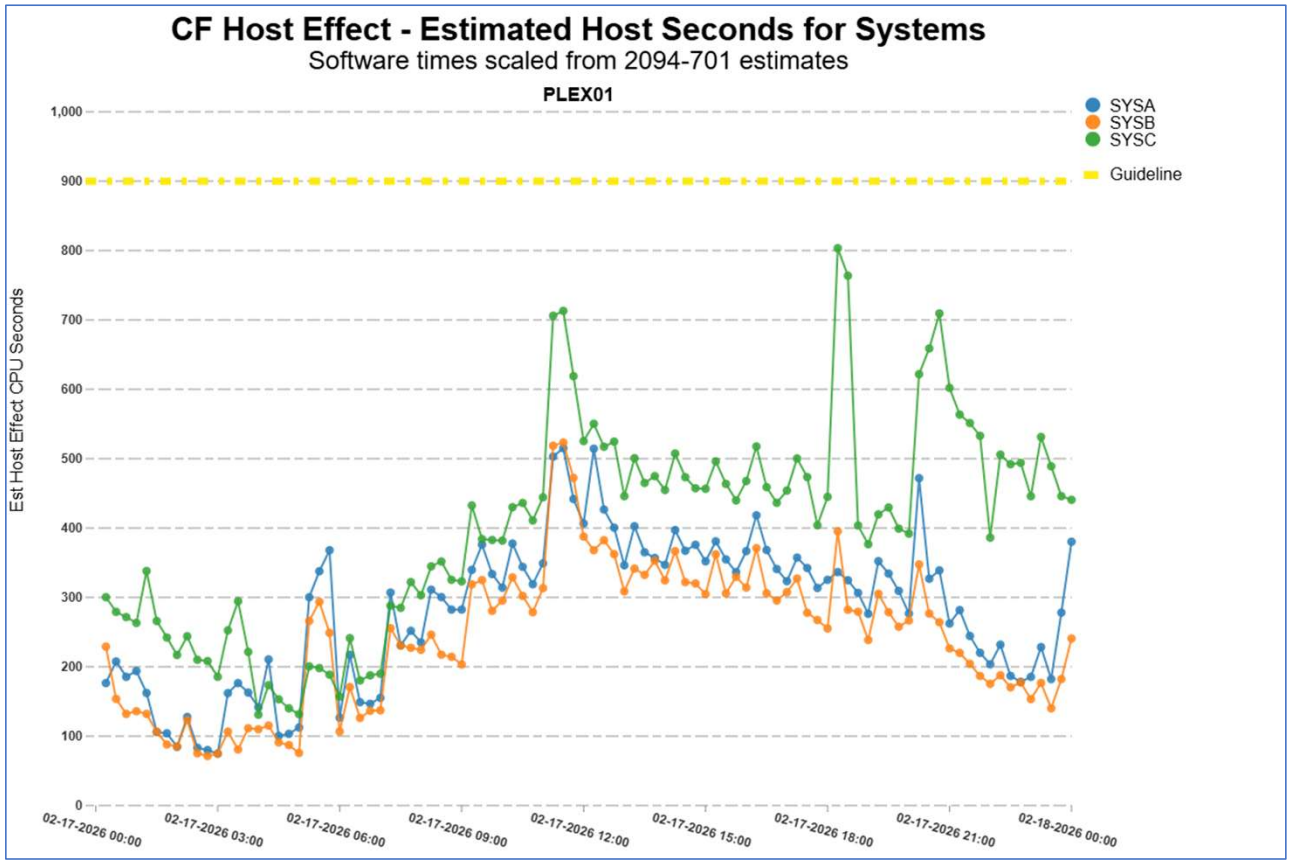
- IIPHONORPRIORITY in IEAOPT (YES or NO)
  - Set to "NO" to disallow most crossover
  - Note that there still will likely be a little bit of crossover
    - Contention resolution and certain interrupt processing will cause some crossover
  - **Not recommended today**
    - In fact, can cause DB2 to stop sending some system tasks to zIIP!
- ZIIPAWMT in IEAOPT (1 to 499999, 3200 default with HD)
  - Increasing the value can result in less work crossing over to the GCPs
  - Work may be delayed more while it waits longer for a zIIP/GCP
  - Can be useful to increase in certain situations, but should be used carefully
- HONORPRIORITY on Service Class
  - Stop crossover on a service class basis
  - In some cases, may consider stopping crossover on all but system-related work
  - Recommended solution today (for most cases)
- Also: Buy more zIIPs (or, maybe, enable SMT)

Decomposing CPU Consumption
-
Transactional Analysis

MSU Usage by Transactions

- Eventually, MSU usage boils down to two main areas:
  - MSU usage on behalf of transactional workloads
  - Examples include:
    - Coupling Facility Sync Spin times
    - Processor cache usage
    - Monitoring
    - Locking
    - Etc..
  - MSU usage by transaction's applications
    - Batch, IMS, CICS, DB2, MQ, WAS, etc.

Coupling Facility Host CPU Times



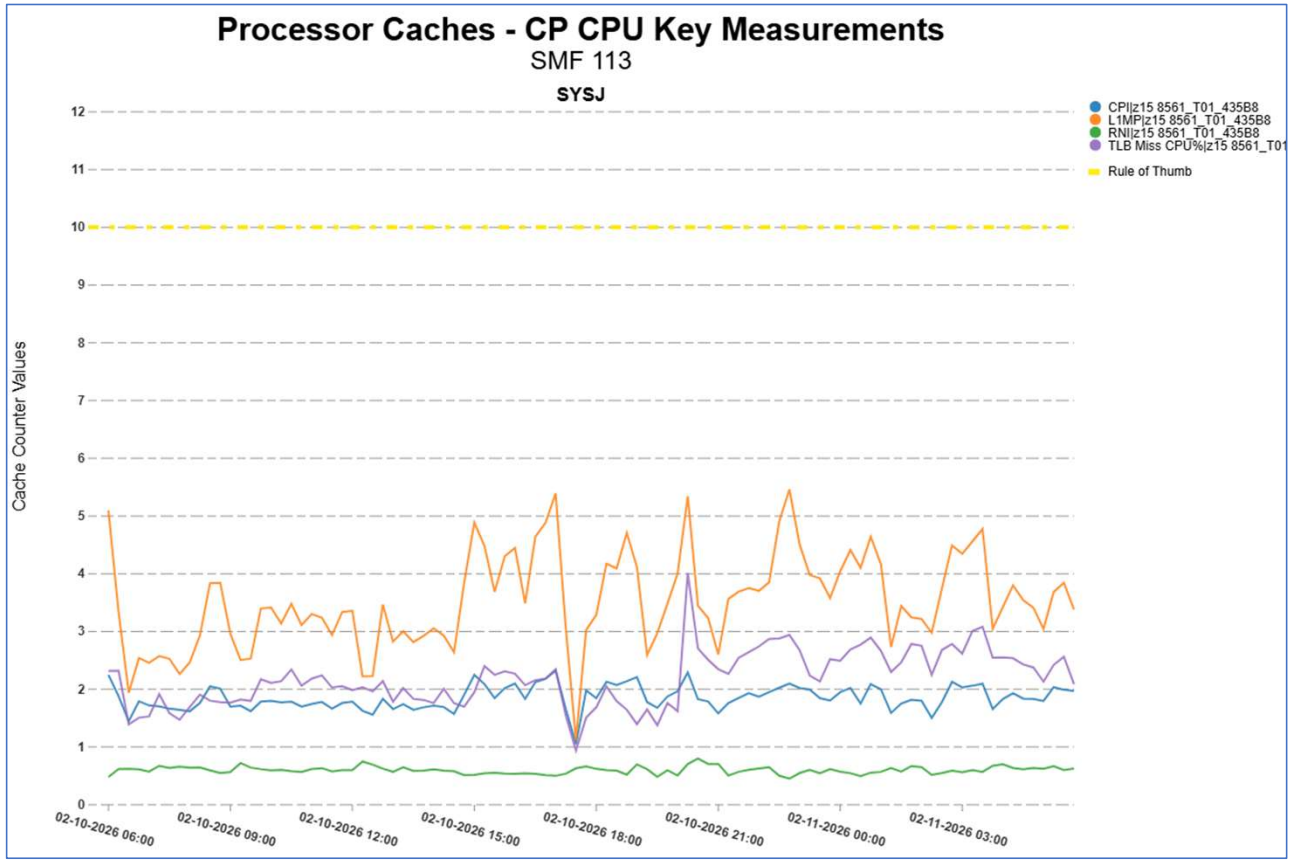
If you are a in a sysplex, remember that usage of the CF by z/OS workloads and system use CPU time on z/OS.

Naturally, these times translate to MSUs

So, understand CPU usage due to usage of the CF, and tune workloads and structures.

Hint: Lock structures typically make up the largest cost.

Usage of Processor Caches influence MSUs

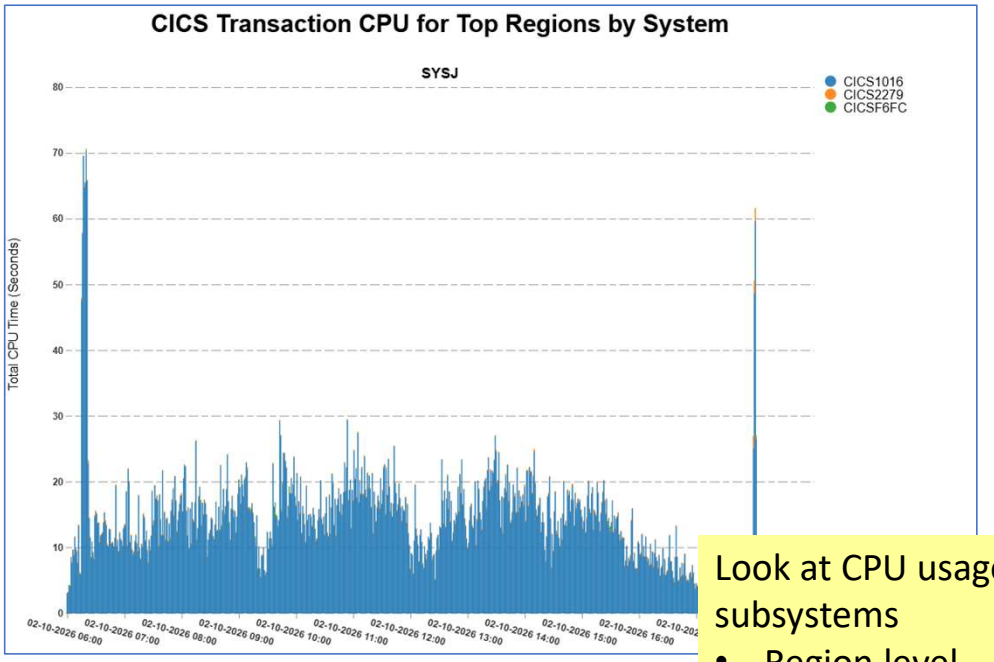


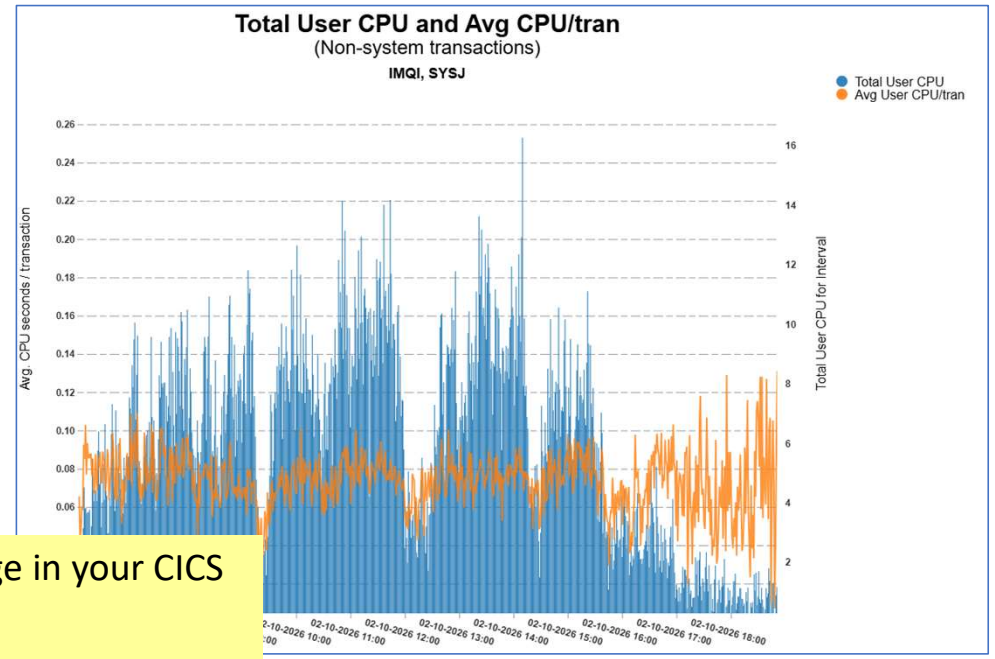
You should not be tuning your workloads attempting to control the processor caches, but remember that certain configuration options result in higher CPU usage on the system.

Examples:

- TLB Miss CPU%
  - Use large frames
- Higher RNIs
  - Control HiperDisptach High, Med, Low pooling
  - Multiple draw usage

CICS Transaction CPU Time

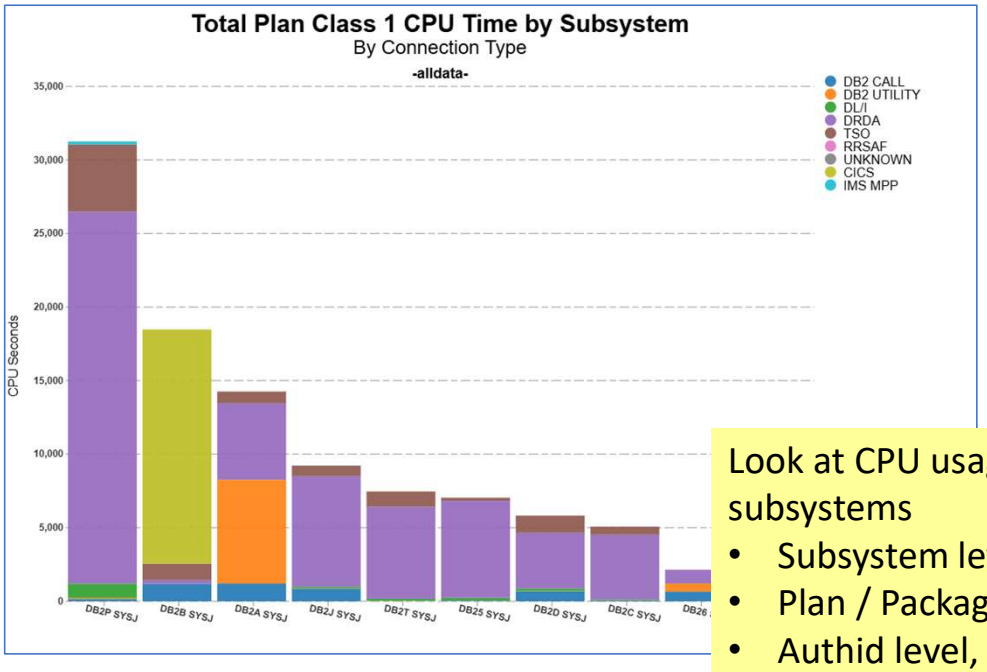


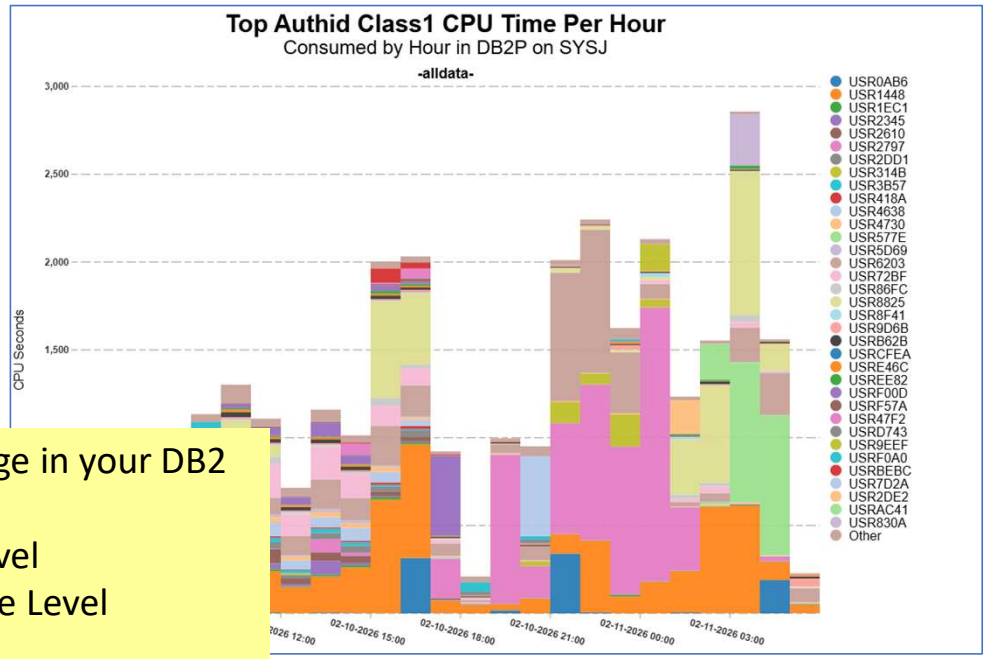


Look at CPU usage in your CICS subsystems

- Region level
- Transaction level

Look at your DB2 CPU Consumption





Look at CPU usage in your DB2 subsystems

- Subsystem level
- Plan / Package Level
- Authid level,
- Etc.

How many MIPS is that batch job?

- Essentially this question is: how much of my capacity is a particular job using?
- The answer is easy to calculate but depends on the time frame you want to consider:
  - Over the duration of the batch job?
  - Over the duration of the RMF interval? SMF Type 30 interval?
  - Over the duration of the hour?
  - Over the duration of the day?

Example Batch Job MIPS

- Example:
  - Job Z starts at 11:10 and runs for 14 minutes. The first interval record cut at 11:15 shows 3 minutes of CPU time. The second (and final interval) shows 4 minutes of CPU time, over that final 9 minutes of elapsed time. Total CPU time was 7 minutes. This is a 5 way machine rated at 840 MIPS.

	CPUs: 5		MIPS: 840		
Description	Duration	Max Possible CPU Time	Recorded CPU Time	Recorded / Max = Pct Total Capacity	Consumed MIPS
Course of Batch Job	14	70	7	10%	84.0
SMF 30 Interval 1	5	25	3	12%	100.8
SMF 30 Interval 2	9	45	4	9%	74.7
RMF Interval 1	15	75	3	4%	33.6
RMF Interval 2	15	75	4	5%	44.8
Hour	60	300	7	2%	19.6
Day	1440	7200	7	0%	0.8

Over the course of the batch job seems to make the most sense, but don't add those consumed MIPS up for all batch jobs over the course of an hour or day because you'll quite possibly end up with more than your total installed MIPS!

Batch MSU-hours

- Remember TFP and MSU-hours vs. MSUs?
- Maybe we'd be better off expressing that job consumption as MSU-hours
- Using previous example:
  - Assume 100 MSUs machine = 20 MSUs / engine
  - (7 / 60) * 20 = 2.3 MSU-hours
  - Or (7 / (60 * 5)) * 100 = 2.3 MSU-hours
- Note that because we're expressly using MSU-hours we can just do the calculation for the entire job, we're not as concerned about what interval
- While we used minutes for this example, more common to work in seconds

Your feedback is important!

Submit a session evaluation for each session you attend:

www.share.org/evaluation

Session TECH_201

SHARE Orlando
February 2026
Session TECH_201

Thanks!
Questions?

Email: peter.enrico@epstrategies.com